



VNIVERSITAT DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria

Departament d'Informàtica

Programa de Doctorat en Tecnologies de la Informació,  
Comunicacions i Computació

TESIS DOCTORAL

*VISUALIZACIÓN DEL CONTENIDO DE GRAFOS  
DE CONOCIMIENTO DEL PATRIMONIO  
CULTURAL*

Autor

Javier Sevilla Peris

Directores:

Dr. Jose Javier Samper Zapater

Dr. Marcos Fernández Marín

17 de mayo de 2022



A Loli, a mi hija Sara, a mis padres, al resto de familia y amigos,  
por todo el tiempo que no he podido compartir con ellos,  
lo más valioso que tenemos.





## ÍNDICE GENERAL

|   |      |
|---|------|
| ÍNDICE DE FIGURAS .....   | vi   |
| ÍNDICE DE TABLAS .....  | xi   |
| AGRADECIMIENTOS.....  | xiii |
| RESUMEN .....   | xiv  |
| SECCIÓN I: INTRODUCCIÓN.....  | 1    |
| CAPÍTULO 1. INTRODUCCIÓN: APROXIMACIÓN AL PROBLEMA .....  | 2    |
| 1.1. INTRODUCCIÓN .....   | 2    |
| 1.2. MOTIVACIÓN.....  | 3    |
| 1.3. PROBLEMAS EN GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL .....   | 5    |
| 1.3.1. Problema 1. Falta de uniformidad.....  | 5    |
| 1.3.2. Problema 2. Incerteza .....  | 6    |
| 1.3.3. Problema 3. Ausencia de herramientas abiertas de visualización interactivas de<br>datos de grafos de conocimiento..... | 7    |
| 1.4. PREGUNTAS DE INVESTIGACIÓN .....   | 8    |
| 1.5. ORGANIZACIÓN DE LA MEMORIA .....   | 9    |
| CAPÍTULO 2. OBJETIVOS Y METODOLOGÍA.....  | 11   |
| 2.1. INTRODUCCIÓN .....   | 11   |
| 2.2. OBJETIVO GENERAL.....  | 11   |
| 2.3. OBJETIVOS ESPECÍFICOS.....   | 11   |
| 2.4. METODOLOGÍA DE INVESTIGACIÓN.....  | 12   |
| SECCIÓN II. MARCO TEÓRICO. TECNOLOGÍA Y ESTADO DEL ARTE. ....   | 14   |
| CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS<br>DE CONOCIMIENTO.....                           | 15   |
| 3.1. RESUMEN .....  | 15   |
| 3.2. CONCEPTO DE GRAFO DE CONOCIMIENTO .....  | 15   |
| 3.2.1. Lenguajes relacionados con grafos de conocimiento .....  | 18   |
| Aplicaciones de grafo de conocimiento .....   | 23   |
| 3.3. GRAFOS DE CONOCIMIENTO EN EL PATRIMONIO CULTURAL .....   | 26   |
| 3.3.1. Situación.....   | 26   |
| 3.3.2. Vocabularios.....  | 30   |
| 3.3.3. Modelos de referencia. ....  | 31   |
| 3.4. CONCLUSIONES .....   | 33   |
| CAPÍTULO 4. VISUALIZACIÓN DE DATOS INTERACTIVA DE PATRIMONIO CULTURAL.....  | 35   |
| 4.1. RESUMEN .....  | 35   |
| 4.2. VISUALIZACIÓN DE DATOS .....   | 35   |

|   |    |
|---|----|
| 4.2.1. Clasificación de datos espacio temporales .....                                    | 36 |
| 4.2.2. Agrupamiento de objetos: Clustering. ....  | 37 |
| 4.2.3. Visualización de relaciones entre objetos.....                                     | 39 |
| 4.2.4. Técnicas de visualización interactivas para la selección del tiempo.....           | 41 |
| 4.3. APLICACIÓN DE ONTOLOGÍAS EN SISTEMAS DE VISUALIZACIÓN. ....                          | 45 |
| 4.3.1. Ontologías de visualización. ....  | 47 |
| 4.4. DESARROLLOS WEB PARA LA VISUALIZACIÓN DE DATOS DEL PATRIMONIO CULTURAL .....         | 51 |
| 4.4.1. The Museum of the World. ....  | 51 |
| 4.4.2. Cronobook.....   | 52 |
| 4.4.3. EAMENA Database .....  | 53 |
| 4.4.4. Chronas .....  | 55 |
| 4.4.5. Sanborn Maps Navigator.....  | 56 |
| 4.4.6. Jewish Cultures Mapped .....   | 56 |
| 4.4.7. Paleontology Information Explorer.....   | 58 |
| 4.4.8. Basados en Google Maps .....   | 58 |
| 4.4.9. Resultados .....   | 59 |
| 4.5. CONCLUSIONES .....   | 60 |
| CAPÍTULO 5. TECNOLOGÍAS DE INTEGRACIÓN .....  | 62 |
| 5.1. RESUMEN .....  | 62 |
| 5.2. HERRAMIENTAS DE GESTIÓN DE GRAFOS DE CONOCIMIENTO.....                               | 62 |
| 5.2.1. Virtuoso .....   | 63 |
| 5.2.2. RDF4J.....   | 63 |
| 5.2.3. StarDog.....   | 63 |
| 5.2.4. Jena + Fuseki.....   | 64 |
| 5.2.5. Resultados .....   | 64 |
| 5.3. HERRAMIENTAS DE VISUALIZACIÓN DE DATOS ESPACIO TEMPORALES sobre plataformas WEB..... | 65 |
| 5.3.1. WebGL.....   | 66 |
| 5.3.2. D3.JS .....  | 67 |
| 5.3.3. Three.js.....  | 68 |
| 5.3.4. Unity3d.....   | 68 |
| 5.3.5. Leaflet.....   | 69 |
| 5.3.6. OpenLayers.....  | 69 |
| 5.3.7. Cesium.JS.....   | 69 |
| 5.3.8. Resultados. ....   | 70 |

|  |     |
|--|-----|
| 5.4. CONCLUSIONES .....  | 70  |
| SECCIÓN III. PROPUESTAS Y CONTRIBUCIONES .....   | 72  |
| CAPÍTULO 6. MODELO DE REPRESENTACIÓN DE ESCENAS DE VISUALIZACIÓN DE DATOS ...  | 73  |
| 6.1. RESUMEN .....   | 73  |
| 6.2. ELEMENTOS DE UN SISTEMA DE VISUALIZACIÓN DE DATOS DE PATRIMONIO CULTURAL.....   | 73  |
| 6.2.1. Objetos visuales .....  | 74  |
| 6.2.2. Agrupamiento de objetos. ....   | 81  |
| 6.2.3. Procesos de interacción .....   | 84  |
| 6.3. PROBLEMAS DE VISUALIZACIÓN .....  | 88  |
| 6.3.1. Agrupamiento de objetos .....   | 89  |
| 6.3.2. Representación de las relaciones de un objeto. ....   | 91  |
| 6.3.3. Incerteza.....  | 92  |
| 6.4. ASOCIACIÓN DE LA VISUALIZACIÓN CON DATOS DE DOMINIO.....  | 95  |
| 6.5. CONCLUSIONES. ....  | 97  |
| CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN modelo DE VISUALIZACIÓN DEL contenido de GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL..... | 104 |
| 7.1. RESUMEN .....   | 104 |
| 7.2. DISEÑO. ....  | 104 |
| 7.2.1. Ontologías de visualización .....   | 106 |
| 7.2.2. Ontologías espaciales y temporales.....   | 115 |
| 7.2.3. Aplicación de METHONTOLOGY .....  | 123 |
| 7.3. IMPLEMENTACIÓN .....  | 151 |
| 7.4. CONCLUSIONES .....  | 152 |
| CAPÍTULO 8. DISEÑO Y DESARROLLO DE UN HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO .....        | 153 |
| 8.1. RESUMEN .....   | 153 |
| 8.2. MARCO DE TRABAJO .....  | 153 |
| 8.2.1. Procesos .....  | 155 |
| 8.2.2. Casos de uso.....   | 156 |
| 8.2.3. Arquitectura del sistema.....   | 157 |
| 8.2.4. Implementación .....  | 159 |
| 8.2.5. Comunicación.....   | 164 |
| 8.2.6. Componente de visualización .....   | 173 |
| 8.3. MÓDULO GRÁFICO.....   | 176 |

|   |     |
|---|-----|
| 8.3.1. Procesos necesarios para la visualización inicial.....                               | 177 |
| 8.3.2. Procesos activados mediante la interacción con el usuario en el Módulo gráfico ..... | 182 |
| 8.3.3. Procesos de finalización .....   | 184 |
| 8.4. CONCLUSIONES .....   | 184 |
| SECCIÓN IV. OTROS RESULTADOS.....   | 186 |
| CAPÍTULO 9. APLICACIÓN DE LOS RESULTADOS DE LA INVESTIGACIÓN .....                          | 187 |
| 9.1. RESUMEN .....  | 187 |
| 9.2. APLICACIÓN DE LOS RESULTADOS EN EL PROYECTO SILKNOW .....                              | 187 |
| 9.2.1. Proyecto SILKNOW .....   | 187 |
| 9.2.2. Instanciación de STEVO .....   | 192 |
| 9.3. APLICACIÓN DE LOS RESULTADOS EN EL PROYECTO ARXIU VALENCIÀ DE DISSENY .                | 210 |
| 9.3.1. Proyecto Arxiu Valencià del Disseny .....  | 210 |
| 9.3.2. Instanciación en STEVO .....   | 211 |
| 9.4. CONCLUSIONES .....   | 213 |
| CAPÍTULO 10. VALIDACIÓN DE LOS RESULTADOS: ARQUITECTURA E IMPLEMENTACIÓN...                 | 215 |
| 10.1. RESUMEN .....   | 215 |
| 10.2. PRUEBAS DE ACEPTACIÓN Y USABILIDAD .....  | 215 |
| 10.2.1. Introducción .....  | 215 |
| 10.2.2. Preparación de los cuestionarios .....  | 216 |
| 10.2.3. Selección de los usuarios.....  | 217 |
| 10.2.4. Preparación de los materiales de formación y envío de los cuestionarios.....        | 218 |
| 10.2.5. Evaluación de los resultados .....  | 218 |
| 10.3. PRUEBAS DE ESTRÉS.....  | 220 |
| 10.3.1. Introducción .....  | 220 |
| 10.3.2. Descripción de las pruebas.....   | 220 |
| 10.3.3. Parámetros a evaluar .....  | 221 |
| 10.3.4. Resultados de las pruebas.....  | 222 |
| 10.3.5. Conclusiones de los resultados .....  | 230 |
| 10.4. CONCLUSIONES .....  | 232 |
| SECCIÓN V. CONCLUSIONES, DISCUSIÓN Y TRABAJO FUTURO.....                                    | 234 |
| CAPÍTULO 11. RESULTADOS Y DISCUSIÓN .....   | 235 |
| 11.1. RESUMEN .....   | 235 |
| 11.2. RESULTADOS Y APORTACIONES .....   | 235 |
| 11.3. DISCUSIÓN.....  | 237 |
| 11.3.1. Evolución de la digitalización de los datos de patrimonio cultural.....             | 237 |

|  |     |
|--|-----|
| 11.3.2. Herramientas para la visualización de datos de patrimonio cultural ..... | 239 |
| 11.4. CONCLUSIONES .....   | 241 |
| CAPÍTULO 12. TRABAJO FUTURO .....  | 243 |
| 12.1. TRABAJO FUTURO .....   | 243 |
| 12.1.1 Visualización de grandes conjuntos de datos .....                         | 243 |
| 12.1.2. Ampliación de nuevas variables de tiempo y espacio sobre la escena.....  | 243 |
| 12.1.3. Definición de nuevas escenas y transiciones entre las mismas .....       | 244 |
| 12.1.4. Nuevos elementos de visualización .....                                  | 245 |
| 12.1.5. Funcionalidades de exportación .....                                     | 245 |
| 12.1.6. <b>Difusión</b> .....  | 246 |
| 12.1.7. Desarrollo de sistema de recomendación de visualización de datos .....   | 246 |
| GLOSARIO DE TÉRMINOS .....   | 247 |
| REFERENCIAS.....   | 251 |
| PUBLICACIONES.....   | 274 |
| DERIVADAS DE LA TESIS .....  | 274 |
| RELACIONADAS .....   | 275 |

## ÍNDICE DE FIGURAS

|  |    |
|--|----|
| Figura 1 Metodología de la investigación. ....   | 12 |
| Figura 2 Aplicaciones de los grafos de conocimiento. Fuente: Adaptada de (Zou 2020). ....  | 24 |
| Figura 3 Porcentaje de datos digitalizados en instituciones de patrimonio cultural europeo. Adaptada de (Jan Nauta, G., van den Heuvel, W. 2015) .....   | 28 |
| Figura 4 Porcentaje de datos digitalizados de patrimonio cultural europeo que están disponibles para su consulta online. Adaptada de (Jan Nauta, G., van den Heuvel, W. 2015) ..   | 28 |
| Figura 5 Ejemplo simplificado de las relaciones de una instancia en un grafo de conocimiento. Fuente: Elaboración propia. ....   | 31 |
| Figura 6 Ejemplos de agrupamiento de puntos, por color e indicando cardinalidad (a la izquierda), por mapas de color (a la derecha). Fuente: Capturas de videos publicados del algoritmo SuperCluster (Agafonkin 2016).....  | 38 |
| Figura 7 Ejemplo de agrupamiento de trayectorias en mapas de flujo. El color, grosor y la dirección aportan información adicional. Fuente: (Guo 2009).....   | 38 |
| Figura 8 Ejemplo de agrupamiento de color por regiones. Fuente: Incidencia de cáncer en Taiwan, adaptada de (Ku, Liaw et al. 2016). ....   | 38 |
| Figura 9 Ejemplo de combinación de mostrar relaciones con técnicas de agrupamiento. Arriba, burbujas de mismo color están relacionadas. Debajo, igual, pero con más elementos gráficos que ayudan a evitar problemas de sobre dibujado. Fuente: (Tomislav Hengl, Pierre Roudier et al. 2015).....                                    | 40 |
| Figura 10 A la izquierda, ejemplo de efectos del sobre dibujado en mapas de trayectoria. A la derecha, solución con reducción de la opacidad al 5%. Fuente:(Andrienko, G., Andrienko, N., Bak, P., Keim, D., Wrobel, S. 2013).....   | 41 |
| Figura 11 Captura de aplicación con línea de tiempo para visualizar hallazgos arqueológicos. Fuente: (Kraak 2005 ) .....   | 42 |
| Figura 12 Captura de aplicación con línea de tiempo para visualizar hallazgos de fósiles de un periodo. Fuente: (Wang, C., Ma, and Chen 2018b) .....   | 42 |
| Figura 13 Efecto ventana de tiempo. El mismo escenario, pero con datos relativos a diferentes intervalos de tiempo. Fuente: (Ku, Liaw et al. 2016). ....   | 43 |
| Figura 14 Izquierda, ejemplo de cubo espacio temporal geoespacial. Izquierda,Fuente:(Bach, Benjamin, Dragicevic et al. Jun 11, 2014).. Derecha, captura de aplicación que usa un cubo espacio temporal para ver cambios en edificios. Fuente: (Windhager F., Mayr E., Schreder, G., Smuc M., Federico, P. and Miksch, S. 2016) ..... | 44 |
| Figura 15 Captura de pantalla de la aplicación LayerCake. Fuente: (Aga Khan Documentation Center at, MIT 2019) .....   | 44 |
| Figura 16 Captura de pantalla de WebOWL, visor de ontologías online. Fuente:(Lohmann, Link et al. 2015) .....  | 46 |
| Figura 17 Visualización de ontologías con combinación de técnicas 2.5D con histogramas. Fuente:(Silva, Santucci et al. 2019) .....   | 46 |
| Figura 18 Ejemplos de visualización con VISO y VRL. A la izquierda sobre una ontología de aminoácidos, y a la derecha una ontología de plantas. Fuente:(Polowinski, J. 2015).....  | 49 |
| Figura 19 Esquemas de los prototipos de visualización definidos por VUMO. Fuente:(Sobral, Thiago, Galvão et al. 2020).....   | 50 |
| Figura 20 Captura de pantalla de la aplicación The Museum of the World. Fuente:(British Museum, Google Cultural Institute 2015).....   | 52 |
| Figura 21 Captura de pantalla de la aplicación Cronobook. Fuente:(Cronobook 2021).....   | 53 |

|   |    |
|---|----|
| Figura 22 Captura de pantalla del sistema de visualización de EAMENA Database.<br>Fuente:(Arcadia Fund, EAMENA Project, British Council 2018).....  | 54 |
| Figura 23 Rueda de tiempo interactiva de EAMENA Database. Fuente:(Arcadia Fund, EAMENA Project, British Council 2018) .....   | 54 |
| Figura 24 Captura de pantalla de la aplicación Chronas. Se aprecian marcadores de dos tipos, frontera de las creencias religiosas de la época, definidas por la línea de tiempo inferior.<br>Fuente:(Dietmar 2021b).....  | 55 |
| Figura 25 Captura de pantalla de Saborn Maps Navigator. Fuente: (Quian 2020b).....  | 56 |
| Figura 26 Captura de pantalla de la aplicación Jewish Cultures Mapped. Fuente:(Daat Hamakom 2017) .....   | 57 |
| Figura 27 Captura de pantalla del Explorador de Patrimonio de Cyark (izquierda) y Atlas Cultural de Australia (derecha). Fuentes respectivamente:(Cyark 2021, Stadler 2021) .....   | 59 |
| Figura 28 Comparativa de Rendimiento en herramientas de gestión de grafos de conocimiento. Fuente: (Nguyen ,H., Mau Quoc, Martin Serrano, Han Nguyen Mau, John G. Breslin, Danh Le Phuoc 2019) .....  | 65 |
| Figura 29 Soporte de WebGL en diferentes navegadores y sistemas operativos.Fuente:(Deveria, Lennart 2022) .....   | 67 |
| Figura 30 Diferentes formas de representar un objeto en escenas de varias dimensiones y usando diferentes propiedades del objeto para mapear su posición en las dimensiones de la escena, que además tiene una parte visualizada (en negro) diferenciada de su totalidad.<br>Fuente: Elaboración propia. ....   | 75 |
| Figura 31 Los contenidos de la escena se actualizan por otros con mejor resolución conforme disminuye la distancia del punto de observación. El tamaño de la escena en el dispositivo de visualización se mantiene. ....  | 80 |
| Figura 32 Ejemplo de solapamiento de objetos cuando no hay agrupación. Al alejar el punto de visión (izquierda) sólo se distingue una nube de marcadores que no aporta información.<br>Fuente:(Helsingin kaupungin museo 2015) .....  | 82 |
| Figura 33 Esquema de técnicas de agrupación de objetos (clustering) más usadas. A la izquierda, con marcadores representantes y a la derecha mapas de calor. Fuente: Elaboración propia. ....   | 83 |
| Figura 34 Ejemplo de combinación de mapa de calor y agrupamiento por marcadores. Fuente: Maptive (Matpive 2022).....  | 84 |
| Figura 35 Esquema de aplicación de técnicas para mostrar los objetos relacionados con los objetos A y B, a través de diferentes propiedades (marcadas en azul y verde). En la parte superior se usa la técnica de marcar los objetos relacionados gráficamente y en la parte inferior unirlos mediante líneas, o arcos. Fuente: Elaboración propia..... | 87 |
| Figura 36 Esquemas de visualización simultánea. A la izquierda se simboliza el modo video, a la derecha dos hipercubos, el primero con capas, y el segundo sencillo. Fuente: Elaboración propia. ....   | 88 |
| Figura 37 Ejemplo de disposición de grupo en espiral. Usada cuando los objetos están en una misma posición espacial. Fuente: Captura de pantalla de la página web del Proyecto Semap. ....  | 89 |
| Figura 38 Ejemplo de navegación jerárquica interactiva sobre objetos categorizados. Fuente: Adaptado de Bigmedilytics Blueprint (Bigmedilytics 2021) .....  | 90 |
| Figura 39 Esquema de estructura en anillo para indicar el porcentaje de objetos relacionados. Fuente: Elaboración propia. ....  | 92 |
| Figura 40 Ejemplo de cambio de marcador en entornos 3D (izquierda) y en entornos 2D (derecha) para indicar que es un objeto con incerteza. ....   | 93 |

|  |     |
|--|-----|
| Figura 41 Ejemplo de cambio de marcador en entornos 3D (izquierda) y en entornos 2D (derecha) para indicar que es un objeto con incerteza replicado. ....  | 94  |
| Figura 42 Diagrama con la conceptualización de un elemento X y una propiedad (material), en el sistema de visualización (izquierda) y en el grafo de conocimiento de dominio basado en CIDOC-CRM (derecha). Fuente: Elaboración propia. .... | 96  |
| Figura 43 Esquema de elementos de diseño básicos de la ontología de visualización propuesta. Fuente: Elaboración propia. ....  | 111 |
| Figura 44 Esquema de las principales clases del módulo GRAPHICS de VISO. Fuente: Elaboración propia. ....  | 112 |
| Figura 45 Diagrama de la ontología de Sistemas de Coordenadas del IGF (Abadie, Atemezing 2015). ....   | 116 |
| Figura 46 Diagrama de la ontología TIME (W3C 2020) ....  | 122 |
| Figura 47 Tareas de la conceptualización en METHONTOLOGY. Fuente:(Corcho, Fernández-López et al. 2005) ....  | 124 |
| Figura 48 Relación taxonómica de Descomposición-Exhaustiva y Partición, entre el concepto Comportamiento y sus subconceptos.....   | 131 |
| Figura 49 A la izquierda, relación taxonómica de Partición entre el concepto Marcador y sus subconceptos, a la derecha relación taxonómica de Descomposición Exhaustiva del Concepto Modificador y sus subconceptos.....                     | 131 |
| Figura 50 Relación taxonómica de Partición entre el concepto Entorno y sus subconceptos. ....  | 131 |
| Figura 51 Relación taxonómica de Partición entre el concepto Sistema de Referencia Espacial y sus subconceptos. ....   | 132 |
| Figura 52 Relación taxonómica del concepto Representación gráfica y sus subconceptos. ....   | 132 |
| Figura 53 Relación taxonómica de los conceptos Sistema de Referencia Temporal, Duración Temporal, Entidad Temporal y sus subconceptos. ....  | 133 |
| Figura 54 Relaciones binarias ad hoc entre los conceptos Concepto Visual y Propiedad Visual, así como con otros conceptos utilizados para asociar estos conceptos a la ontología de dominio. ....  | 133 |
| Figura 55 Relaciones binarias ad hoc de los conceptos Entorno Temporal y Entorno Espacial con los conceptos que los definen y sus respectivos sistemas de referencia.....  | 134 |
| Figura 56 Relaciones binarias ad hoc del concepto de Escena con Comportamiento, Concepto Visual y los subconceptos de Entorno.....   | 135 |
| Figura 57 Relaciones binarias ad hoc del concepto de Concepto Visual con Marcador, que a su vez está relacionado con Representación gráfica.....   | 136 |
| Figura 58 Esquema de generación de valores para dos propiedades visuales de un concepto visual, sobre la misma consulta.....   | 138 |
| Figura 59 Captura de pantalla de la aplicación Protege, con la inferencia de la ontología STEVO, tras aplicar el razonamiento HermiT.....  | 151 |
| Figura 60 Esquema de funcionamiento de sistema de visualización web usando ontología STEVO. ....   | 154 |
| Figura 61 Procesos del marco de trabajo para visualizar el contenido de un grafo de conocimiento mediante la ontología STEVO.....  | 156 |
| Figura 62 Diagrama de casos de uso del usuario con la aplicación web de visualización. ....  | 157 |
| Figura 63 Arquitectura del marco de trabajo.....   | 158 |
| Figura 64 Diagrama con la tecnología propuesta para el desarrollo de los componentes del marco de trabajo y su comunicación con el framework web y el resto de elementos necesarios. ....  | 163 |
| Figura 65 Diagrama de clases de los módulos de la aplicación web de visualización. ....  | 166 |



|  |     |
|--|-----|
| Figura 66 Diagrama de estados de la aplicación web de visualización.....   | 168 |
| Figura 67 Diagrama de la comunicación entre VisualizationComponent y el Módulo gráfico. ....   | 176 |
| Figura 68 Diagrama con los procesos que se realizan en el Módulo gráfico y los flujos de comunicación que emiten y reciben. ....   | 177 |
| Figura 69 Diagrama de clases con las principales clases del Módulo Gráfico.....  | 178 |
| Figura 70 Esquema de funcionamiento de Interfaz de filtrado en Módulo Gráfico. ....  | 182 |
| Figura 71 Esquema de pantalla de información de objeto del Módulo gráfico, e interfaz para solicitar datos relacionados.....   | 183 |
| Figura 72 Diagrama con los resultados del proyecto SILKNOW. Fuente:(Portalés, Sebastián et al. 2018).....  | 189 |
| Figura 73 Captura de pantalla de los resultados del telar virtual del proyecto SILKNOW: Fuente:(Portalés, Pérez et al. 2021).....  | 190 |
| Figura 74 Diagrama sobre captura de pantalla del mapa espacio temporal para indicar las diferentes áreas del componente. Fuente:Adaptada de ADASILK (SILKNOW Consortium 2022) .....  | 191 |
| Figura 75 Diagrama con relaciones y propiedades de instancia de Escena en STEVO de la escena de SILKNOW .....  | 193 |
| Figura 76 Interfaz de usuario de mapa espacio temporal mostrando las propiedades por las que se puede filtrar y sus posibles valores. Fuente: ADASILK (SILKNOW Consortium 2022) ..   | 194 |
| Figura 77 Captura de pantalla con ventana emergente con información del objeto representado en el mapa espacio temporal de SILKNOW. Fuente: ADASILK (SILKNOW Consortium 2022).....   | 195 |
| Figura 78 Diagrama con conceptos (fondo gris) e instancias de la ontología STEVO para representar los conceptos visuales y sus propiedades visuales para representar los datos del proyecto SILKNOW .....  | 197 |
| Figura 79 Representación gráfica de los conceptos representados en el mapa espacio temporal. Fuente: ADASILK (SILKNOW Consortium 2022).....  | 203 |
| Figura 80 Zoom con agrupamientos sobre mapa espacio temporal de ADASILK. Composición con capturas de pantallas de ADASILK.Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022) .....  | 204 |
| Figura 81 Captura de ADASILK mostrando la interfaz para seleccionar los diferentes modos de control de tiempo y sus parámetros. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022). ....  | 205 |
| Figura 82 Diferentes capturas de pantallas con el control de tiempo "TimeLine" activo en la parte superior sobre un mismo conjunto de datos. Cada captura corresponde a un siglo diferente. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022) .....  | 206 |
| Figura 83 Capturas de pantalla de visualización de tiempo simultáneo en mapa SILKNOW. Cada capa representa un intervalo de tiempo (en siglos). Los botones de la derecha permiten mover las capas de arriba a abajo para mejor visualización. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022).....                         | 207 |
| Figura 84 Figura compuesta por dos capturas de pantalla en las que se muestran las relaciones de un objeto con el resto. La imagen superior es a nivel de zoom alto, y la inferior es a un nivel de zoom más bajo, las relaciones se muestran entre las agrupaciones. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)..... | 208 |
| Figura 85 Captura de pantalla del mapa espacio temporal de SILKNOW. En esta imagen se aprecia el anillo de con las relaciones totales de los objetos. Se aprecia la leyenda de propiedades que muestra que el 100% tienen el mismo material. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022).....                          | 209 |

|   |     |
|---|-----|
| Figura 86 Captura de pantalla del portal web prototipo de la web d l'arxiu. Fuente: Web de AVD (U. València 2022) .....   | 212 |
| Figura 87 Herramienta de visualización del mapa de producto del portal de l'Arxiu. La visualización ha sido representada con la ontología STEVO.Fuente: Web de AVD (U. València 2022) .....   | 213 |
| Figura 88 Distribución de los participantes de la evaluación por género, lenguaje, formación y edad. Fuente: Deliverable D7.3 proyecto SILKNOW .....  | 218 |
| Figura 89 Resultados de pruebas SUS por ámbito profesional del entrevistado. Fuente: Deliverable D7.3 de proyecto SILKNOW. ....   | 219 |
| Figura 90 Capturas de pantalla de la herramienta de evaluación con los diferentes conjuntos de datos (1:301, 2:3001, 3:15001, 4: 32249) .....   | 221 |
| Figura 91 Gráfica con los tiempos de ejecución para la obtención y procesamiento de datos en las pruebas de carga.....  | 224 |
| Figura 92 Gráfica con tiempos de ejecución del proceso de generación de marcadores gráficos en las pruebas de carga.....  | 225 |
| Figura 93 Gráfica con tiempos de ejecución del proceso de generación de agrupaciones. ....  | 226 |
| Figura 94 Gráfica con tiempos de ejecución para la generación de estructuras de relaciones durante las pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos. ....  | 227 |
| Figura 95 Gráfico con los tiempos de ejecución para generar relaciones anillo en pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.....   | 228 |
| Figura 96 Gráfica con los resultados de velocidad de refresco de la aplicación en cuadros por segundo (FPS) obtenida durante las pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos. ....                | 229 |
| Figura 97 Memoria reservada por el componente de visualización durante las pruebas de carga. Las barras punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.....   | 230 |
| Figura 98 Memoria usada por el componente de visualización durante las pruebas de carga. Las barras punteadas indican que no hay datos para ese ordenador en ese conjunto de datos. ....  | 230 |
| Figura 99 Gráfica con el tiempo total del proceso de carga del componente de visualización, durante las pruebas de carga. La línea roja punteada indica el umbral a partir del cual no es tolerable más tiempo, en opinión del creador de esta investigación..... | 232 |
| Figura 100 Captura de pantalla de los resultados del proyecto SEMAP. Vista helicoidal del contenido de una agrupación con 263 objetos de un yacimiento de Jaén. Fuente: Semap (U.Valencia 2021) .....   | 244 |
| Figura 101 Ejemplo de posibles escenas integrables con la herramienta propuesta. A la izquierda un libro desarrollado con Turn.JS (Garcia 2013), y a la derecha una galería 3D (Hindle 2017) generada a partir de imágenes, con Three.JS. ....                    | 245 |
| Figura 102 Representación del tamaño de las agrupaciones mediante colores (izquierda), o tamaño de los símbolos (derecha). Fuente: Semap (U.Valencia 2021). ....  | 245 |

## ÍNDICE DE TABLAS

|   |     |
|---|-----|
| Tabla 1 Número de instancias, Hechos y tipos de Entidades y Relaciones de los grafos de conocimiento más populares. (Paulheim 2017) .....   | 17  |
| Tabla 2 Tabla con la lista de tareas, o consultas posibles, por cada nivel. Fuente: (Wang, D., Guo et al. 2017) .....   | 36  |
| Tabla 3 Resumen de los datos obtenidos a partir del análisis de las aplicaciones de visualización web de patrimonio cultural.....   | 59  |
| Tabla 4 Características de librerías de visualización de datos espacio temporales en web. ....  | 70  |
| Tabla 5 Diferentes tipos de representaciones de objetos en la escena. Para cada tipo, además de visualizar un ejemplo se expone su transformación de escena 2D a escena 3D. ....  | 76  |
| Tabla 6 Ejemplos de las diferentes propiedades temporales y espaciales asociadas a un objeto de patrimonio cultural. ....   | 78  |
| Tabla 7 Tipos de incerteza en los objetos de patrimonio cultural y sus motivos.....   | 93  |
| Tabla 8 Principales conceptos del modelo de un Sistema de visualización de datos de patrimonio cultural .....   | 98  |
| Tabla 9 Propiedades del concepto Escena. ....   | 100 |
| Tabla 10 Atributos y Relaciones del concepto Elemento Visual .....  | 101 |
| Tabla 11 Atributos y Relaciones del concepto Marcador .....   | 102 |
| Tabla 12 Atributos y Relaciones del concepto Agrupación.....  | 102 |
| Tabla 13 Descripción del objeto gráfico Marcador del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados. ....  | 113 |
| Tabla 14 Descripción del objeto gráfico Escena del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados.....   | 114 |
| Tabla 15 Descripción del objeto gráfico Agrupación del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados.....   | 115 |
| Tabla 16 Objetos del modelo a representar y su correspondencia con los conceptos de la ontología de Sistemas de Coordenadas del IGF que serán utilizados.....   | 117 |
| Tabla 17 Términos temporales usados en patrimonio cultural y conceptos relacionados con ontología TIME .....  | 120 |
| Tabla 18 Objetos del modelo y conceptos de la ontología TIME utilizados y/o extendidos. ...   | 120 |
| Tabla 19 Selección del Glosario de términos de actividad de conceptualización de METHONTOLOGY para construir ontología de visualización. ....   | 125 |
| Tabla 20 Diccionario de conceptos para los conceptos Concepto_Visual, Propiedad_Visual y Consulta.....  | 136 |
| Tabla 21 Diccionario de conceptos para los conceptos Marcador, Modificador y Representación_Gráfica.....  | 139 |
| Tabla 22 Diccionario de conceptos para los conceptos Escena, Entorno Espacial, Sistema Referencia Espacial, Sistema de Coordenadas, Sistema de Referencia de Coordenadas, Punto Espacial, Entorno Temporal, Unidad Temporal, Sistema de Referencia Temporal, e Instante de Tiempo. .... | 140 |
| Tabla 23 Diccionario de conceptos para los conceptos de Comportamiento y sus extensiones. ....  | 142 |
| Tabla 24 Tabla de relaciones binarias ad hoc.....   | 143 |
| Tabla 25 Propiedades de datos. ....   | 145 |
| Tabla 26 Axiomas de la ontología STEVO.....   | 147 |
| Tabla 27 Reglas de la ontología STEVO. ....   | 148 |

|   |     |
|---|-----|
| Tabla 28 Instancias de la ontología STEVO.....  | 149 |
| Tabla 29 Librerías para acceder a RDF con JS y con NODE.JS . Fuente:(W3C 2019) .....  | 161 |
| Tabla 30 Formato JSON usado en los parámetros stevoAddress, domainAddress y domainData .....  | 167 |
| Tabla 31 API de la Interfaz pública de la clase VisualizationApplication .....  | 169 |
| Tabla 32 API del Gestor de Datos de la aplicación web de visualización.....   | 171 |
| Tabla 33 API del Componente de Visualización de la aplicación web de visualización .....  | 172 |
| Tabla 34 API de la Interfaz entre los diferentes módulos del Componente de Visualización. .   | 173 |
| Tabla 35 Propiedades y relaciones de la propiedad visual visualTextileMaterial.....   | 196 |
| Tabla 36 Propiedades de la instancia de Query, queryForTextileMatyTec.....  | 196 |
| Tabla 37 Propiedades y relaciones de la propiedad visual visualTextileTechnique.....  | 197 |
| Tabla 38 Propiedades y relaciones de la instancia de propiedad visual visualTextileCategory   | 198 |
| Tabla 39 Relaciones y propiedades de la instancia de propiedad visual<br>visualTextileProductionName .....  | 199 |
| Tabla 40 Relaciones y propieades de la instancia de propiedad visual visualTextilePosX.....   | 199 |
| Tabla 41 Relaciones y propiedades de la instancia de Query, queryForTextileProduction.....  | 199 |
| Tabla 42 Relaciones y propiedades de datos de la instancia visualTextileTimeStart .....   | 201 |
| Tabla 43 Relaciones y propiedades de datos de la instancia queryForTextileTime.....   | 201 |
| Tabla 44 Relaciones y propiedades de la Instancia de circleTextile.....   | 202 |
| Tabla 45 Relaciones y propiedades de la instancia de Marcador silknowTextileMarker .....  | 202 |
| Tabla 46 Relaciones y propiedades de la instancia correspondingTextileTypeClothes del<br>concepto Modificador.....  | 203 |
| Tabla 47 Propiedades y relaciones de la instancia silknowGroupBehaviour<br>(Clusters_Group_Behaviour) para representar cómo visualizar los agrupamientos..... | 204 |
| Tabla 48 Propiedades y relaciones de la instancia silknowTimeLineControl (Control_Individual)<br>.....  | 205 |
| Tabla 49 Relaciones y Propiedades de la instancia silknowSimultaneousTimeControl<br>(Control_Simultaneo).....   | 206 |
| Tabla 50 Relaciones y propiedades de la instancia silknowLInkedRelationBehaviour<br>(Relacion_Enlaces) .....  | 207 |
| Tabla 51 Relaciones y propiedades de la instancia silknowUncertaintyBehaviour<br>(Tratamiento_Incerteza_Con_Replicacion) .....                                | 209 |
| Tabla 52 Relaciones y propiedades de la instancia silknowTime (Entorno_Temporal).....   | 210 |
| Tabla 53 Preguntas de los cuestionarios de aceptación y usabilidad. Fuente: Adaptado de<br>Deliverable D.7.3 de proyecto SILKNOW. ....                        | 217 |
| Tabla 54 Resultados de las pruebas SUS por idioma de entrevista. Fuente: Deliverable D.7.3 de<br>proyecto SILKNOW. ....                                       | 219 |
| Tabla 55 Conjunto de datos de prueba para realizar las pruebas de carga de la herramienta.<br>Fuente: Deliverable D.5.7 de SILKNOW. ....                      | 220 |
| Tabla 56 Descripción de hardware usado en pruebas de carga .....  | 221 |
| Tabla 57 Recursos y velocidad obtenidos para las pruebas de carga de herramienta de<br>visualización. ....  | 223 |
| Tabla 58 Tiempo de procesos iniciales obtenidos para las pruebas de carga de la herramienta<br>de visualización .....   | 223 |

## AGRADECIMIENTOS

Tengo mucho que agradecer a Cristina Portalés Ricart. Primero por involucrarme en el mundo del patrimonio cultural y muchas de las áreas tecnológicas con las que está relacionado, especialmente la visualización de datos. Segundo por empujarme y animarme a hacer esta investigación, sin este empuje y ánimo nunca habría existido. Por último, lo más importante, por su amistad. Muchas gracias Cris.

Agradecer la labor de coordinación y metodología de mis directores Jose Javier Samper Zapater y Marcos Fernández Marín. Gracias por animarme a publicar, orientarme en cualquier duda que he tenido y también cuando no las tenía y había que tenerlas.

También quiero agradecer la colaboración, los conocimientos que han compartido y los buenos ratos pasados a todo los compañeros y compañeras de la Facultad de Geografía e Historia que han trabajado conmigo en el proyecto SILKNOW: Arabella León, Mar Gaitán, Jorge Sebastian y Esther Alba, así como a los compañeros de mi grupo de investigación que también han participado en el proyecto: Manolo Pérez y Jesús Gimeno.

Quiero dar las gracias también a mi compañero Pablo Casanova, por su trabajo y colaboración en esta investigación y por ayudarme a ponerme al día en las tecnologías de visualización en tiempo real. Sin él, el resultado de esta investigación, así como los obtenidos en los proyectos relacionados hubieran sido mucho peores.

Gracias a Enrique Bonet y Susana Pons por su ayuda en todas las labores que han realizado, sin su consejo y ayuda, hubiera sido imposible poder poner en marcha todos los sistemas que ha requerido esta investigación.

Por último, agradecer al grupo de investigación ARTEC, a IRTIC y a sus responsables, en especial a Marcos Fernández Marín, como director del grupo ARTEC y también de IRTIC, las facilidades que me han proporcionado para poder realizar este trabajo, la confianza y su apoyo incondicional en todo momento.

Muchas gracias a todos.

## RESUMEN

Esta tesis integra dos áreas científicas, la de las Ciencias Humanas y la de Ciencia y Tecnología.

Dentro de las Ciencias Humanas está relacionada con la semantización y visualización de la información de datos del Patrimonio. En cuanto a la Ciencia y Tecnología, este trabajo está relacionado con grafos de conocimiento, diseño de ontologías y la visualización interactiva en tiempo real de datos en dos y tres dimensiones, mediante plataformas web.

Tal y como se expone en esta tesis, en la última década ha crecido enormemente la digitalización de los objetos del patrimonio cultural. Este proceso, no se refiere únicamente a digitalizar mediante imágenes, reconocimiento de texto, o modelos 3D, de este tipo de objetos. También comprende el mantenimiento de toda la información posible relacionada con los objetos, su publicación y relación con otras entidades (personas, hechos, materiales, etc.) y la accesibilidad de los mismos a otras personas y/o sistemas informáticos. Por ello, muchos de los datos de objetos son soportados por grafos de conocimiento, que implementan modelos que representan este tipo de información.

A pesar de que hay mucha información soportada en diferentes grafos de conocimiento, queda mucho que hacer en este proceso de digitalización del patrimonio. En la actualidad todavía existen grandes cantidades de objetos que aún no están digitalizados. Además, muchos de los datos informatizados no están disponibles públicamente, y usan medios propietarios para representar la información, lo que dificulta su acceso y tratamiento.

Debido a que muchos trabajos de investigación se han desarrollado de forma paralela, y/o se han enfocado en las particularidades del mismo, se han digitalizado, y en ocasiones creado desde cero, todo tipo de vocabularios para manejar la información de tipologías, materiales, técnicas, épocas, ubicaciones, etc., relacionado con estos objetos, muchos de ellos con información duplicada. Además, el modelo más utilizado para representar este tipo de datos es el CIDOC-CRM, aún necesita mayor definición para representar algunos procesos importantes relacionados con estos objetos, y tiene una complejidad excesiva. Por estos motivos, existen otros modelos alternativos que también se están utilizando.

Debido también a la rapidez de este proceso de digitalización, se ha manifestado en esta década la necesidad de visualizar grandes cantidades de información, mediante plataformas web, que permitan visualizar los datos y relaciones de estos objetos, así como su fluctuación a nivel temporal y espacial. Esta necesidad ha generado el desarrollo de varios proyectos, que generalmente la han resuelto de forma concreta, orientándose al contexto del proyecto. Además, los desarrollos de estos proyectos, no suelen ser abiertos, ni reutilizables por otros sistemas. En estos trabajos existen áreas de trabajo que han sido abordadas con poca profundidad, como la visualización de la fluctuación del tiempo de forma simultánea y la visualización de relaciones entre los objetos.

Las dificultades relacionadas con el tiempo y el espacio se hacen mayores en los objetos del Patrimonio Cultural. Esto es debido a que en este campo es muy habitual que exista incerteza en la datación de los hechos relacionados con los datos (creación de los objetos, hechos históricos, etc.), al igual que con el lugar en que acontecieron. Es muy normal que existan objetos que se asocie su creación a diferentes regiones, incluso continentes,

además de estar datados en diferentes momentos del tiempo, en ocasiones con varios siglos de diferencia.

Tras descubrir la problemática existente, la presente tesis tiene como objetivo principal, la definición de un modelo de visualización de datos de patrimonio cultural que estén soportados por un grafo de conocimiento, y una ontología que lo represente, para diseñar y desarrollar un sistema de visualización interactivo web, basado en el contenido de esta ontología.

Para alcanzar este objetivo se ha realizado una serie de tareas y obtenido algunos resultados fruto de éstas:

- Revisión del estado del arte en varios campos relacionados con las áreas de investigación que se integran en este trabajo. Por una parte, la representación de datos de patrimonio cultural mediante grafos de conocimiento, por otra parte, analizar las técnicas de visualización de datos, especialmente en datos donde el espacio y el tiempo son variables clave. También se han analizado diferentes proyectos de visualización de datos de patrimonio cultural sobre plataformas web. Además, también se ha investigado la definición formal de la visualización de datos mediante ontologías. Por último, se han estudiado y expuesto en el trabajo diferentes herramientas para la gestión de grafos de conocimiento, así como de visualización de datos en dos y tres dimensiones en tiempo real sobre páginas web.
- En base a los estudios realizados en el estado del arte, se han expuesto los aspectos necesarios para definir los conceptos de un modelo que defina la visualización de datos de objetos de patrimonio cultural soportados por grafos de conocimiento.
- Se ha diseñado una ontología denominada STEVO para representar el modelo de visualización diseñado. Para el proceso de diseño se ha seguido la metodología METHONTOLOGY.
- Se ha diseñado la arquitectura de un marco de trabajo capaz de procesar el contenido de la ontología STEVO y poder representar el contenido del dominio de datos, de acorde a lo especificado en STEVO.
- Se ha desarrollado una propuesta de implementación del marco de trabajo, mediante tecnología JavaScript y Unity3D.

La propuesta de implementación ha sido integrada en dos proyectos de investigación: SILKNOW y Arxiu Valencià del Disseny, cuyos resultados se han expuesto de forma detallada en esta memoria.

Finalmente, la implementación ha sido evaluada en el proyecto SILKNOW, primero mediante una evaluación de usabilidad, mediante la escala SUS, y también con diferentes pruebas de carga sobre los resultados del proyecto SILKNOW. Los resultados de dichas evaluaciones se han descrito detalladamente en esta investigación.

## SECCIÓN I: INTRODUCCIÓN.



## CAPÍTULO 1. INTRODUCCIÓN: APROXIMACIÓN AL PROBLEMA

### 1.1. INTRODUCCIÓN

El estudio y la preservación del patrimonio cultural son objetivos fundamentales en nuestra sociedad. Estudiar su evolución y su influencia en nuestros días es un pilar básico para entender nuestra sociedad.

Por ejemplo, un tejido realizado en seda puede parecer algo insignificante, un artículo de lujo, más si hablamos de los siglos XV, o XVI. Sin embargo, pocos materiales como la seda han tenido tanta presencia económica, técnica, funcional, cultural y simbólica. Ha estado presente desde banderas y tapices, hasta muebles, abanicos y vainas de espadas. Abarcó varios aspectos, como el comercio, la ciencia, la artesanía, el arte, la diplomacia y la cultura. En la actualidad sigue estando presente, por ejemplo, en trajes nacionales y vestidos de novia. El lujo, la artesanía, la belleza y el confort han estado y siguen estando asociados a este artículo. Sin embargo, este tipo de tejidos es un patrimonio más frágil que otros bienes culturales más conocidos y está en grave peligro de extinción. Muchas de las entidades que lo mantienen tienen graves problemas económicos para seguir funcionando, y para muchas es impensable la creación de recursos digitales de última generación que permitieran difundirlo, o al menos preservarlo digitalmente. El Musée des Tissus de Lyon, una de las mejores colecciones de este tipo con más de 150 años de existencia ha estado muy cerca de cerrar recientemente, debido a la escasez de financiación.

Muchas de las instituciones de conservación del patrimonio cultural, algunas de las más grandes y conocidas, pero sobre todo las de pequeño y mediano tamaño, cuentan con colecciones digitales de sus piezas. Estas colecciones, sin embargo, no son accesibles para el público en general y mucho menos para otras aplicaciones. Están clasificados con diferentes herramientas, en diferentes idiomas, clasificados sin un patrón común. Aunque quedan muchos materiales y piezas por digitalizar, en la actualidad hay enormes y heterogéneas bases de datos digitalizadas.

Esta información es conocida por las instituciones europeas, que están apostando fuertemente para solucionar este problema, desde que a finales de 2011 se creó la recomendación para preservar, digitalizar y facilitar el acceso online a los datos de patrimonio cultural (European Commission 2021), por ello:

- Promueve y financia iniciativas como el portal Europeana Collections (EUROPEANA 2021) que da acceso a más de 50 millones de archivos digitalizados
- Ha financiado con más de 70 millones de euros 6 programas bajo Horizon 2020 Societal Challenges entre 2014 y 2020.

El procesamiento y la comprensión de esta información es un reto para los campos de Big Data y Visualización de datos. Sin embargo, para una mejor explotación de esta información es imprescindible que esté accesible y a ser posible clasificada. Esto ha motivado que en la última década se haya detectado la necesidad de definir modelos de referencia como CIDOC-CRM (Bekiari, C, Bruseker, G., Doerr, M., Ore, C., Stead, S., Velios, A. 2021), o CHARM (González-Pérez 2018), que sirvan de base para implementar grafos de conocimiento con esta información.

Estos avances e impulsos, han ocasionado la generación de varios proyectos relacionados con datos de patrimonio cultural y grafos de conocimiento (Nafis, Yahyaouy et al. 2019, Data for History Consortium 2018) , con el objetivo de preservar y difundir esta información de forma digitalizada. Una gran parte de estos proyectos, permite analizar esta información mediante diversas herramientas de visualización de datos. El problema es que hay muy pocos de estos proyectos que hayan diseñado sus herramientas con un propósito general, más allá de las necesidades del proyecto y que estén abiertos a la comunidad de investigación.

La visualización de datos de grafos de conocimiento de patrimonio cultural es un campo en el que se han hecho importantes avances y desarrollos. En este campo, existe una necesidad de investigación y desarrollo en el diseño de herramientas genéricas que permitan visualizar su contenido.

## 1.2. MOTIVACIÓN

La participación en varios proyectos relacionados con la visualización de datos de grafos de conocimiento del patrimonio cultural, me ha otorgado un conocimiento importante sobre la problemática existente a nivel tecnológico en esta área. También he conocido como las instituciones europeas afrontan esta situación, las líneas de trabajo e investigación que están impulsando para superar el reto, así como los inconvenientes que han surgido en el proceso.

El patrimonio cultural es un área muy grande y muy importante de nuestra sociedad. A pesar de que existen muchos esfuerzos para mantenerlo y mejorar su estado, he conocido de primera mano los graves problemas que tienen instituciones de renombre y como la tecnología es cada vez una herramienta más importante para investigadores del campo de las humanidades.

Trabajar con estos investigadores me ha transmitido que tan importante es preservar el conocimiento, como difundirlo y ponerlo al alcance de toda la sociedad. Para ello, la tecnología es una herramienta fundamental. Existen cantidades ingentes de información en todas las áreas. Poder analizarla y extraer relaciones entre los datos es un pilar básico para seguir avanzando en su trabajo. Permitir el acceso a esta información a cualquier persona de la sociedad de forma sencilla y rápida da la opción de sectores como el turismo, la educación y el ocio se interesen de una forma más profunda por el patrimonio cultural. Conociéndolo y valorando este patrimonio es la mejor forma de mantenerlo.

La empresa iSOCO, S.A., fundada en 1999, se dedica a la investigación y a la aplicación de soluciones software de Inteligencia Artificial. Mi trabajo en el departamento de investigación de la empresa durante cinco años me dio la oportunidad de conocer el mundo de la inteligencia artificial y los grafos de conocimiento, participando en varios proyectos internacionales de investigación en ese ámbito (CORDIS 2005, Contreras, Benjamins et al. 2004) .

El grupo ARTEC (Advanced Research and Technological Expansion in Computer Graphics) es un equipo de investigación perteneciente al IRTIC (Instituto de Robótica y Tecnologías de la Información y Comunicación) de la Universidad de Valencia, dedicado desde su fundación en 1992 a los gráficos 3D interactivos, la Realidad Virtual, Realidad Aumentada y Simulación Civil. Se trata de un equipo multidisciplinar formado por unas 20 personas compuesto por Informáticos, Físicos, Matemáticos, Artistas Informáticos y Psicólogos. Desde 2005 he participado en este grupo en varios proyectos nacionales e internacionales en el área de realidad virtual y aumentada, con el objetivo de mejorar la calidad de vida de personas con autismo. En muchos de estos proyectos se utilizaron grafos de conocimiento para conocer las necesidades de estas personas y las tecnologías que necesitan (Sevilla, Javier, Samper et al. 2018).

Desde 2018 he participado en varios proyectos donde mi principal tarea ha sido la investigación y desarrollo de herramientas de visualización de datos de grafos de conocimiento del patrimonio cultural. Esto, junto con el hecho de participar en la docencia de la asignatura “Visualización de datos”, del grado “Ciencia de Datos” de la Universidad de Valencia, ha incrementado notablemente mi conocimiento e interés por esta área de investigación. Los proyectos de investigación de esta área en los que he participado en los últimos años son:

- **SILKNOW** (Pagán, Salvatella et al. 2020) es un Proyecto de investigación que persigue mejorar la comprensión, conservación y diseminación del patrimonio cultural europeo de

la seda de los siglos XV al XIX. Aplica los últimos avances tecnológicos para cumplir este objetivo y a su vez preservar este patrimonio y haciéndolo accesible a varios colectivos (museos, educación, turismo, industria creativa, medios, etc.). Partiendo de los registros de catálogos existentes, reproduce digitalmente antiguas técnicas de tejido con la herramienta Virtual Loom, utiliza avanzadas técnicas de visualización espacio temporal, además de mantener y ofrecer la información desde un grafo de conocimiento. Este proyecto ha sido financiado por el programa European Union's Horizon 2020 de investigación e innovación con el identificador 769504.

- **SEMAP** (U.Valencia 2021) , proyecto que pretende ofrecer una difusión innovadora de los bienes muebles custodiados en museos, enlazándolos semánticamente y a través de mapas espacio-temporales. De esta forma, los usuarios podrán acceder más intuitivamente tanto a los objetos culturales como a las relaciones significativas entre ellos. De este modo, permitirá poner a España en la vanguardia europea de acceso a las colecciones digitales, que no sólo se enfoca en los usuarios tradicionales de este tipo de herramientas, sino que va más allá y aborda las exigencias del sector educativo, turístico, TIC y del ciudadano en general. Los objetivos de SeMap son:

- Dotar de semántica a objetos de patrimonio digitalizados, para permitir la vinculación entre diferentes colecciones.
- Hacer más accesibles, de una manera innovadora, los objetos de patrimonio digitalizados.
- Difundir y mantener vivo nuestro patrimonio a través de la web.

Este proyecto ha sido financiado por la Fundación BBVA bajo la convocatoria “Ayudas a Equipos de Investigación Científica”, en la disciplina de “Humanidades Digitales”, del año 2019

- **AVD** (Arxiu Valencià de Disseny) (PLA-VIVAS 2021). Proyecto de investigación cuyo objetivo es la creación de un grafo de conocimiento con los datos del Arxiu Valencià de Disseny, un buscador semántico, que permita acceder a la información del grafo de conocimiento y el desarrollo de la herramienta Mapa de Producto. El mapa de producto ha de poder comparar diferentes cualidades de los objetos del grafo de conocimiento de forma gráfica en dos y tres dimensiones.

Una gran parte de los trabajos realizados en esta tesis han sido realizados dentro del proyecto SILKNOW. Es el proyecto más grande y el que comenzó primero, sobre todo en el área de visualización de datos espacio temporales. Los proyectos SEMAP y AVD, son más pequeños, pero han sido desarrollados por completo por IRTIC, siendo nuestra responsabilidad y tarea la visualización de datos espacio temporales, en los que he participado activamente, además, también lo hice en la creación del grafo de conocimiento, importación y acceso a su contenido mediante servicios web.

Mi participación en estos tres proyectos, ha sido fundamental para motivar el interés en estas áreas de investigación, en las que he tenido la suerte de participar y aprender de equipos internacionales y multidisciplinarios. Todo esto me ha hecho comprender la importancia del objetivo y profundizar en la investigación y desarrollo de estos campos, utilizando los últimos avances tecnológicos y científicos.

Todas estas actividades y trabajos realizados, anteriormente expuestos, me ha motivado a la realización de esta tesis.

### 1.3. PROBLEMAS EN GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL.

#### 1.3.1. Problema 1. Falta de uniformidad.

Muchos de los datos de patrimonio cultural se encuentran almacenados en diferentes soportes tecnológicos. Desde sistemas complejos, como pueda ser la red de información generada en el proyecto ARIADNE (Meghini, Niccolucci et al. 2017) , el portal EUROPEANA (EUROPEANA 2021) , o el portal de Google Arts & Culture (Google 2021a). Muchos de los museos más famosos publican y ofrecen sus colecciones a través de portales web mediante un API (Louvre 2021, Museo del Prado. 2021, Victorian Albert Museum 2021). Las instituciones gubernamentales también realizan acciones para unificar y digitalizar los contenidos de los museos, como es el caso del portal CERES (Ministerio de Cultura y Deporte. 2021). A pesar de este esfuerzo hay muchas colecciones, soportadas por pequeñas organizaciones, o con problemas económicos, que no tienen sus contenidos digitalizados, o lo tienen en sistemas cerrados (Samaroudi, Echavarria et al. 2020) . La información sobre objetos de patrimonio cultural es compleja, además de datos multimedia, hay que tener en cuenta el autor, o autores, cuando y donde se realizó, dimensiones, materiales y técnicas utilizadas, sus diferentes propietarios, lugares en los que ha estado, modificaciones, y más información. Estos datos tienen que estar debidamente clasificados y seguir un formato para poder ser procesados. Esto es muy difícil sin un modelo de referencia único. En las dos últimas décadas se han definidos varios modelos de referencia y ontologías para representar este tipo de datos (Nafis, Yahyaouy et al. 2019), pero bien porque ha sido reconocido como estándar ISO (21127:2014), o porque lleva más de veinte años activo y en continuo refinamiento, ha sido el modelo CIDOC-CRM (Bekiari, C., Bruseker, G., Doerr, M., Ore, C., Stead, S., Velios, A. ) el que más usado en proyectos de investigación, como ARIADNE, HERACLES (Hellmund, Hertweck et al. 2018), o SILKNOW (Portalés, Sebastián et al. 2018), y también por instituciones y museos, como el Museo del Prado (Pantoja, J., Docampo, J., Martín, A., Maturana, R.A., Navalón, C. 6/04/2016)

A pesar de que CIDOC-CRM es el modelo más usado, debido a la complejidad y abstracción del mismo han aparecido varias extensiones del mismo, con el fin de adaptarse más a la realidad de los datos que se manejaban, como en el proyecto HERACLES, o la ontología ArchOnto (Koch, Ribeiro et al. 2020) Por estas mismas razones se publican nuevos modelos como alternativa a CIDOC-CRM, como es el caso del modelo CHARM (González-Pérez 2018) .

Tanto los trabajos que usan CIDOC-CRM, como sus extensiones y otros modelos se apoyan en tesauros para indicar materiales, técnicas, tipos de objeto, etc., ( UNESCO (UNESCO 1977), Getty (Getty 1997a), FISH (FISH 2003), SILKNOW (León, Gaitán et al. 2020)) además de usar diferentes fuentes de datos espaciales (Google Maps (Google 2021c), Geonames (Wick 2021) , o temporales (Getty 1997b). En los últimos años se han creado nuevos tesauros que permiten su acceso de forma digital, y otros, inicialmente editados en papel, han sido migrados a plataformas digitales. La existencia de estos tesauros facilita la clasificación de los datos, sin embargo, añaden el problema de que hay una gran variedad de tesauros con contenidos similares, y muchos de ellos no están interconectados (Isaac, Zinn et al. 2007).

Resumiendo, en la actualidad, muchos datos se encuentran en plataformas tecnológicas, pero muchas de ellas no están debidamente clasificadas en un único modelo. Muchas de las que están clasificadas en un modelo como CIDOC-CRM, hacen uso de diferentes tesauros para apoyar la clasificación. Además, hay mucha información, sobre todo procedente de museos pequeños, o de empresas artesanales, que no tienen su información digitalizada, o la tienen en sistemas cerrados. Aunque se ha avanzado mucho en la última década, aún queda mucho trabajo por hacer para poder afirmar que los datos de patrimonio cultural tienen uniformidad.

### *1.3.2. Problema 2. Incerteza*

Los datos de patrimonio cultural poseen una dificultad extra, falta de información o incerteza. Bien por motivos técnicos, o bien por falta de documentación, suele ser frecuente el desconocimiento de parte de información que se desconoce, o que presente incerteza. Esto sucede con información importante, como los diferentes propietarios que ha tenido a lo largo del tiempo, o el autor, hasta información fundamental como el lugar, o el tiempo en que se produjo.

Por ejemplo, en este enlace de la textiloteca del Museo Tèxtil de Terrasa (IMATEX 2021), se observa que hay un tejido cuya procedencia no está clara y se le atribuyen orígenes tan distantes como Francia, Países Bajos y China. Este hecho no es que sea la norma en los datos de patrimonio cultural, pero tampoco es precisamente extraño y sucede con relativa frecuencia.

Clasificar y representar información que se desconoce es sencillo, simplemente, ni se clasifica ni se representa, ya que es desconocida. El problema surge cuando se tienen diferentes posibilidades sobre un dato. Esto es un problema, porque modelos como CIDOC-CRM no ofrecen opciones claras sobre como clasificar posibles interpretaciones de un conocimiento y esto implica que se hayan desarrollado extensiones del modelo, no oficiales y adaptadas a las necesidades surgidas en diferentes desarrollos (Cantone, D. and Cristofaro, S. and Nicolosi-Asmundo, M. and Prado, F. and Santamaria, D.F. and Spampinato, D. 2019, Van Ruymbeke, Hallot et al. 2018). Este hecho, también contribuye a que haya menos uniformidad en el conocimiento sobre patrimonio cultural, que además genera el problema de la representación.

La representación de esta incerteza, para que el usuario conozca su existencia, así como las diferentes interpretaciones también es un reto, que actualmente aún está en desarrollo, y sobre el que se han realizado varios trabajos (Windhager, F. and Filipov, V. A. and Salisu, S. and Mayr, E. June 2018, Martin-Rodilla, P. & Gonzalez-Perez, C. October, 2018)

Relacionado con este problema, y que afecta especialmente a su representación gráfica, aparece el tema de la granularidad espacial y temporal (McKeague, P., van't Veer, R., Huvila, I., Moreau, A., Verhagen, P., Bernard, L., Cooper, A., Green, C. and van Manen, N. 2019). El problema se da cuando se trabaja con objetos con diferente nivel de precisión geográfica. Por ejemplo, se pueden encontrar objetos relacionados con un país, y también objetos relacionados con una ciudad, o con unas coordenadas espaciales concretas. Exceptuando aplicaciones muy concretas, conocer la ciudad es más que suficiente, pero cuando existen varios datos donde sólo se conoce el país, es complejo decidir dónde y cómo representarlos y distinguirlos de aquellos que están mejor geolocalizados. Afortunadamente, dado el número de aplicaciones de este tipo de datos, las instituciones cada vez están participando de forma más activa en la definición de estos datos y cómo compartirlo. Prueba de ellos es la iniciativa INSPIRE (European Commission 2019).

Aunque se ha comentado el problema de la incerteza sobre en temas de localización espacial, este mismo problema sucede con la escala temporal. Tanto el espacio, como el tiempo, son datos ligados a la procedencia del objeto, que suele ser la propiedad del objeto con menos conocimiento exacto. La procedencia del objeto hace referencia al origen del objeto, anterior a su ubicación actual. Esta propiedad, puede ser su lugar y momento de producción, o no, además, puede constar de varios puntos espaciales y temporales, a lo largo de la historia. Dado su nivel de incerteza, su clasificación y representación es un tema complejo, pero debido a su importancia, se está impulsando cada vez más su inclusión en los modelos del conocimiento (Padfield, Kontiza et al. 2019) para que puedan soportar información sobre todas las posibles interpretaciones.

*1.3.3. Problema 3. Ausencia de herramientas abiertas de visualización interactivas de datos de grafos de conocimiento.*

Las colecciones de objetos de patrimonio cultural que están cada vez son más accesibles mediante interfaces web, que están asumiendo un rol similar al de los museos (Cloonan 2014). Y dado el variado perfil de los usuarios que puede visualizar esta información (científicos, estudiantes, turistas, etc.) hace muy necesario que técnicas de visualización avanzada se incorporen a estos interfaces (European Commission 2021) .

En la actualidad existen numerosas herramientas para ver estos datos. La mayoría son ad hoc, no son abiertas, ni en su totalidad ni en parte, y están diseñadas para integrarse en una aplicación determinada, o forman parte intrínseca de ella. Además de que, al estar diseñadas expresamente para un proyecto, no se suele establecer ninguna relación genérica con el dominio de datos, que permite su uso con otras fuentes de información.

Sin embargo, estas herramientas tienen una serie de elementos en común que es interesante analizar. La forma en la que han diseñado estos elementos, principalmente con innovadores interfaces de usuario, es lo que las diferencia. Los principales elementos a considerar son:

- Representación geoespacial. Comprende las diferentes técnicas para georreferenciar al objeto. Lo más común es utilizar un mapa (Zerbini 2018, National Cultural Fund of Hungary ), pero en otras representaciones conocer el lugar exacto no es tan importante. En estos casos se suele representar su pertenencia a una región sin importar su localización exacta, como es el caso del sistema de visualización del Museo Británico (British Museum. ) , donde sólo importa el continente al que pertenecen.
- Capas utilizadas para mostrar la zona espacial. Esto es usado cuando la información se representa mediante un mapa. Generalmente son mapas, con mayor o menor resolución, procedente de repositorios online, como pueda ser OpenTreeMap (AZAVEA 2021), o Google Maps. También hay otros sistemas que usan mapas antiguos escaneados, combinados o no, con los tradicionales (Helsingin kaupunginmuseo 2015, National Cultural Fund of Hungary ) .
- Selección y visualización de datos en función de variables temporales. Este elemento es otro de los elementos diferenciadores clave. En algunas herramientas se ha desarrollado la llamada línea de tiempo, que permite acotar un intervalo de tiempo (Helsingin kaupunginmuseo 2015). Esto es lo más habitual, junto con un filtro simple. Otras soluciones, optan por estrategias más elaboradas, como estructurar el tiempo en capas, con estructuras de Treemap (Wang, C., Ma, and Chen 2018a) , o con círculos concéntricos(Zerbini 2018). Existen sistemas que muestran varias etapas de tiempo de forma simultánea, permitiendo ver de forma más sencilla la evolución temporal de los datos. Estos sistemas utilizan técnicas basadas en hiper cubos(Bach, B., Dragicevic et al. 2017), pero hay otras más innovadoras (British Museum. , Aga Khan Documentation Center at MIT 2019).
- Algoritmos y técnicas de visualización de agrupamiento de datos, o clustering. Generalmente hay muchos datos que visualizar y es necesario agruparlos para poder manejarlos de forma adecuada, además de para permitir una renderización óptima. Lo más habitual es usar burbujas, o círculos, para representar una agrupación. Además de esta técnica, que utiliza varios algoritmos (Ester M, Kriegel HP, Sander J, Xu X 1996, Agafonkin 2016), se usan colores, tamaños, animaciones y agrupaciones geométricas, o de otro tipo, que agrupan varios grupos, o clusters.
- Visualización de los objetos y agrupaciones de objetos. En un determinado momento de la navegación, el sistema ha de poder mostrar información sobre un objeto, o sobre un grupo de ellos. Este proceso cambia mucho en función de las herramientas, desde simples iconos,

hasta modelos 3D, para los objetos individuales, hasta galerías virtuales, listados, o composiciones fotográficas, para un grupo de objetos.

- Relaciones entre los objetos. El cálculo de relaciones entre los objetos suele implicar un coste, que además requiere de elaboradas interfaces para poder analizar esta información de una forma usable. Seguramente estos motivos son la causa de que no suele ser habitual representar estas relaciones, fuera de aplicaciones de visualización de grafos, o de ontologías (Museu Picasso 2020). Generalmente, para analizar los objetos que están relacionados se usan filtros, con el inconveniente de que se visualizan los objetos que verifican todos los valores de los filtros y no se representa la relación entre los que tienen uno, o varios valores iguales. Por ejemplo, generalmente se representan objetos de un mismo país, o que usan un material, pero entre los datos filtrados no se suele representar cuales usan la misma técnica de elaboración que un objeto dado.
- Uso de tecnologías de representación de gráficos tridimensionales. El uso de representaciones tridimensionales para representar información espacio temporal está presente en varias soluciones, aunque no suele ser lo más habitual. Sin embargo, cuesta encontrar soluciones que realmente obtengan un beneficio importante de esta tercera dimensión (Claus O. Wilke 2019). La visualización de datos en entornos tridimensionales es un área de investigación que necesita un desarrollo importante, sobre todo debido al abaratamiento de nuevos dispositivos de realidad virtual que cada vez tienen una calidad más elevada.
- La interacción debe de estar presente en la mayoría de los procesos del sistema de visualización. Por una parte, debe de controlar cómo navegar por la escena, independientemente de las dimensiones de la misma, por otra parte, seleccionar los diferentes objetos, y permitir la interacción, filtrado y visualización de relaciones. Además de forma transversal tiene que considerarse la visualización de la dimensión temporal de los datos y ofrecer diversas formas de visualizar la información.

Existen numerosas herramientas diseñadas para visualizar datos espacio temporales de patrimonio cultural, algunos de ellos toman como fuente de datos grafos de conocimiento, otros usan bases de datos, etc. Muchos de los elementos anteriormente expuestos, que son necesarios para construir un visualizador de estas características se encuentran disponibles de forma abierta, pero las herramientas desarrolladas que aúnen estos elementos suelen ser cerradas y creadas para un proyecto concreto.

Además, hay elementos, como la visualización de relaciones y el uso de tres dimensiones que tiene que mejorar.

#### 1.4. PREGUNTAS DE INVESTIGACIÓN

Por lo expuesto hasta ahora, surgen una serie de preguntas, que requieren la realización de una investigación exhaustiva, para obtener, o al menos avanzar en el conocimiento de las posibles respuestas:

- **¿Se puede considerar CIDOC-CRM como un modelo “estándar” para representar datos de patrimonio cultural?** Esto implicará analizar los CIDOC-CRM, junto a los diferentes modelos utilizados hasta la fecha, así como su utilización en los diferentes proyectos desarrollados por la comunidad científica.
- **¿Existe algún sistema para definir de forma semántica como visualizar los elementos de un grafo de conocimiento?** Para responder a esta pregunta, habrá que revisar los diferentes proyectos de visualización de grafos de conocimiento, profundizando especialmente en aquellos que usan una ontología para representar cómo visualizar la información.



- **¿Se podría crear un sistema de visualización interactivo de datos de patrimonio cultural tomando como ejes principales, el espacio y el tiempo, pero además enlazando esta dimensión con otras variables de los datos?** Dentro de esta investigación se propondrá un diseño que cumpla estos requisitos.
- **¿Se podría aplicar un sistema de estas características en una plataforma web independiente del dispositivo?** Parte de los objetivos de este trabajo consisten en realizar estos desarrollos, así como evaluar los resultados obtenidos. Esto implicará analizar las posibilidades tecnológicas actuales, que permiten visualizar datos en dos y tres dimensiones a través de una aplicación web, así como las diferentes plataformas que soportan un grafo de conocimiento y sus posibilidades de conexión para acceder a los datos desde una aplicación remota.

## 1.5. ORGANIZACIÓN DE LA MEMORIA

Con el fin de responder a las cuestiones anteriores, la memoria ha sido estructurada en cinco secciones principales:

En el siguiente capítulo de **la presente sección**, se enumeran los principales objetivos a alcanzar. Para ello, se expone el objetivo general y los objetivos específicos que lo componen. Para final el capítulo se detalla la metodología de investigación utilizada en la realización de esta tesis.

En la **sección segunda** se describe el marco teórico, así como las tecnologías relacionadas y el estado del arte.

En el capítulo 3 se analiza el concepto de grafo de conocimiento, analizando el concepto y su evolución. Se describen los diferentes modelos de referencia para representar datos de patrimonio cultural. También se enumeran los diferentes tesauros usados en esta área, haciendo más hincapié en lo que definen propiedades más generales y comunes a este tipo de datos, como son el espacio y el tiempo.

El capítulo 4 expone el estado del arte en la visualización de datos. Parte de una visión general, para ir centrando el contenido entorno a la visualización de datos espacio temporales y datos de patrimonio cultural. Se describen las diferentes herramientas usadas, así como diferentes elementos que las componen. En el capítulo también se analizan los diferentes problemas existentes en este campo y las soluciones desarrolladas. El contenido también comprende los diferentes estudios realizados con el objetivo de visualizar la información de una ontología, comenzando con una visión amplia, para enfocar el discurso sobre el uso de ontologías que describan cómo visualizar la información presente en una ontología. Este es un campo de investigación en el que no se ha avanzado demasiado, pero sí que hay resultados, cuyas ventajas e inconvenientes serán expuestas en este trabajo.

El último capítulo de esta sección, el 5, contiene una exposición de análisis de las tecnologías y herramientas relacionadas con la implementación y conectividad de grafos de conocimiento, servicios web, las últimas plataformas de desarrollo de aplicaciones web y visualización de gráficos bidimensionales y tridimensionales en tiempo real. Algunas de estas tecnologías serán usadas para los desarrollos realizados en esta tesis.

En la **sección tercera**, se exponen las principales propuestas y contribuciones.

El capítulo 6 describe el modelo de un sistema de visualización de datos de patrimonio cultural. Se describirán los elementos y las principales consideraciones a tener en cuenta a la hora de definir un sistema de estas características. Además, se exponen principales soluciones



y alternativas propuestas para evitar, o reducir algunos de los problemas de visualización de datos espacio temporales, teniendo en cuenta las particularidades del patrimonio cultural.

El capítulo 7 explica una de las propuestas principales de esta tesis, definir una vía genérica en la que se establezcan los procesos seguidos para determinar cómo visualizar y definir la interacción de los diferentes objetos y clases de un grafo de conocimiento mediante una plataforma de visualización. Para ello se partirá del modelo descrito en el capítulo 6, combinándolo con los avances y conclusiones del capítulo 4, para finalmente diseñar e implementar una ontología que represente un sistema de visualización de datos de patrimonio cultural y su conexión con el grafo de conocimiento que soporta la información.

El capítulo 8 expone un marco de trabajo basado en una aplicación web, que permita la conexión entre las plataformas que soportan la información de un grafo de conocimiento y diferentes tecnologías de representación gráfica. En este marco juega un papel fundamental los resultados del capítulo 7, ya que define cómo se relacionan las dos partes principales de este marco de trabajo: los datos de un grafo de conocimiento y un sistema de visualización de datos de patrimonio cultural.

En la **sección cuarta** se exponen los resultados alcanzados, así como las pruebas efectuadas para validar las contribuciones aportadas.

El capítulo 9 explica dos diferentes implementaciones del marco de trabajo definido en el capítulo 8. Todas las implementaciones tienen información del patrimonio cultural soportada en diferentes grafos de conocimiento, y diferentes formas de visualizar los datos, ya que los análisis y funcionalidades son diferentes.

El capítulo 10 describe las pruebas realizadas sobre las implementaciones expuestas en el capítulo anterior. Las pruebas han sido realizadas por varios tipos de usuarios, y el objetivo ha sido medir la usabilidad y el rendimiento del sistema.

En la **sección quinta** se detallan las conclusiones alcanzadas, temas de discusión, así como recomendaciones para futuras investigaciones.

El capítulo 11 recoge las conclusiones obtenidas a partir de los resultados de esta investigación.

El capítulo 12 explica las futuras líneas de trabajo que se pueden realizar para mejorar y ampliar los resultados de esa tesis doctoral.

Por último, se enumeran las referencias bibliográficas y las publicaciones relacionadas y derivadas de la investigación realizada en esta tesis y anexos.

## CAPÍTULO 2. OBJETIVOS Y METODOLOGÍA

### 2.1. INTRODUCCIÓN

En el capítulo anterior, se han expuesto los problemas existentes en la actualidad con los modelos de datos para soportar datos espacio temporales de patrimonio cultural, y las tecnologías, medios y diseño para su visualización y análisis. También se han identificado cuestiones relativas a posibles soluciones a estos problemas. A continuación, se describen los objetivos más importantes que se han de cubrir, primero se expondrá el objetivo general, que se descompone en una serie de objetivos específicos, explicados en segundo lugar. Por último, se describe la metodología de investigación seguida en la realización de esta tesis.

### 2.2. OBJETIVO GENERAL

Proponer un modelo de integración de datos de patrimonio cultural, que estén soportados por un grafo de conocimiento, para el desarrollo de un sistema de visualización interactivo basado en escenas. El sistema debe de ser capaz de manejar fuentes desde diferentes dominios de datos y permitir la definición del aspecto visual de las diferentes instancias en función de su clase y la escena en la que se representa. Debe de permitir filtrar información de forma interactiva en función de las propiedades y valores de los datos recibidos, además facilitar el análisis de las relaciones entre los datos visualizados, así como su evolución en el espacio y tiempo.

El requisito de mayor importancia que se exige a la información a visualizar es que posea información espacial y temporal. El sistema debe de soportar diferentes posibilidades para visualizar este tipo de información, tanto en dos, como en tres dimensiones, incluso debe permitir definir un marco espacial propio.

### 2.3. OBJETIVOS ESPECÍFICOS

Los objetivos específicos a alcanzar con esta tesis son los siguientes:

- Analizar las ontologías de visualización de información con el fin de elegir la más adecuada para representar los elementos gráficos necesario para visualizar información de datos espacio temporales del patrimonio cultural. Una vez seleccionada la más idónea, habrá que diseñar e implementar las extensiones adecuadas en caso de que sea necesario.
- Diseñar, implementar y evaluar un componente web de visualización de datos espacio temporales de patrimonio cultural. Este componente ha de ser capaz de realizar las funcionalidades de los sistemas actuales, visualizar de varias formas la evolución de la información en diferentes momentos de tiempo y mejorar la visualización de las relaciones entre los diferentes datos. Con estas mejoras se facilitará la detección de patrones entre los datos.
- Exponer un marco de trabajo que permita definir la comunicación entre uno, o varios grafos de conocimiento y el componente de visualización. Además de definir los datos que se van a visualizar, tiene que permitir especificar como se visualizan estos datos mediante la ontología resultante del primer objetivo.
- Presentar y validar una arquitectura que haga posible integrar el componente de visualización en el marco trabajo anteriormente expuesto, con varios grafos de conocimiento con datos espacio temporales de patrimonio cultural. Esto se hará mediante

dos implementaciones sobre distintas plataformas web, que permitan visualizar de varias formas el contenido de diferentes grafos de conocimiento.

## 2.4. METODOLOGÍA DE INVESTIGACIÓN

La investigación realizada en esta tesis ha sido de tipo experimental, en la que se ha contado con la ayuda de un equipo multidisciplinar de primer nivel. Como se comentó en la introducción de esta tesis, los resultados han sido diseñados, desarrollados e implementados en el marco de varios proyectos de investigación. En estos proyectos se ha contado con la colaboración de varios científicos internacionales de áreas tecnológicas y del campo de historia del arte, historia del diseño y del área de geografía. Su conocimiento en estos campos ha sido fundamental para asesorar durante la investigación y también para validar el trabajo realizado desde el punto de vista de usuario final. Durante el proceso de la investigación, se han desarrollado varios bancos de pruebas, usando diferentes herramientas, en los que se han incorporado nuevas ideas de forma constante.

Las fases que se han seguido en la metodología se muestran en la Figura 1 .

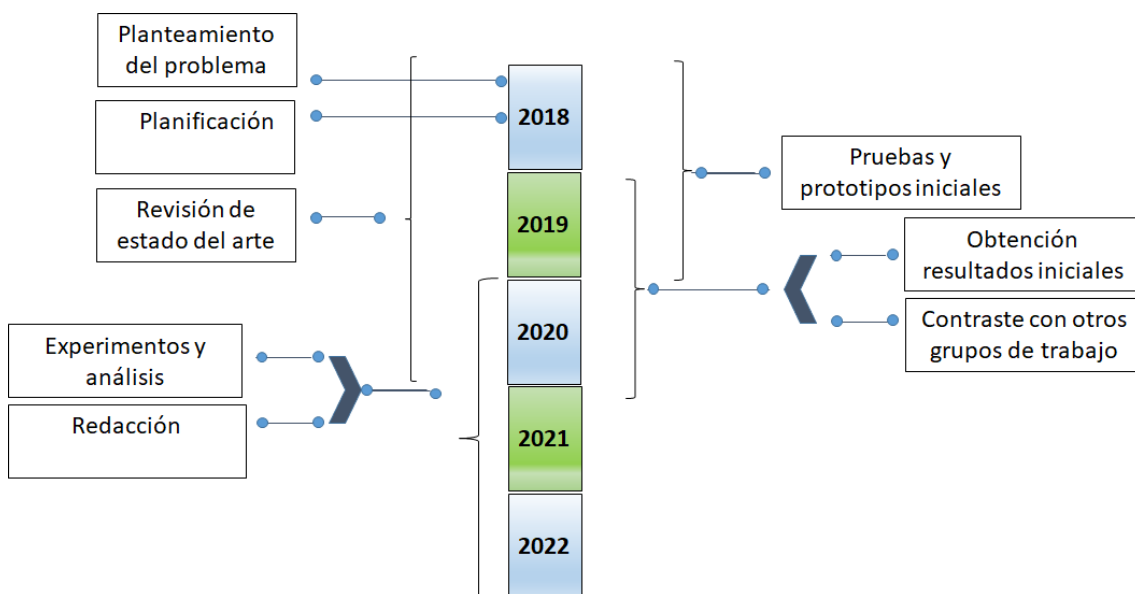


Figura 1 Metodología de la investigación.

Una vez planteado el problema y propuesta su planificación, el trabajo llevado a cabo en la investigación ha tenido una fuerte componente de revisión de todas la tecnologías y avances relacionados con el proceso, principalmente las de visualización de información y grafos de conocimiento, en concreto la de datos espacios temporales basados en plataformas web. En esta etapa ha sido fundamental asentar un proceso de formación del estado del arte mediante la revisión bibliográfica. El estado del arte se ha realizado de forma continua durante el resto de la investigación, y se ha basado en la revisión de todo tipo de fuentes de información, principalmente libros, revistas científicas especializadas, sitios web de programación web, sitios web de congresos especializados, recomendaciones W3C, asistiendo a congresos internacionales, como International Conference on Computational Science (ICCS) en Junio de 2019, asistiendo a webinars y conferencias web organizadas por museos, o entidades gubernamentales, relacionadas con la visualización de su contenido como “Museo del Prado y la Línea del tiempo”(British Museum, Google Cultural Institute 2015, Museo del Prado. 2021). Además de aprender mucho del resto de partners de los proyectos de investigación en los que he trabajado durante la investigación, también he participado en reuniones con otros

organizaciones internacionales, como ICCROM (ICCROM 2021), HOCHSCHULE LUZERN (Lucerne 2021) y equipos de proyectos europeos del mismo ámbito (Mingei 2020).

La investigación fue llevada a cabo dentro del grupo de investigación ARTEC del Instituto de Robótica y Tecnologías de la Información. Varias publicaciones han sido desarrolladas como parte de los resultados de este trabajo.

Debido a que gran parte del trabajo se desarrolló mediante tecnologías y estudios de reciente creación, aparecieron varios problemas, tanto tecnológicos como de interpretación de resultados de algunos autores. En ocasiones aparecieron problemas derivados con las dependencias de terceros, no siempre disponibles en todos los sistemas operativos, otras con la estabilidad de las herramientas. Esta situación, además de conllevar retrasos por problemas técnicos, también provocó retrasos debidos a falsos resultados, que con el avance del trabajo se demostró que no eran tan buenos, dando lugar a retrocesos y obligando a volver a comenzar desde ese punto. Estos problemas fueron resueltos por recursos propios y por la ayuda de miembros del grupo de trabajo, además de la colaboración de algunos de los autores, que además de responder y atender por correo electrónico, estuvieron disponibles a atenderme por videoconferencia, como es el caso de Jan Polowinski, creador de la ontología VISO (Polowinski, Jan, Voigt 2013).

Se establecieron varias pruebas y revisiones de los resultados del proyecto, ya que el sistema necesitaba el cumplimiento y verificación de objetivos, tanto funcionales, como de comunicación, integración y de usabilidad. Tras cumplir estas validaciones y realizar los análisis pertinentes, revisando las alternativas encontradas, se definió una propuesta que se ha ido mejorando sucesivamente.

Debido a la evolución y cambio constante de las tecnologías y el trabajo realizado, una de las estrategias más acertadas para avanzar en la investigación ha sido la consulta y participación en foros de discusión. Tanto para resolver problemas tecnológicos, como para avanzar en el trabajo en direcciones comúnmente aceptadas, ha sido fundamental el apoyo y la colaboración con la comunidad científica a través de este medio. Algunos de los foros de discusión más consultados han sido:

- [crm-sig@ics.forth.gr](mailto:crm-sig@ics.forth.gr): Lista de correo de CIDOC-ICOM (International Committee for Documentation)
- <https://community.openlinksw.com/> (Foro de discusión de OpenLink Software Community)
- <https://www.nodeslackers.com/> (Foro de discusión Slack sobre desarrolladores Node)
- <https://forum.vuejs.org/> (Foro de discusión de la Comunidad Vue.js)
- <https://forum.unity.com/> (Foros de Unity)

## SECCIÓN II. MARCO TEÓRICO. TECNOLOGÍA Y ESTADO DEL ARTE.

## CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

### 3.1. RESUMEN

Aunque la comunidad científica no termina de ponerse de acuerdo en una definición del concepto de grafo de conocimiento, en la última década se han diseñado e implementado varias herramientas, así como varios estudios y trabajos relacionados con este concepto.

Su aplicación se extiende a varios campos, desde medicina, estructuración del lenguaje y patrimonio cultural.

En este capítulo se analiza el concepto de grafo de conocimiento, así como los últimos avances tecnológicos realizados. Se exponen sus diferentes aplicaciones, haciendo especial hincapié en el área de preservación del patrimonio cultural. Para hacer realidad estos proyectos es necesario contar con un modelo de referencia, además de varios vocabularios especializados. En el capítulo también se estudian los diferentes modelos de referencia en esta área, junto con los vocabularios más utilizados, especialmente los relacionados con el espacio y el tiempo. Además, a modo de referencia y consulta, se citan varios proyectos de patrimonio cultural que usan grafos de conocimiento.

### 3.2. CONCEPTO DE GRAFO DE CONOCIMIENTO

Aunque la idea de representar el conocimiento mediante un grafo es antigua, y en 1956 se define como tal, el concepto de red semántica (Richens 1956). Es en la última década cuando el concepto de grafo de conocimiento se ha utilizado cada vez con más frecuencia, tanto en el ámbito de investigación como en el de negocio, generalmente relacionado con tecnologías de la web semántica (Ehrlinger, L., & Wöß, W. 2016). La introducción del grafo de conocimiento de Google, en 2012 (Singhal 2012), fue un factor determinante para que el término fuera más popular.

A pesar de su popularidad, la comunidad científica no se pone de acuerdo a la hora de definir lo que es un grafo de conocimiento.

En algunos trabajos se define el concepto de grafo de conocimiento como una representación del conocimiento basada en un grafo. Un ejemplo de ello lo tenemos en las siguientes definiciones:

“Un grafo de conocimiento principalmente describe las entidades del mundo real y sus relaciones, organizadas en un grafo, define las posibles clases y relaciones de las entidades mediante un esquema, permitiendo una interrelación potencialmente arbitraria entre las entidades, cubriendo varios dominios.” (Paulheim 2017)

“Un grafo de conocimiento es un grafo multirelacional compuesto de entidades y relaciones, los cuales se representan respectivamente como nodos con diferentes tipos de aristas” (Wang, Q., Mao et al. 2017).

“Un grafo de conocimiento es un grafo  $G = \{E, R, F\}$ , donde  $E$ ,  $R$  y  $F$  son respectivamente conjuntos de entidades, relaciones y hechos. Un hecho se denota como una tripleta  $(h, r, t) \in F$ .” (Ji, Pan et al. 2021)

Analizando estas definiciones se observa que hacen referencia a los diferentes elementos que forman el conocimiento, sus relaciones y cómo se representan. El problema de estas

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

definiciones es que no ayuda a diferenciarlo del concepto base de conocimiento, que a menudo es utilizado como sinónimo de ontología, generando una confusión importante (Ehrlinger, L., & Wöß, W. 2016).

Una de las definiciones más aceptadas de ontología es: “Una ontología es una fuerte estructura semántica que codifica reglas implícitas restringiendo la estructura de una porción de la realidad” (Guarino 1998).

Esta definición se complementa con el hecho de que ha de incluir un vocabulario de términos, que contiene definiciones e indica cómo los conceptos están relacionados entre sí. Imponiendo una estructura sobre el dominio y restringiendo posibles interpretaciones de los términos (Uschold, King et al. 1998, Natalya F. Noy and Deborah L. McGuinness. 2001).

Atendiendo a estas definiciones, es comprensible que se identifique el concepto de ontología con el de base de conocimiento, ya que no hay diferencias entre ellos, una ontología es una base de conocimiento. Sin embargo, esto no se puede extrapolar a grafo de conocimiento, ya que la mayoría de trabajos se esfuerzan en marcar diferencias entre estos conceptos.

Existen publicaciones, que indican que la principal diferencia está relacionada con la dimensión, donde un grafo de conocimiento sería una base de conocimiento de grandes dimensiones (Blumauer 2016).

En otros trabajos, se asocia el concepto de grafo de conocimiento al de una base de conocimiento junto a un sistema de razonamiento, definiendo el concepto de grafo de conocimiento como sigue: “Un grafo de conocimiento adquiere a íntegra información en una ontología y aplica un sistema de razonamiento para obtener nuevo conocimiento.” (Ehrlinger, L., & Wöß, W. 2016)

Después de analizar las diferentes posibilidades que la comunidad científica considera a la hora de definir que es un grafo de conocimiento se concluye que hay una cierta confusión a la hora de definirlo. Generalmente los autores suelen considerarlo como algo superior a una ontología, aunque en lo que no hay discusión es que se trata de una base de conocimiento, u ontología, compuesta por el esquema y por sus datos. Para diferenciarlo de una base de conocimiento, algunos autores definen el grafo de conocimiento vinculándolo a un conjunto de datos grande y/o al uso de un sistema de razonamiento. Esto probablemente es debido a que el concepto ha obtenido su popularidad en la última década, donde la mayoría de proyectos tecnológicos en los que se usan grafos de conocimiento manejan una gran cantidad de datos y suelen requerir de un sistema de razonamiento para obtener resultados.

Algunos de los grafos de conocimiento más populares son (Chen, X., Jia et al. 2020):

- Freebase (Bollacker, Evans et al. 2008), es una base de conocimiento colaborativa compuesta por datos generados por su comunidad. Además, tiene datos de Wikipedia, NNDB, Fashion Model Directory y MusicBrainz. Posee cerca de 3.000 millones de hechos. Es una de las fuentes de Google Knowledge Graph.
- DBPedia (Lehmann, Isele et al. 2015), el objetivo de este proyecto es estructurar información procedente del proyecto Wikipedia, considerando todas las lenguas con las que trabaja este proyecto. Tiene alrededor de 50 millones de interrelaciones, muchas de ellas con conjuntos de datos externos, por ejemplo, Freebase. Usa el formato RDF (Resource Description Framework) (W3C 2004b) para representar la información. Maneja todo tipo de entidades, desde personas, hasta videojuegos, películas y desastres naturales.

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

- YAGO (Suchanek, Kasneci et al. 2007), es una base de conocimiento de código abierto desarrollada por el Instituto Max Planck. Extrae información de Wikipedia, WordNet (Fellbaum 1998) y GeoNames. Combina la riqueza de Wikipedia con la taxonomía bien definida de WordNet. Muchos de sus hechos y relaciones tienen la dimensión temporal y espacial.
- Wikidata (Pellissier Tanon, Vrandečić et al. 2016), es una base de conocimiento abierta, enlazada y estructurada. Mantiene información de más de 280 lenguas de Wikipedia. Posee el mismo mecanismo de crowdsourcing de Wikipedia.
- NELL (Never-Ending Language Learner) (Mitchell, Cohen et al. 2018) es un sistema automatizado de aprendizaje semántico que está siempre activo. Creado en la Universidad de Carnegie Mellon. Partiendo de una ontología base, formada por cientos de categorías, y de 10 a 15 ejemplos semilla por categoría, obtiene la información de la web. Posee más de 120 millones de hechos candidatos con un nivel de confianza asignado.
- OpenCyc (Matuszek, C., Cabral, J., Witbrock, M., DeOliveira, J. 2006) es una versión abierta de la base de conocimiento Cyc, que se remonta a la década de los 80. Fue creado en 2002 por la empresa Cycorp. Llegó a tener más de 2 millones de hechos. Sin embargo, en 2017 dejó de estar disponible de forma abierta.
- Grafo de conocimiento de Google (Singhal 2012) , fue publicado en 2012 y es el más popular debido a que aparece en el popular buscador cada vez que se realiza una búsqueda. Tiene más de 500 millones de instancias.
- Knowledge Vault (Dong, Xin, Gabrilovich et al. 2014), es un grafo de conocimiento creado por Google con el objetivo de acceder al contenido de la web para aprender e incorporar nuevos datos. Usa análisis de texto, anotaciones humanas, junto con la estructura de las páginas, para acceder a la información de la web. Esta información es combinada con varias fuentes de datos base, mediante técnicas de inteligencia artificial supervisada.

El volumen de datos que tienen estos grafos de conocimiento se muestra en la Tabla 1

*Tabla 1 Número de instancias, Hechos y tipos de Entidades y Relaciones de los grafos de conocimiento más populares. (Paulheim 2017)*

| Nombre          | Instancias  | Hechos        | Entidades (tipos) | Relaciones (tipos) |
|-----------------|-------------|---------------|-------------------|--------------------|
| DBpedia         | 4,580.000   | 583,000,000   | 685               | 2,795              |
| YAGO            | 4,595,906   | 25,946,870    | 488,469           | 77                 |
| Freebase        | 47,560,817  | 2,903,361,537 | 26,507            | 37,781             |
| Wikidata        | 15,759,256  | 43,189,154    | 23,157            | 1,203              |
| NELL            | 1,908,694   | 441,807       | 274               | 296                |
| OpenCyc         | 118,499     | 2,413,894     | 1,500             | 35,000             |
| Google K. Graph | 500,000,000 | 3,500,000,000 | 1,500             | 35,000             |
| Knowledge Vault | 45,000,000  | 271,000,000   | 1,100             | 4,469              |



### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

#### *3.2.1. Lenguajes relacionados con grafos de conocimiento*

Existen tres tipos de lenguajes relacionados con los grafos de conocimiento. Por una parte, los lenguajes de definición y/o intercambio de ontologías, que se utilizan tanto para definir los conceptos, relaciones y hechos del grafo de conocimiento. Por otra parte, están los lenguajes de reglas, que permiten definir las reglas semánticas, usadas por los razonadores para inferir sobre el conocimiento. En último lugar, se encuentran los lenguajes que permiten realizar consultas sobre los datos y el esquema del conocimiento.

##### *Especificación de ontologías*

Los lenguajes de especificación de ontologías son lenguajes que fueron diseñados para ser usados por tecnologías de la Web Semántica. El objetivo era que con estos lenguajes se especificará conocimiento, mediante anotaciones en páginas web, para que aplicaciones externas, o agentes, fueran capaces de procesarlos y comprender el contenido de la página web en la que se encontraban (Maniraj, Sivakumar 2010, Samper Zapater 2005).

Estos lenguajes están basados en el metalenguaje XML (Extensible Markup Language), donde la última edición de la versión 1.1, fue publicada como recomendación del consorcio W3C (World Wide Web Consortium) el 26 de Noviembre de 2008 (W3C 2006).

Este lenguaje se caracteriza por su flexibilidad. Permite definir nuevas etiquetas, así como la estructura que ha de seguir un documento en lenguaje XML, mediante un archivo en formato DTD (Document Type Definition) (W3C 1998), o mediante el formato XSD (XML Schema Document) (W3C 2012)

XML fue diseñado para el intercambio de datos, por lo que tiene problemas de diseño que hace que no sea adecuado para la representación del conocimiento:

- Cuando un esquema XML cambia, consultas anteriores basadas en la anterior estructura, pueden invalidarse.
- El orden en un documento XML es importante, y en ocasiones es un requisito necesario. Esto es no tiene sentido en los metadatos, los datos que son representados en una ontología no tienen porqué presentarse siguiendo un orden. Aunque mediante relaciones, o a través de los valores de sus atributos, pueda establecerse un orden, pero no que sea obligatorio en la especificación de los datos.

A pesar de estos problemas, debido a que este lenguaje es el pilar en el que se basan los lenguajes relacionados con las tecnologías de la Web, fue la base para definir los lenguajes, cuyas características se exponen a continuación, que sí que se diseñaron para especificar ontologías.

**XOL (XML-Based Ontology Exchange Language)** es un lenguaje diseñado por la comunidad de bioinformática de E.E.U.U., para el intercambio de definición de ontologías entre aplicaciones (Karp, Chaudhri et al. 2000). Su sintaxis está basada en OKBC-Lite, pero definida en XML, lo que favorece el intercambio de datos en la Web. Este lenguaje, permite representar desde el punto de vista de la lógica las clases y los atributos de una ontología. Sin embargo, las restricciones sintácticas son demasiado restrictivas y pueden ocasionar deficiencias en la capacidad de razonamiento, debido a las limitaciones en la combinación de primitivas, a la hora de definir una clase (Weng, Qiu et al. 2020).

**SHOE (Simple HTML Ontology Extension)** es un lenguaje desarrollado en la Universidad de Maryland, creado inicialmente como una extensión de HTML, aunque posteriormente se adaptó su sintaxis a XML (Luke, Heflin 2000). El lenguaje permitía la definición de una ontología,

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

compuesta por una jerarquía de clases y las relaciones existentes entre ellas, junto a una serie de reglas de inferencia definidas mediante cláusulas de Horn. Una vez definida la ontología, se podría anotar información directamente sobre las páginas web, para describir hechos sobre su contenido y al de otras páginas que se referenciaban desde dicha página. De esta forma, agentes software, o aplicaciones externas, podrían acceder a información semántica sobre el contenido de la página.

El lenguaje **RDF (Resource Description Framework)** fue publicado como recomendación del consorcio W3C (World Wide Web Consortium) el 10 de febrero de 2004 (W3C 2004d, W3C 2004c). Con esta recomendación, se define la estructura y sintaxis que permite describir metadatos, asociando su significado mediante RDFS (Resource Description Framework Schema), diseñado para definir ontologías de dominio específicas (W3C 2004b).

RDF está basado en XML, utilizando sus características para facilitar el intercambio y proceso de metadatos con otras aplicaciones, proporcionando independencia, validación, legibilidad humana y extensibilidad. Además, impone estructuras que permiten expresividad semántica inequívoca (Samper Zapater 2005). Es también un medio que permite la publicación de vocabularios, que pueden ser procesados por máquinas, pero también por humanos, que pueden ser reutilizados, extendidos y adaptados a otros dominios.

El modelo RDF está basado en tres partes:

- Recursos (sujeto), definidos por expresiones RDF identificadas por URIs.
- Propiedades (predicado), que definen aspectos específicos, características, atributos, o relaciones utilizadas para describir un recurso.
- Declaraciones (objeto), que asignan un valor a las propiedades del recurso, este valor puede ser un recurso referenciado por su URI, una cadena de texto, u otra primitiva de datos definido por XML.

Para definir jerarquías de clases de objetos y propiedades, además de especificar restricciones (rango y dominio) se usa el RDF Schema. Por lo tanto, RDF Schema ofrece primitivas para definir el modelo de conocimiento, cercanas a los modelos de frames. Se trata pues, de un modelo semántico de datos, que permite realizar consultas sobre su contenido y no a la estructura de la especificación.

**Ontology Interchange Language (OIL)** es un lenguaje diseñado para especificar e intercambiar ontologías (Horrocks, Fensel et al. 2000). Su sintaxis está basada en OKBC, XOL y RDF(S). OIL, además de permitir definir la ontología y asociar instancias, también soporta axiomas, permitiendo realizar razonamiento sobre este tipo de ontologías. Este lenguaje, está compuesto por las siguientes capas:

- *Core*, que comprende las primitivas de OIL, las cuales son un mapeo directo a las primitivas de RDF(S) (Gomez-Perez, Corcho 2002).
- *Standard*, que hace referencia al modelo completo OIL, compuesto por primitivas adicionales a las definidas en RDF(S).
- *Instance*, que permite definir instancias al modelo definido en la capa anterior.
- *Heavy*, la capa para realizar extensiones sobre OIL.

**DARPA Agent Markup Language (DAML) + OIL.** Debido a las limitaciones existentes a finales de los 90 para definir lenguajes que definan ontologías que permitirán inferir conocimiento, la agencia DARPA (Defense Advance Research Projects Agency) de E.E.U.U., definió el lenguaje

DAML (DARPA Agent Markup Language). Aunque DAML no surgió del consorcio W3C, muchos de sus desarrolladores lo eran, y aunaron esfuerzos con los desarrolladores de OIL, beneficiándose de sus sistemas de clasificación (Samper Zapater 2005), dando como resultado DAML + OIL (Mcguinness, Fikes et al. 2002) . A pesar de que, como es lógico, tiene muchas cosas en común con OIL, se distancia de la línea definida por los frames de OIL y se acerca más a un lenguaje de lógica descriptiva. DAML + OIL es una evolución de RDFs en el que se redefinen muchas de sus descripciones y se crea un conjunto importante de nuevas descripciones para mejorarlo y añadir nuevas propiedades y mecanismos. Estos cambios se hacen con el objetivo de facilitar que las ontologías puedan ser usadas por sistemas de razonamiento para inferir información.

**Ontology Web Language (OWL)** es una extensión de RDF(S), publicado como recomendación por el consorcio W3C (World Wide Web Consortium) en 2004 (W3C 2004a), y posteriormente OWL 2 en 2009 (W3C 20009). Este lenguaje fue diseñado con el objetivo de incorporar la mayoría de funcionalidades como tenía el lenguaje DAML + OIL. De hecho, incluye la posibilidad de definir clases con relaciones de conjunción, disyunción y equivalencia, además de definir restricciones de cardinalidad en propiedades, establecer la transitividad o simetría y el uso de enumeración. OWL tiene tres sub-lenguajes diseñados para ser usados en función del nivel de expresividad que se requiera:

- OWL Lite. La versión más simple, que es recomendada para aquellos usuarios que necesiten principalmente una jerarquía y restricciones sencillas. Por ejemplo, ofrece cardinalidad, pero únicamente valores de 0, o 1. Puede ser usado para migrar de forma sencilla tesauros y otras taxonomías sencillas.
- OWL DL. Es una versión con el vocabulario completo, pero tiene limitaciones orientadas a que los usuarios puedan conservar la computacionalidad de las conclusiones, así como su resolubilidad. Lo consigue con restricciones, como, por ejemplo, la que impide que una clase sea instancia de otra clase, aunque haya una relación de herencia entre ellas.
- OWL Full, es la versión DL, pero sin limitaciones que aseguren la computacionalidad. El problema de esta versión es que tiene un nivel tan avanzado de expresividad, que es poco probable que se diseñe un software racional que soporte por completo el razonamiento para cada característica de OWL Full (Lamarca 2018).

La segunda versión del lenguaje, OWL 2, también generó otros tres sub-lenguajes, orientados a permitir la creación de soluciones, más sencillas y eficientes, a los desarrolladores:

- OWL 2 RL. Un subconjunto de OWL 2 que puede ser implementado usando tecnología basada en reglas. Fue diseñado para que las aplicaciones soportaran la expresividad completa de OWL 2 con mejor eficiencia.
- OWL 2 QL. Es una intersección entre RDFS y OWL 2 DL. Fue diseñado para que los datos, expresados con tripletas, que estuvieran soportados por una base de datos relacional, pudieran ser consultados mediante una ontología, con un mecanismo simple de sobreescritura.
- OWL 2 EL. Esta versión está diseñada para poder gestionar eficientemente modelos de grandes dimensiones, con decenas o cientos de clases, aunque con poca información. Al contrario que las dos anteriores.

**Contexto actual.**

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

Parece obvio, analizando la evolución de los lenguajes, que los mejores lenguajes son DAML + OIL, u OWL. Esta conclusión, es lógica, simplemente por el hecho de que son lenguajes que han ido evolucionando, o son extensión de los anteriores, y han sido diseñados y validados por varios expertos en el área.

Además, si tenemos en cuenta estudios (Pulido, Ruiz et al. 2006, Tonti, Bradshaw et al. 2003, Samper Zapater 2005) que demuestran que RDF(S) tiene varios problemas como:

- No se pueden definir propiedades de propiedades.
- No hay equivalencia, ni disyunción, en la definición de las clases.
- No se pueden imponer restricciones de cardinalidad.
- No se pueden definir condiciones necesarias y suficientes.

Queda claro que en RDF(S) no todos los términos tienen semántica y tiene deficiencias importantes para poder inferir conocimiento.

Dado que OWL estaba basado en RDF(S) y fue publicado como recomendación por el World Wide Web Consortium (W3C), este se estableció como el lenguaje más usado (Kalibatiene, Vasilecas 2011). El problema es que el lenguaje OWL tiene un alto coste computacional debido a su alta complejidad. De hecho, las plataformas software únicamente dan soporte a las versiones más ligeras de OWL (OWL Lite y OWL 2 RL), únicamente la herramienta StarDog implementa OWL DL (Ramis Ferrer, Mohammed et al. 2021, Polikoff 2021).

Debido a que la expresividad del lenguaje OWL es muy rica y supera en creces a RDF(S), es el lenguaje de definición de ontologías prevalente, desde el punto de vista teórico. Aunque debido a la complejidad computacional, y a la falta de soporte de esta expresividad en las plataformas software, en la práctica esta expresividad no es explotada. Por este motivo, cada vez más desarrolladores están estudiando otras alternativas (Simsek, Angele et al. 2019, Oliveira 2020, Labra-Gayo, García-González et al. 2019) y una de las más utilizadas es la combinación entre RDF(S) y el lenguaje de restricciones SHACL (Shapes Constraint Language).

SHACL es una recomendación publicada por el consorcio W3C en 2017 (W3C 2017) para validar ontologías expresadas en RDF (Boneva, I., Labra-Gayo, J.E., Hym, S., Prud'hommeau, E.G., Solbrig, H.R., Staworko, S 2014) Una forma, o shape, de SHACL, define una serie de restricciones que los datos de un grafo de conocimiento deben de cumplir. Hay dos tipos de restricciones, aquellas que se refieren al modelo de datos (por ejemplo, cardinalidad) y las que se aplican a los valores de los datos (por ejemplo, patrones de cadenas de texto). Esto, junto al hecho de que la mayoría de plataformas de gestión de grafos de conocimiento soportan este lenguaje (Polikoff 2021), hace que sea una alternativa muy útil para validar la información de un grafo de conocimiento. Sin embargo, el desarrollo de una forma, o shape, en ontologías que tienen un nivel de conocimiento alto, es una tarea difícil y costosa. Debido a esto han aparecido varias soluciones que asisten en el proceso de creación de shapes (Labra-Gayo, García-González et al. 2019, Spahiu, Blerina and Maurino, Andrea and Palmonari, Matteo 2018) y también que permiten generarlas de forma automatizada (Cimmino, Fernández-Izquierdo et al. 2020).

#### Lenguajes de consulta

Al igual que sucede en las bases de datos, para facilitar el acceso a la información, condicionando los resultados basándose en la existencia de relaciones entre las instancias y valores en determinadas propiedades, es fundamental que existan lenguajes de consulta.

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

Seguidamente se exponen los lenguajes de consultas, primero algunos que prácticamente no se usan, pero fueron los primeros, en cuyos resultados se basaron los más utilizados actualmente que también se detallan.

Este tipo de lenguajes se estructuran en dos grandes bloques, RQL (Karvounarakis, Alexaki et al. 2002) y SPARQL (W3C 2008).

#### **Lenguajes de consultas basados en RQL.**

**RQL (RDF Query Language)** es un lenguaje de consulta definido en 2002, que permite consultar datos y esquemas expresados en RDF. RQL permite la interpretación de datos en RDF a través de uno o varios esquemas RDF. Adapta la funcionalidad de los lenguajes de consulta XML a RDF.

Sigue la sintaxis del lenguaje OQL, un ejemplo de posible consulta sería el siguiente:

```
select Libro from {Libro} ns2:autor {a} where a = "Tolkien" using namespace ns2 = ...
```

RQL no es del todo compatible con las especificaciones semánticas de RDF (Haase, Broekstra et al. 2004) , por ello es necesario incluir varias restricciones en los modelos RDF para poder usarlo. Por ejemplo, cada propiedad de RDF sólo puede tener un dominio y un rango específico.

Dentro de este bloque de lenguajes de consulta se encontraría también el lenguaje **SeRQL (sesame RDF Query Language)** (Jeen Broekstra and Arjohn Kampman. 2004). Este lenguaje está basado en RQL, pero tiene modificaciones que hace que sea más sencillo de procesar y la interpretación formal está basada en el modelo RDF. Inicialmente se desarrolló dentro del proyecto Sesame, pero hasta el 2020 formaba parte del proyecto RDF4J, que en su última versión dejó de incluirlo como parte del proyecto.

Dentro de este grupo de lenguajes también se hallan otros como **eRQL (easy RDF Query Language)** (Wleklinski., F. and Tolle, K. 2005) , desarrollado en 2003 por ICS Databases and Information Systems de la Universidad de Frankfurt. La principal característica de este lenguaje era la simplicidad de su sintaxis. No requería un gran conocimiento de la ontología, y estaba basada en palabras clave combinadas con los operadores lógicos AND y OR. A pesar de que se desarrolló un motor como erqlEngine, la comunidad de desarrollo no lo terminó usando y en la actualidad el proyecto no está activo.

#### **Lenguajes de consultas basados en SPARQL**

SPARQL (SPARQL Protocol and RDF Query Language) es un language estandarizado de consulta sobre RDF, que se convirtió en 2008 en una recomendación de W3C(W3C 2008) . Su última versión SPARQL 1.1, también es una recomendación W3C desde 2013.

Este lenguaje y todos los de este conjunto, se les denomina lenguajes relacionales porque tienen operaciones relacionales. Interpretan el contenido en RDF como un conjunto de tripletas, sin necesidad de que exista un esquema. Están diseñados con una similitud al lenguaje de consultas de bases de datos SQL (Structured Query Language).

Se ha convertido en el lenguaje de consulta de la web semántica (Bhushan 2016) . La mayoría de plataformas de soporte de grafos de conocimiento, ofrecen un punto SPARQL, admitiendo consultas en este lenguaje desde la web.

Un ejemplo de consulta en este lenguaje sería el siguiente:

```
SELECT ?title
```

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

WHERE

```
{  
  <http://example.org/book/book1>  
  <http://purl.org/dc/elements/1.1/title> ?title .  
}
```

Esta consulta, busca la existencia de una tripleta compuesta por los valores:

- <http://example.org/book/book1>
- <http://purl.org/dc/elements/1.1/title>
- <valor del título >

Y devuelve todos los valores de título existentes en el grafo de conocimiento.

La similitud con SQL es muy alta, y también permite incluir subconsultas, agregados, etc.

Otros lenguajes que estarían dentro de este bloque serían **SquishQL** (Miller, Seaborne et al. 2002), y **RDQL** (*RDF Data Query Language*) (W3C 2003), considerados los dos predecesores de SPARQL, el cual tomó las mejores características de cada uno, haciendo que el uso de estos lenguajes fuera cada vez menor, a pesar de que RDQL llegó a ser candidato a estándar de W3C.

Por último, dentro de este bloque cabe citar una evolución del lenguaje SPARQL, pero especialmente adaptada a las necesidades de referenciación espacial. Se trata del lenguaje de consultas GeoSPARQL (Perry, Herring et al. ). Este lenguaje ha sido descrito por el Open Geospatial Consortium (OGC). El objetivo de GeoSPARQL es la posibilidad de realizar consultas de forma estandarizada sobre datos geoespaciales expresados en RDF cuya geometría ha sido previamente especificada. Con este lenguaje, por ejemplo, se pueden realizar búsquedas a través del contenido de rectángulos cuyos vértices están expresados en coordenadas geoespaciales, en varios sistemas de coordenadas. Debido a la importancia del uso de datos geoespaciales en los grafos de conocimiento, este estándar se ha convertido en el más importante dentro del ámbito de la web semántica (Car, Homburg 2021).

#### *Aplicaciones de grafo de conocimiento.*

En la actualidad hay varios trabajos publicados, describiendo el estado del arte de los grafos de conocimiento, en los que se enumeran las diferentes aplicaciones que tienen.

Unos trabajos, como es el caso de (Chen, X., Jia et al. 2020, Wang, Q., Mao et al. 2017) las exponen principalmente sobre las que actúan principalmente sobre el mismo grafo de conocimiento, para mejorarlo y las que lo hacen hacia aplicaciones externas, a las que ofrecen servicios determinados.

Las que actúan sobre el mismo grafo de conocimiento, son aplicaciones de:

- Completación del grafo, en los que el objetivo es utilizar el contenido del grafo para crear nuevos hechos, entidades, relaciones, etc. Estas aplicaciones enriquecen el grafo, creando nueva información a partir de la ya existente. Este tipo de tareas engloba subtarefas como clasificación de hechos (saber si son ciertos, o falsos), predicción de nuevos enlaces, o relaciones y eliminar entidades duplicadas. En muchas de ellas el resultado tiene un porcentaje de confianza, que a partir de un cierto valor se puede considerar correcto. Puede parecer que si estas tareas se aplican constantemente el grafo tendría más y mejor información. Sin embargo este proceso no se puede realizar de forma repetitiva, a no ser

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

que entre nueva información, ya que al estar basado en niveles de confianza, se suele generar información, pero cada vez con menor precisión, reduciendo la fiabilidad de la información en el grafo (Dong, X. L., Gabrilovich et al. 2014).

- Clasificación de entidades, que comprende aquellos trabajos que tienen como objetivo categorizar las entidades en diferentes categorías semánticas.

Las que el grafo forma parte de una aplicación principal y se usa para dar servicios a los objetivos de esta aplicación. Aquí se suelen incluir las aplicaciones de mayor uso que son para los campos de medicina, finanzas y sistemas de respuesta automatizada de preguntas, generalmente asistentes virtuales y similares.

Otros trabajos, como (Zou 2020) se centran principalmente en el uso que se hace del grafo de conocimiento, en su funcionalidad, enumerando varias aplicaciones de los grafos en diferentes áreas, profundizando más en las diferentes aplicaciones desde el punto de vista del usuario final.

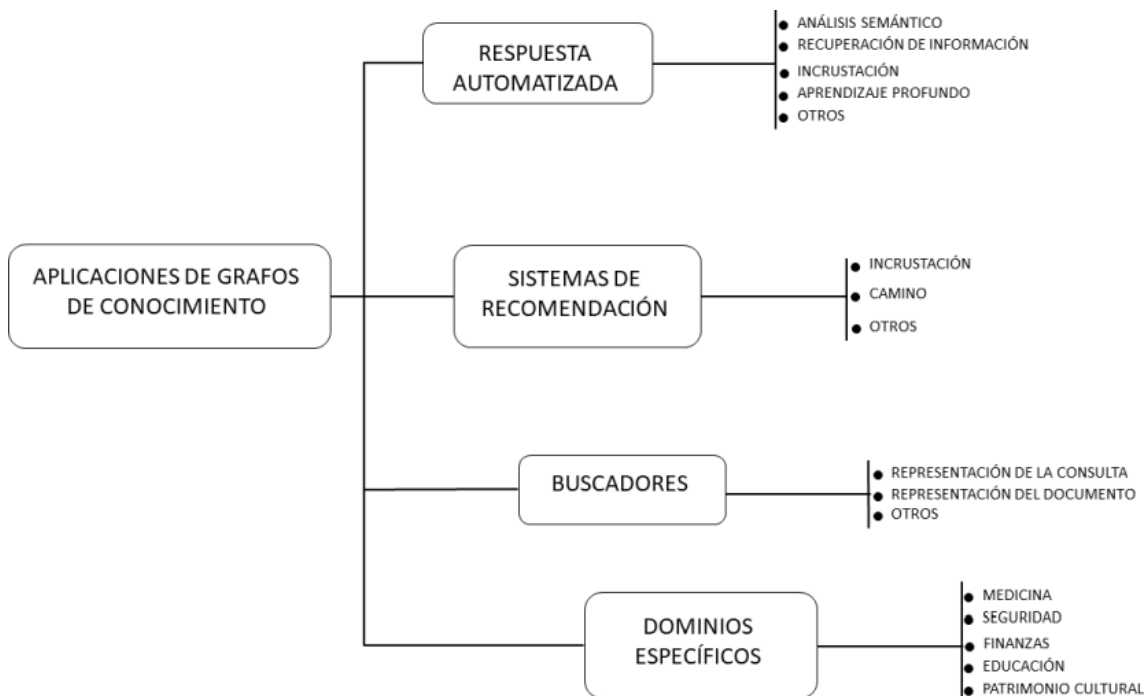


Figura 2 Aplicaciones de los grafos de conocimiento. Fuente: Adaptada de (Zou 2020).

La Figura 2 muestra un esquema con las principales áreas de aplicación de los grafos de conocimiento.

Las principales áreas de aplicación de los grafos de conocimiento son:

- Aplicaciones de respuesta automatizada. Que pueden estar basadas en:
  - Análisis semántico. Transforman preguntas formuladas en lenguaje natural en formas lógicas que pueden expresarse mediante consultas estructuradas, generalmente usando el lenguaje de consultas SPARQL. Estas consultas son refinadas, bien con ayuda de grafos externos, y/o factorizadas en consultas más pequeñas, combinando las respuestas para obtener el mejor resultado. Este tipo de metodología funciona bien con preguntas complejas.

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

- Recuperación de información. Al igual que el caso anterior, obtiene consultas estructuradas partiendo de consultas en lenguaje natural. Pero se limita a ponderar la validez de las posibles respuestas sin trabajar sobre las consultas previamente. Este tipo de metodología es adecuada para consultas simples.
  - Incrustación. Utilizan incrustaciones vectoriales de baja dimensión de la pregunta y de las entidades. Posteriormente calculan la mejor similitud entre la pregunta y las respuestas candidatas. Este método tiene mejor rendimiento que los anteriores, pero únicamente es válido para preguntas simples, ya que se ignora el orden de las palabras.
  - Aprendizaje profundo. Utilizan los avances de las técnicas de aprendizaje profundo en el campo del lenguaje natural, utilizando generalmente sistemas basados en redes neuronales, y comienzan a obtener mejores resultados que los sistemas tradicionales.
- Sistemas de recomendación.
- Incrustación. Los trabajos que siguen este método suelen aplicar algoritmos de incrustación para preprocesar el grafo de conocimiento, aplicando las incrustaciones de las entidades aprendidas a un marco de recomendación.
  - Camino. Este método es más intuitivo, consiste en diseñar y aplicar un algoritmo sobre el grafo de conocimiento, con el fin de obtener patrones y conexiones entre los nodos del grafo. De esta forma se obtiene información adicional sobre las recomendaciones. El resultado depende mucho de esas conexiones, que en algunas situaciones es muy difícil de realizar, porque las entidades y sus relaciones pertenecen a dominios diferentes. Un ejemplo de esta situación es el caso de recomendación de noticias.
- Buscadores. Estas aplicaciones obtienen utilizan grafos de conocimiento para mejorar su comprensión de las consultas y los documentos relacionados en la búsqueda de un usuario. Los métodos más utilizados son:
- Representación de la consulta. Consistentes en aplicar un grafo para obtener entidades relacionadas y sus textos para ampliar la consulta.
  - Representación del documento. Los documentos se enriquecen añadiendo entidades anotadas en el modelo de espacio vectorial del documento. Estos se clasifican en función de las coincidencias con las consultas en el espacio de entidades.
- Dominios específicos. Generalmente, los dominios en los que se aplica el uso de grafos de conocimiento, son aquellos que tienen un conjunto de entidades relacionadas entre sí de forma diversa y compleja. Aunque cada vez hay más campos en los que se aplica el uso de grafos de conocimiento, los campos en los que más se ha explotado su uso son los siguientes:
- Medicina. La información sanitaria ha crecido de forma exponencial en los últimos años. Debido a ello, la incorporación de grafos de conocimiento en sistemas de información sanitaria es cada vez más importante, existen trabajos desde la integración de servicios médicos (Shi, L., Li et al. 2017) hasta generación de grafos partiendo de registros médicos (Rotmensch, Halpern et al. 2017) para asistir a los profesionales en urgencias.
  - Seguridad informática. La seguridad informática es un tema que preocupa mucho a la sociedad y la industria. Los grafos de conocimiento se aplican en este campo para predecir ataques dinámicos, asociando determinados eventos y estados del sistema con los posibles pasos de un ataque.



### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

- Finanzas. En esta campo, los grafos de conocimiento se usan para varios objetivos, hay trabajos para conocer el estado financiero de las empresas (Liu, J. 2018), relacionándolo con las noticias publicadas por y sobre la empresa, hasta otros creados para obtener información sobre incidentes para empresas de seguros (Elnagdy, Qiu et al. 2016).
  - Noticias. Los agregadores de noticias son sistemas en los que cada vez tienen más presencia los grafos de conocimiento. Cada vez es más habitual que los sistemas recomienden noticias basándose simplemente en las consultas de los usuarios, sin que tenga que personalizar nada, o muy poco. Un ejemplo de aplicación de grafos para este fin se puede observar en (Rudnik, Ehrhart et al. 2019) . En los últimos años se ha puesto muy de moda la aparición de noticias falsas en las redes sociales con el fin de influir en la población. El trabajo de (Ciampaglia, Shiralkar et al. 2015) usa grafos de conocimiento para detectar estas noticias falsas.
  - Educación. En este ámbito se han desarrollado varios trabajos (Chen, P., Lu et al. 2018, Rizun 2019) que usan grafos para recomendar recursos de aprendizaje a los alumnos, en función de su rendimiento y conocimiento de la materia.
  - Patrimonio cultural. En este campo, donde hoy en día hay un gran número de museos y centros que se resisten a publicar datos digitales sobre las piezas que mantienen, la estructuración y difusión de la información es una necesidad acuciante. Gracias al impulso de entidades gubernamentales, como es el caso de la UE (European Commission 2021) , en la última década se han podido desarrollar iniciativas para mantener y difundir este tipo de datos heterogéneos (obras de arte, restos arqueológicos, etc.) que en ocasiones no está digitalizado y en generalmente, cuando lo estaba era siguiendo una clasificación y protocolos propios, además de usar muchas veces soportes digitales cerrados. Algunos ejemplos de aplicaciones desarrolladas en este campo que usan grafos se encuentran en (Carriero, Gangemi et al. 2019, Pantoja, J., Docampo, J., Martín, A., Maturana, R.A., Navalón, C. 6/04/2016)
- Otros. Además de los campos anteriormente expuestos, que son los que más desarrollos e investigaciones ha generado. La aplicación de grafos de conocimiento se está extendiendo a prácticamente cualquier campo que necesite estructurar, o buscar información en grandes repositorios de datos heterogéneos. Ejemplos de ello lo tenemos en redes sociales (Bayrakdar, Yucedag et al. 2020), o las ciencias de la tierra (Zhou, Wang et al. 2021).

#### 3.3. GRAFOS DE CONOCIMIENTO EN EL PATRIMONIO CULTURAL

Dado que esta tesis se centra en la visualización de grafos de patrimonio cultural, a continuación, se analizará con más detalle la problemática del ámbito, sus necesidades y las herramientas existentes a tener en cuenta para implementar un grado de conocimiento en este campo.

##### 3.3.1. Situación.

Según (UNESCO 2014) , “Patrimonio Cultural es, en su sentido más amplio, tanto un producto como un proceso, que proporciona a las sociedades una riqueza de recursos que se heredan del pasado, se crean en el presente y se otorgan en beneficio de las generaciones futuras. Lo más importante es que no sólo incluye el patrimonio material, sino también el natural y el inmaterial.”

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

La definición de patrimonio cultural ha sufrido muchos cambios a lo largo de la historia, especialmente en lo referente a lo que se considera dentro del patrimonio cultural (Vecco 2010). En la actualidad, la sociedad en general no sólo acepta de forma generalizada los contenidos enunciados por UNESCO, sino que hay un esfuerzo importante por darle el valor que se merece y por preservarlo. Las instituciones gubernamentales son cada vez más conscientes de su importancia y por ello, además de promover legislación para protegerlo, en la última década se está haciendo un especial esfuerzo por preservarlo, destinando financiación para lograr este objetivo (European Commission 2017).

Una de las formas de preservar este patrimonio, por lo menos en cuanto al legado e información se refiere, es digitalizando dicha información. Además del uso del material multimedia más usual: imágenes, sonidos y videos, cada vez se están aplicando técnicas de escaneo tridimensional para mantener información realista de edificios y lugares de grandes dimensiones (Acke, De Vis et al. 2021). Este material multimedia, junto a la información que se tiene de la pieza: descripción, autor, lugar y momento de creación, materiales usados, técnicas, etc. Es información que en la actualidad es sencillo de digitalizar. A pesar de que no es muy complejo, hay muchas colecciones, generalmente conservadas por entidades de pequeña dimensión, u otras con problemas económicos, que tienen digitalizados los datos sobre sus piezas, o bien lo tienen en sistemas cerrados (Samaroudi, Echavarria et al. 2020), generalmente en bases de datos propietarias, o en documentos de aplicaciones de ofimática. Por lo tanto, estas colecciones no están accesibles para el público en general y menos aún para ser procesadas por sistemas informáticos. En el caso de que estos datos estén digitalizados, generalmente están clasificados por procesos de catalogación propios, o locales, generalmente en el idioma oficial de la región.

Las instituciones europeas, conocedoras de este problema, crearon a finales de 2011 la recomendación para preservar, digitalizar y facilitar el acceso online a los datos de patrimonio cultural (European Commission 2021):

- Promoviendo y financiando iniciativas como el portal Europeana Collections (EUROPEANA 2021) que mantiene y da libre acceso a más de 50 millones de archivos digitalizados
- Ha dotado con más de 70 millones de euros de financiación a 6 programas dentro de Horizon 2020 Societal Challenges entre 2014 y 2020, para el desarrollo de proyectos que faciliten la preservación del patrimonio cultural con herramientas para la digitalización de las colecciones.

En Europa el patrimonio cultural además de ser un conjunto de valor cultural de gran valor que hay que preservar, también es un activo que favorece el desarrollo de una economía basada en lo digital (García Moreno, A., Hernández-Pérez, T. 2019).

Afortunadamente también hay varias colecciones que tienen sus datos digitalizados y están abiertos y accesibles tanto para personas, como para aplicaciones externas, mediante servicios web, o tecnologías similares:

- Una gran parte de los museos internacionales más conocidos (Louvre 2021, Museo del Prado. 2021, Victorian Albert Museum 2021, The Metropolitan 2020).
- Portales web que se dedican a recopilar, publicar y dejar accesible esta información. Algunos de ellos financiados por entidades gubernamentales, al menos en su creación, como puede ser la red de información generada en el proyecto ARIADNE (Meghini, Niccolucci et al. 2017), o el portal EUROPEANA (EUROPEANA 2021). Otros creados por entidades privadas, como el portal de Google Arts & Culture (Google 2021a).
- Instituciones gubernamentales, como es el caso de la Red Digital de Colecciones de Museos de España (Ministerio de Cultura y Deporte. 2021).

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTUAL MEDIANTE GRAFOS DE CONOCIMIENTO

A pesar de estos esfuerzos sólo el 23% de los datos de patrimonio cultural europeo están digitalizados (Jan Nauta, G., van den Heuvel, W. 2015)

En la Figura 3 se muestra el porcentaje de elementos no digitales de las colecciones de patrimonio cultural europeo, por tipo de institución, que han sido digitalizados, los que necesitan digitalizarse y los que no se digitalizarán.

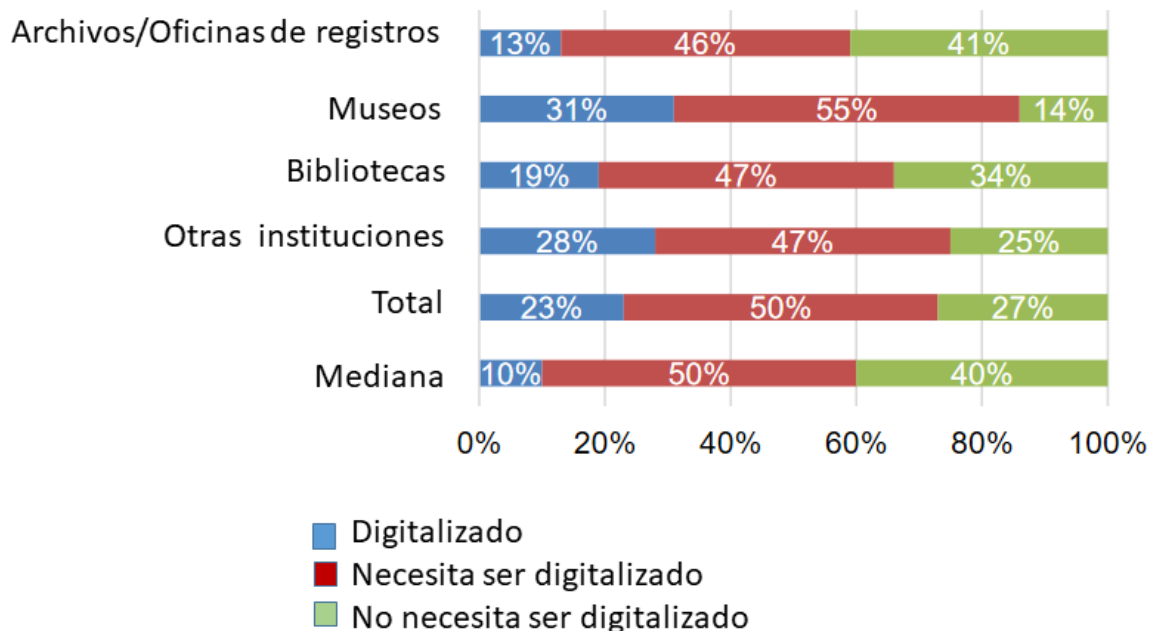


Figura 3 Porcentaje de datos digitalizados en instituciones de patrimonio cultural europeo. Adaptada de (Jan Nauta, G., van den Heuvel, W. 2015)

En la Figura 4 se representa una gráfica con el porcentaje de los datos digitalizados de patrimonio cultural europeo, por tipo de institución, que están disponibles online para el uso general.

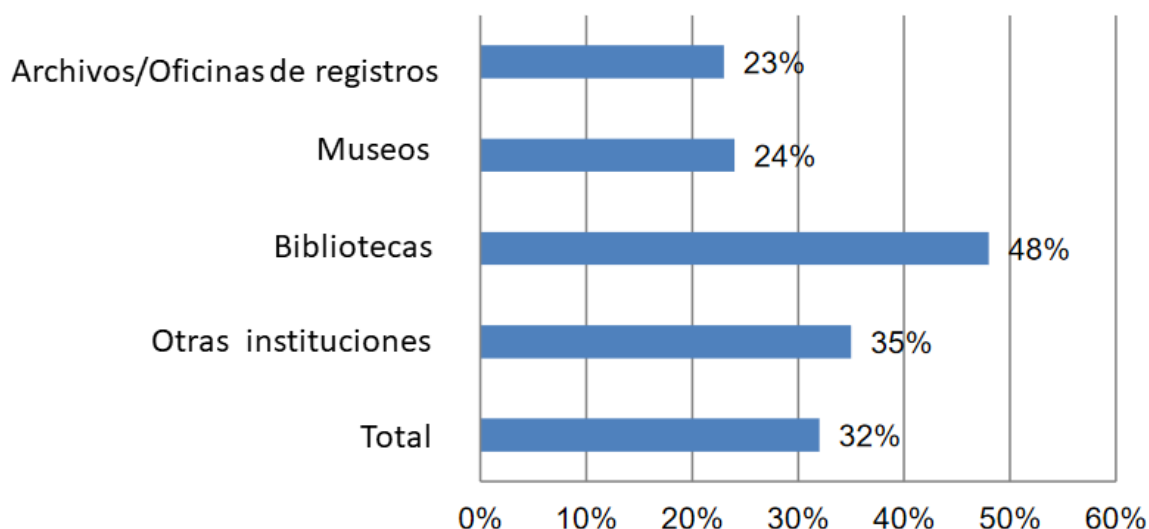


Figura 4 Porcentaje de datos digitalizados de patrimonio cultural europeo que están disponibles para su consulta online. Adaptada de (Jan Nauta, G., van den Heuvel, W. 2015)

NEMO (Network of European Museum Organisations) ha publicado en 2020 un informe (NEMO Working Group Digitalisation and IPR 2020) sobre la digitalización y difusión de datos

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

del patrimonio cultural europeo, pero únicamente en museos. En este informe se plasma que los museos europeos sólo han digitalizado el 43% de sus colecciones. Comparando con el informe previo, generado 5 años antes, se observa que sólo ha crecido un 12% el porcentaje de datos digitalizados. Además, este informe indica que los museos ofrecen públicamente acceso online a un 20% de sus colecciones. Un porcentaje menor que en el último informe. Esto indica que muy pocos de los nuevos datos digitalizados han sido publicados online.

Estos datos, reflejan que aún queda mucho trabajo por realizar para poder tener digitalizados la mayoría de los datos de patrimonio cultural europeo.

Además de la digitalización de los datos, mediante información textual, o archivos multimedia, es necesario realizar una labor de estructuración y catalogación, además de elegir un formato adecuado para que los datos puedan ser procesados por aplicaciones externas, o simplemente ser publicados junto a datos de otras entidades con un mínimo de uniformidad.

Los objetos de patrimonio cultural tienen asociada una información compleja, además de datos multimedia, hay que tener en cuenta el autor, o autores, que la crearon, si existen, al igual que quién la encargó, y para qué, cuándo y dónde se realizó, dónde se encontró y quién la encontró, dónde está en la actualidad y quién es su propietario, dimensiones, materiales, y más información específica dependiendo de cada tipo de objeto. Las piezas, pueden tener una entidad propia, o formar parte de un conjunto más grande.

Este primer bloque de información se corresponde con información básica del objeto, que principalmente son datos que dan información sobre el objeto y su origen.

Además de esta información, que se ha etiquetado en este trabajo como básica, hay que tener en cuenta que la información de estos objetos fluctúa en el tiempo. Los objetos de este tipo de patrimonio pueden tener una vida desde décadas, hasta varios miles de años. En ese tiempo han podido cambiar de propietario, por un regalo, por una compra, por una conquista, etc., además de ser cambiar su uso y ser modificados por otros autores, aplicando diferentes técnicas y materiales, y estos hechos realizarse en diferentes lugares del espacio y del tiempo.

En el caso de piezas de un museo, o de un archivo, la catalogación y descripción de la pieza, también varía en el tiempo, y es realizada por diferentes profesionales, los cuales aportan y cambian diferentes datos, según su criterio y conocimientos. Estos cambios también deben guardarse, con el fin de que no se pierda información sobre la pieza. Obviamente, también debe de especificarse la lengua en la que se introducen los nuevos datos y descripciones.

Si se desea que la información sobre los diferentes objetos pueda ser intercambiada con otras aplicaciones, simplemente en referencia al bloque de información básica del objeto, es necesario que exista, como mínimo, unificación en los siguientes datos:

- Tipo de objeto.
- Materiales utilizados.
- Técnicas aplicadas para su elaboración.
- Referencias temporales.
- Referencias espaciales.

Por lo tanto, esta información, ya sea en valores de propiedades, o dentro de cualquier información descriptiva, si se desea que pueda ser comprendida por otras aplicaciones tiene que estar especificada por un vocabulario. Con la misma finalidad, la de compartir datos para que puedan ser comprendidos por aplicaciones externas, pero atendiendo a los datos que fluctúan

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

en el tiempo, además de actividades y propiedades que requieren de una estructuración más compleja, se hace necesario el uso de modelos de referencia. Estos modelos se representan mediante una, o varias ontologías. Estas ontologías, junto con los vocabularios, serán la base para construir un grafo de conocimiento, que contendrá la información sobre colecciones de datos de patrimonio cultural.

Seguidamente, en esta investigación, se detalla en primer orden el estado del arte de los vocabularios y su aplicación en el campo del patrimonio cultural. A continuación, se muestra una discusión sobre los diferentes modelos de referencia existentes en este ámbito y su aplicación.

#### *3.3.2. Vocabularios*

El uso de vocabularios controlados, o tesauros, en los ámbitos de la web semántica y la estructuración de la información ha experimentado un renovado interés en la última década (García-Marco 2016). Reflejo de este interés ha sido la actualización de su formato ISO 2788 (ISO 1986) a la ISO 25964 (ISO 2011, ISO 2013) así como una nueva versión del consorcio W3C: SKOS (Miles, Bechhofer 2009). En estas nuevas versiones se aprecia una evolución del tesoro basado en términos al tesoro basado en conceptos, siendo SKOS el formato que más se usa generalmente. (Martínez-González, Alvite-Díez 2020).

En la actualidad hay una gran diversidad de tesauros en el campo de patrimonio cultural, y generalmente la mayoría de ellos están en formato ISO 25964 y SKOS, y están accesibles de forma online, además de permitir el acceso mediante un punto SPARQL y la descarga completa en forma de archivo. Algunos ejemplos de estos vocabularios son:

- Vocabularios realizados por entidades de prestigio internacional, como es el caso de los tesauros de patrimonio cultural del Getty Research Institute (Getty 1997a, Getty 1997b, Getty 1997c, Getty 2005), o el tesoro de educación, cultura, y ciencias naturales de UNESCO (UNESCO 1977)
- Vocabularios de patrimonio cultural creados por organizaciones compuestas por varias entidades de este campo. Una de las asociaciones más activas últimamente es el FISH Forum on Information Standards in Heritage (FISH 1998), que ofrece más de 10 tesauros de arqueología, materiales, tipología, cronología, elementos aéreos, marítimos, crímenes históricos, etc., en su página web (FISH 2003) . Además, recientemente ha unido sus esfuerzos a los logrados por el proyecto SENASCHAL (Tudhope, D., Binding, C. 2013), que también ha elaborado varios tesauros sobre el mismo campo. Estos tesauros hacen especial hincapié en los datos del patrimonio cultural del Reino Unido. En este caso, no se ofrece un punto de acceso SPARQL, pero sí que pueden descargarse los tesauros y visualizarse. La mayoría de ellos, sobre todo los del proyecto SENASCHAL están en formato SKOS.
- Entidades gubernamentales, como la Red Digital de Colecciones de Museos de España (Ministerio de Cultura y Deporte. 2021), que además de vocabularios sobre tipología, materiales, técnicas y regiones, mantiene información de más de 330.000 bienes culturales de colecciones de 13 comunidades autónomas. Los tesauros pueden visualizarse de forma online, además de descargarse y realizar consultas mediante un punto de acceso SPARQL.

El uso de estos vocabularios es fundamental para poder definir de forma inequívoca la tipología de un objeto, los materiales de que está hecho, etc. Usando lenguaje natural, o catálogos propios es muy difícil integrar datos de diferentes fuentes. Sin embargo, este problema, aunque se ha reducido, sigue existiendo. Al haber tantos vocabularios, diferentes conjuntos de datos, usan diferentes vocabularios, y aunque la integración es más sencilla, es un trabajo difícil, que

tiene que ser realizado, o supervisado por humanos. Por este motivo sigue siendo necesario unificar criterios, también a la hora de diseñar vocabularios (Isaac, Zinn et al. 2007).

### 3.3.3. Modelos de referencia.

El uso de vocabularios favorece la integración de los datos, además de dar uniformidad al significado de los mismos. Aunque, como se ha discutido previamente, la existencia de una gran diversidad genera otros problemas nuevos, aunque menores.

Sin embargo, los vocabularios solucionan problemas en lo referente a valores de conceptos de una estructura simple. Pero cuando se trata de conceptos más complejos se requieren estructuras de un nivel superior.

Por ejemplo, mediante un vocabulario se puede representar el tipo de objeto, o los materiales que lo componen. Pero si se desea saber, si un objeto fue creado, encontrado, comprado, o restaurado, en qué lugar, además de por quién, o quiénes, es más complejo. Estos hechos requieren una relación entre dos conceptos, que, a su vez, pueden ser de varios tipos.

En la Figura 5 se muestra una instancia de un posible grafo de conocimiento en el que se ve que un ejemplar de “Don Quijote de la Mancha”, que tiene una propiedad que indica su tipo, con el valor “Libro”. Se quiere representar mediante relaciones de las clases “CREADA” Y “RESTAURADA” saber quién, cuándo y dónde se creó y se restauró ese ejemplar. Aunque es un ejemplo muy simplificado, se observa que tiene que haber una relación entre el lugar de creación, su fecha y la persona que crea el ejemplar, lo mismo para la actividad de restauración. Además, estas relaciones pueden ser con instancias de diferentes clases, por ejemplo, el libro podría haber sido creado por una instancia de “PERSONA”, o de “ORGANIZACIÓN”. Aunque hay propiedades que pueden tomar valores procedentes de un vocabulario, mantener esta información únicamente con este tipo de propiedades sería complejo de mantener.

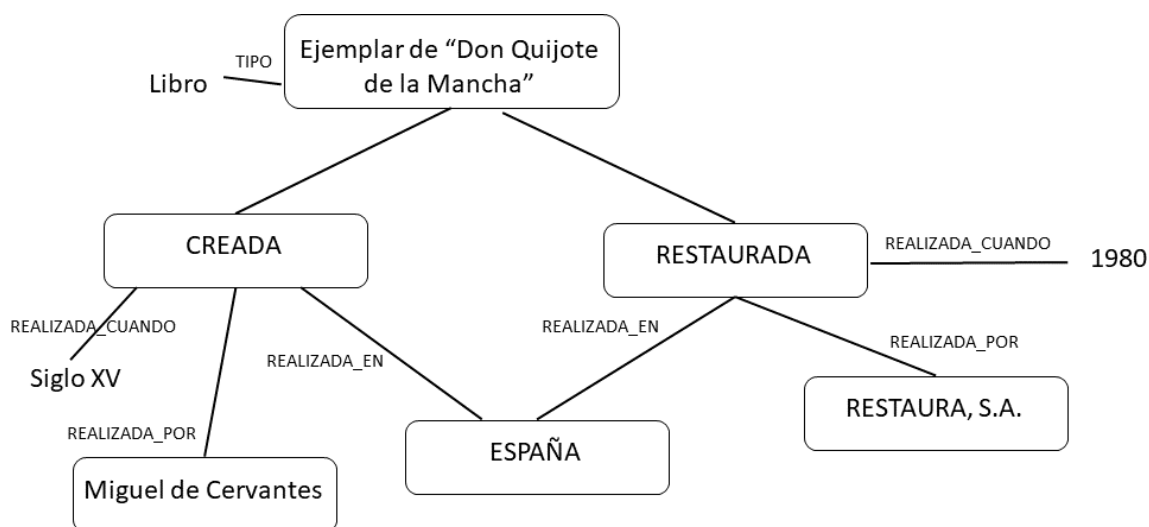


Figura 5 Ejemplo simplificado de las relaciones de una instancia en un grafo de conocimiento. Fuente: Elaboración propia.

Si con el ejemplo simplificado, se observa la necesidad de una estructura de nivel superior, es obvio que este requerimiento, es aún mayor, para intentar representar la compleja realidad del patrimonio cultural. Por lo tanto, es necesario el uso de un modelo de referencia, que esté

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

representado por una ontología, o una estructura similar, que pueda usarse para definir un grafo de conocimiento.

Esta necesidad ha sido la causa de que en las dos últimas décadas se hayan definido varios modelos de referencia para representar este tipo de datos (Nafis, Yahyaouy et al. 2019). Sin embargo, aunque ha habido varios diseños de modelos de referencia y se han desarrollado proyectos siguiéndolos, el modelo de datos más usado para representar datos de patrimonio cultural (Dijkshoorn, Aroyo et al. 2018) es el modelo CIDOC-CRM (Bekiari, C., Bruseker, G., Doerr, M., Ore, C., Stead, S., Velios, A. )

El modelo CIDOC-CRM se publicó hace veinte años, y desde entonces se ha usado en varios proyectos de investigación ARIADNE, HERACLES (Hellmund, Hertweck et al. 2018), o SILKNOW (Portalés, Sebastián et al. 2018), y también por instituciones y museos, como el Museo del Prado (Pantoja, J., Docampo, J., Martín, A., Maturana, R.A., Navalón, C. 6/04/2016). Está centrado en los eventos para el sector del patrimonio cultural, y está basado en estudios empíricos de sistemas de gestión de colecciones. Desde su publicación, la organización que lo creó publica continuas actualizaciones, la última versión, la 7.1.1, es de Abril de 2021 (Bekiari, C, Bruseker, G., Doerr, M., Ore, C., Stead, S., Velios, A. 2021). Además, este modelo ha sido reconocido como estándar ISO 21127:2014 (ISO 2014a) para el intercambio de información sobre el patrimonio cultural.

El modelo de datos de Europeana (EDM) (Doerr, M., Gradmann, S., Hennicke, S., Isaac, A., Meghini, C., Van de Sompel, E. 2021) también se usa en varios proyectos con bastante regularidad (Dijkshoorn, Aroyo et al. 2018). Este modelo se ha desarrollado para representar y estructurar los datos del patrimonio cultural de manera con el objetivo de su integración en Europeana. Es usado internamente por Europeana, pero también para agregar y difundir los datos. Es un modelo de alto nivel, que permite la integración de otros modelos, como por ejemplo CIDOC-CRM.

A pesar de que CIDOC-CRM es un modelo muy popular, debido a la complejidad y abstracción del mismo han aparecido varias extensiones del mismo, con el fin de adaptarse más a la realidad de los datos que se manejaban, como en el proyecto HERACLES, o la ontología ArchOnto (Koch, Ribeiro et al. 2020) Por ejemplo en este modelo los objetos de patrimonio cultural realizados por el hombre se representan con la clase *E22 Man-Made Object*. Esta clase es muy genérica y si en los datos a procesar se necesita trabajar con temas muy concretos de diferentes tipos de objetos, probablemente haya que extenderla, o utilizar otros recursos para poder integrar estos datos en CIDOC-CRM. El problema adicional es que es un modelo complejo y para integrar las clases y propiedades necesarios requiere un esfuerzo importante.

El principal problema a la hora de diseñar un modelo es encontrar un equilibrio entre lo genérico y lo concreto. Cuando más genérico es un modelo, más amplio es el espectro de escenarios que puede representar, el problema es que se requieren más extensiones para representar datos concretos de los conceptos a modelizar (Gonzalez-Perez, Martín-Rodilla et al. 2012).

Hay autores que indican que tanto CIDOC-CRM, como EDM, son modelos muy genéricos, que requieren un esfuerzo importante para extender y poder representar la realidad a representar. Debido a ello se publican y proponen nuevos modelos, como puede ser el caso del modelo CHARM (González-Pérez 2018) . En este modelo se presenta una aproximación en dos pasos, por una parte, se parte de un modelo muy genérico como base, y por otra, se permite realizar extensiones mediante unas directrices, de esta forma, se cubre la parte genérica, y se permite representar de forma sencilla la parte más concreta de la realidad a representar.

### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

Además de estos problemas previamente comentados, existe un problema adicional vinculado estrechamente con los datos de patrimonio cultural. Se trata de la incerteza. Debido a falta de documentación, imposibilidad de realizar pruebas técnicas y otros motivos, es muy frecuente que existan datos sobre los objetos de patrimonio cultural que se desconozcan. Esto afecta a todos los niveles: espaciales, temporales, pero también otros como su uso, su creador, su propietario, técnicas utilizadas para su construcción, etc.

Ante este problema de incerteza se suele reaccionar de dos formas, no se indica el dato, o bien se le asumen varias posibilidades, de acuerdo a otros datos que sí son conocidos y al criterio de los investigadores. Por este motivo es frecuente que datos tan básicos, como dónde, quién, o cuándo se creó un objeto aparezcan, a veces con varias posibilidades.

Los datos de patrimonio cultural poseen una dificultad extra, falta de información o incerteza. Bien por motivos técnicos, o bien por falta de documentación, suele ser frecuente el desconocimiento de parte de información que se desconoce, o que presente incerteza. Esto sucede con información importante, como los diferentes propietarios que ha tenido a lo largo del tiempo, o el autor, hasta información fundamental como el lugar, o el tiempo en que se produjo. Para ilustrar esta situación, a modo de ejemplo, puede consultarse este enlace de la textiloteca del Museo Tèxtil de Terrasa (IMATEX 2021), en el que se observa que hay un tejido cuya procedencia no está clara y se le atribuyen orígenes tan distantes como Francia, Países Bajos y China.

Por ejemplo, en este enlace de la textiloteca del Museo Tèxtil de Terrasa (IMATEX 2021), se observa que hay un tejido cuya procedencia no está clara y se le atribuyen orígenes tan distantes como Francia, Países Bajos y China. Este hecho no es que sea la norma en los datos de patrimonio cultural, pero tampoco es precisamente extraño y sucede con relativa frecuencia.

Este problema se puede extender a varias propiedades del objeto, y por supuesto a las que fluctúan en el tiempo. Por ejemplo, se pueden conocer los orígenes del objeto, pero desconocer algunos de sus propietarios, que incluso pudieron encargar modificaciones del objeto. Conociendo cuando y que se modificó, pero no quién lo hizo, ni por qué.

A nivel de representación mediante modelos de referencia, el problema aparece al tener que soportar varias y diferentes opciones sobre un dato. Esto es un problema, porque modelos como CIDOC-CRM no ofrecen una opción concreta sobre cómo representar estas y clasificar las posibles interpretaciones de un conocimiento. Debido a esto se han diseñado y desarrollado extensiones del modelo, no oficiales y adaptadas a las necesidades surgidas en diferentes desarrollos (Cantone, D. and Cristofaro, S. and Nicolosi-Asmundo, M. and Prado, F. and Santamaria, D.F. and Spampinato, D. 2019, Van Ruymbeke, Hallot et al. 2018). Al no ser una representación oficial, esto supone un problema a la hora de compartir los datos con otras aplicaciones.

#### 3.4. CONCLUSIONES

En este capítulo se ha expuesto que en la actualidad el campo de investigación sobre los grafos de conocimiento está en plena expansión. Hay multitud de proyectos de investigación y desarrollo, así como comerciales, que aplican sus resultados. Con mucha frecuencia aparecen resultados y algoritmos nuevos, que permiten mejorar su rendimiento, el nivel de razonamiento, así como herramientas para poder explotarlos con un rendimiento adecuado.

También se ha discutido la realidad sobre el nivel de digitalización de los datos de patrimonio cultural europeo. Como resultado de esta discusión queda manifiesto que aún hay mucho trabajo que hacer en este campo, y aunque las instituciones gubernamentales, como la Unión



### CAPÍTULO 3. REPRESENTACIÓN DE DATOS DEL PATRIMONIO CULTURAL MEDIANTE GRAFOS DE CONOCIMIENTO

Europea, hacen importantes esfuerzos para que esta situación cambié, no son suficientes. Se impulsa la idea de que el patrimonio cultural también fomenta de forma sostenible la economía, a través de la difusión digital, pero hay que seguir esforzándose. Este esfuerzo ha de centrarse en dos áreas, por una parte, la digitalización de estos datos, y por otra parte su difusión a través de plataformas online, tanto para personas, como para aplicaciones externas.

Según mi criterio se da un marco muy bueno para que se logre el objetivo de digitalizar estos datos. A nivel tecnológico, las plataformas online están más que establecidas en nuestra sociedad. Por ello, existiendo la necesidad, que además de cultural y social, tiene un interés económico claro, junto a que la tecnología está lo suficientemente desarrollada para soportar y explotar esta información, lo más complejo está solucionado. Ahora restan dos tareas, una que tiene una base económica, que es encontrar recursos para digitalizar la información, otra, unificar criterios y protocolos para intercambiar la información.

Esta última tarea de unificación, desde mi punto de vista es la más compleja. Esta afirmación es consecuencia de lo analizado en el resto del capítulo.

A pesar de que hay disponible una gran cantidad de vocabularios, y formatos definidos, algunos como estándares, y varias herramientas para procesarlo, muchos de ellos son redundantes. Es decir, hay tal variedad, que hay mucho solapamiento, obligando a desarrolladores y a las entidades que publican información, a elegir uno. Además, para compartir e integrar la información, con otra herramienta que use otros vocabularios, habrá que hacer un mapeo de información. Esto no es un problema muy grave a nivel técnico, pero requiere una participación importante de supervisión humana.

Este mismo problema lo tenemos en la variedad de modelos de referencia publicados y usados en los diferentes proyectos. En este punto, es verdad, que parece probable que el modelo de referencia CIDOC-CRM sea el que termine siendo el más usado. Este modelo sufre continuas actualizaciones, pero hay algunas como las requeridas para soportar la incerteza de los datos, que siguen resistiéndose a ser incluidas, y hace que cada vez más proyectos hagan extensiones propias para representar algo tan básico como saber dónde y cuándo se creó un objeto de patrimonio cultural.

Dada la situación, en los próximos años ha de hacerse un esfuerzo importante por solucionar este problema de uniformidad, que en mi opinión es el más difícil de solucionar. Probablemente se alcance una situación, donde coexistirá un conjunto reducido de plataformas de grandes dimensiones, que mantienen y difunden una gran cantidad de información. Cada una de estas plataformas usará un modelo diferente, pero podrá ser mapeado para poder compartir la información.

## CAPÍTULO 4. VISUALIZACIÓN DE DATOS INTERACTIVA DE PATRIMONIO CULTURAL

### 4.1. RESUMEN

La visualización de información es un campo muy activo en la actualidad. Hay trabajos y resultados de este campo en una gran cantidad de ámbitos.

En este capítulo se analiza el estado del arte de este campo. Aunque se parte de una visión general, el análisis se enfoca progresivamente sobre la visualización de datos espacio temporales, y también sobre datos de patrimonio cultural. Esto es debido a que los datos espacio temporales están vinculados a los de patrimonio cultural. En este análisis, además de describir diferentes técnicas y trabajos, también se explican las diferentes herramientas utilizadas para su representación. Los problemas encontrados a la hora de representar visualmente este tipo de información y las diferentes soluciones aportadas también serán estudiados.

En el capítulo, con el fin de definir de forma estructurada como se ha de visualizar la información, se analizan los estudios existentes que exponen cómo visualizar la información mediante una ontología. Además, dado que los datos a visualizar que son objeto de estudio en esta tesis, están en un grafo de conocimiento, se estudia la aplicación de estos trabajos sobre datos que están en dichos grafos, en una ontología, o base de conocimiento. Se concluye exponiendo las ventajas, e inconvenientes de los diferentes modelos analizados.

### 4.2. VISUALIZACIÓN DE DATOS

La visualización de datos, su análisis e interpretación, son áreas de gran interés para la comunidad científica (Liu, S., Cui et al. 2014) . En la actualidad cantidades masivas de información son procesadas constantemente, en ocasiones para realizar investigaciones que permiten descubrimientos científicos (Lavalle, S., Lesser, E., Shockley, R., Hopkins, M.S., Kruschwitz, N. 2011) Por ello, la visualización de datos dirigida a detectar patrones y relaciones se ha convertido en un área de gran importancia.

Las entidades gubernamentales cada vez son más conscientes de la importancia política, económica y de los beneficios sociales asociados a publicar los datos, que obtiene para su gestión (Zhao, Fan 2018). Esto ha generado una cantidad ingente de información para desarrolladores de aplicaciones, que usan estos datos, pero también para investigadores que usan esta información para validar y constatar sus investigaciones.

Actualmente las redes sociales son medios de comunicación y difusión que se explotan continuamente. La mayoría de estas plataformas ofrecen APIs públicas para acceder a sus datos. Teniendo en cuenta que generan cerca de 2.5 quintillones de bytes diariamente y existen una gran variedad de herramientas y trabajos que permiten analizar sus datos (Jaffali, Jamoussi et al. 2020), se convierten en una fuente de información de todo tipo (social, científico, política, etc.).

Entre los diferentes y variados datos que están disponibles, los datos espacio temporales son los conjuntos de datos que se analizan con más frecuencia. En este tipo de datos la evolución de la información está relacionada con las variables tiempo y espacio. El objetivo general de analizar estos datos es conocer dónde y cuándo fluctúan. Se han desarrollado multitud de aplicaciones y estudios (Zhong, Wang et al. 2012, Bach, Benjamin, Dragicevic et al. Jun 11, 2014) en este campo, las cuales han requerido estrategias especiales para la representación gráfica de

información con varias dimensiones. Para alcanzar este objetivo, durante las dos últimas décadas, los datos se han estructurado en diferentes niveles, los cuales se procesan aplicando una serie de operaciones. Estas clasificaciones definen qué se puede consultar y que se puede obtener en cada consulta. A continuación, se citan y detallan las técnicas de clasificación más utilizadas, así como sus resultados.

#### 4.2.1. Clasificación de datos espacio temporales

Según Peuquet (Peuquet 1994), la información de los datos espacio temporales está dividida en tres grupos, o partes: “dónde”, “cuándo” y “qué”. Estos grupos describen una ubicación, un momento, o intervalo de tiempo, y la información sobre los objetos de estudio relacionada con esos valores de espacio y de tiempo. Esta clasificación es la principal y más seguida, aunque algunos autores la extienden, como Andrienko (Andrienko, Natalia, Andrienko et al. 2003), que clasifica el espacio en puntos, regiones y trayectorias.

Con la evolución de la tecnología este tipo de información ha cambiado, principalmente en tamaño, hay mucha más información disponible y más atributos a considerar en el último bloque de datos, el que contiene la información asociada a una zona y a un tiempo concretos. La visualización de estos datos será más compleja, cuanta más información exista, pero también permite realizar más análisis y permite detectar mejor posibles relaciones, o patrones.

Tabla 2 Tabla con la lista de tareas, o consultas posibles, por cada nivel. Fuente: (Wang, D., Guo et al. 2017)

| Nivel | Tareas contenidas por nivel.  |
|-------|---|
| 1     | dónde + cuándo + quién → qué<br>dónde + cuándo + qué → quién<br>dónde + quién + qué → cuándo<br>quién + qué + cuándo → dónde  |
| 2     | dónde + cuándo → quién + qué<br>qué + cuándo → quién + dónde<br>dónde + quién → cuándo + qué<br>cuándo+ quién → dónde + qué<br>dónde + quién → cuándo + qué<br>quién + cuándo → dónde + qué |
| 3     | dónde → cuándo + quién + qué<br>qué → cuándo + quién + dónde<br>cuándo → cuándo + dónde + qué   |
| 4     | NADA → cuándo + dónde + qué + quién   |

Debido al incremento de datos, tanto en cantidad, como en complejidad, se han desarrollado nuevos métodos para clasificar los datos espacio temporales. Un ejemplo de esto lo tenemos en la propuesta de Wang (Wang, D., Guo et al. 2017), que añade un nuevo grupo de datos: “quién”, con el que se refiere a objetos, o grupo de objetos con el grupo “qué”. Estas cuatro partes, o grupos, pueden combinarse en cuatro niveles, generando diferentes consultas, de mayor a

menor número de variables. De esta forma, cuantas menos variables se usen para definir la consulta más información se mostrará, y también se reducirá la certeza del análisis. En la Tabla 2 se muestran las diferentes tareas, o diferentes consultas, que se pueden realizar en función del nivel. Estas nuevas formas de clasificar los datos generan nuevos resultados, que permiten obtener un mejor análisis de los datos y descubrir relaciones, o patrones, de forma sencilla.

Sin embargo, dado que la realidad es muy compleja, a medida que los sistemas de visualización mejoran, se generan nuevas demandas en el campo de visualización de datos. Esto sucede cuando un objeto está relacionado con otros objetos. En este caso, los objetos relacionados son nuevos objetos y atributos del objeto inicial.

#### *4.2.2. Agrupamiento de objetos: Clustering.*

Uno de los principales problemas de la visualización de datos en el espacio, cuando hay una gran cantidad de datos, es que con mucha frecuencia hay un gran número de datos muy próximos espacialmente, o incluso en el mismo punto espacial. Un ejemplo clásico en el ámbito de patrimonio cultural son los museos, el museo de Louvre tiene una base de datos de más de 480.000 objetos digitalizados (Louvre 2021). Obviamente, no tiene ningún sentido representar todos esos datos en un mismo punto, ni siquiera en una misma región. Ya que, debido al solapamiento, sería difícil discernir entre tanta información.

Una solución clásica a este problema es la aplicación de agrupamientos, o “clustering”. Que consiste en representar todos esos objetos mediante una representación gráfica adicional. En la actualidad se han desarrollado numerosos algoritmos para obtener grupos de forma automatizada (Shi, Z., Pun-Cheng 2019, Ansari, Ahmad et al. 2020). En cuanto a las representaciones gráficas, también se han estudiado varias soluciones, en función del tipo de elemento a agrupar:

- Puntos. La mayoría consiste en elegir un punto real, o céntrico al conjunto de puntos, que represente al grupo. Este punto puede variar el tamaño, o el color, en función del número de puntos que representa, o sus características, e incluso indicar el número de datos que representa (Jänicke, Heine et al. 2013). En la Figura 6 se muestran representaciones de los resultados de la aplicación de estos algoritmos.
- Trayectorias. Principalmente se usan líneas, o arcos, para representar las trayectorias, combinando con colores, o tamaños, para indicar cantidad de trayectorias, u otras características. Este tipo de recursos se usan en los mapas de flujos (Guo 2009). En la Figura 7 se muestra una visualización de mapas de flujo con varios elementos gráficos adicionales (color, dirección, grosor).
- Regiones. Para representar la presencia de objetos en una región, las técnicas más habituales consisten en usar mapas de color y/o de temperatura sobre la región (Ku, Liaw et al. 2016). De esta forma, además de ver la presencia de datos en la región, se puede ver cómo fluctúan los datos en las diferentes partes de la región. En la Figura 8 se muestra una representación que utiliza esta técnica para mostrar la incidencia de cáncer en Taiwán.

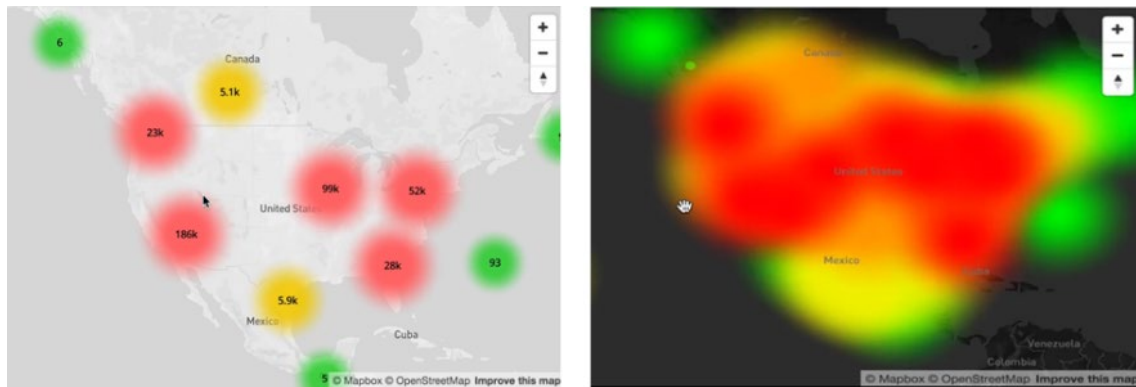


Figura 6 Ejemplos de agrupamiento de puntos, por color e indicando cardinalidad (a la izquierda), por mapas de color (a la derecha). Fuente: Capturas de videos publicados del algoritmo SuperCluster (Agafonkin 2016)



Figura 7 Ejemplo de agrupamiento de trayectorias en mapas de flujo. El color, grosor y la dirección aportan información adicional. Fuente: (Guo 2009)

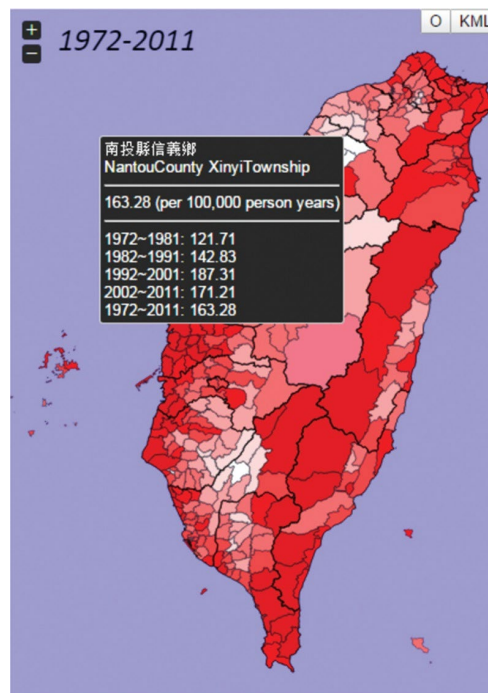


Figura 8 Ejemplo de agrupamiento de color por regiones. Fuente: Incidencia de cáncer en Taiwan, adaptada de (Ku, Liaw et al. 2016).

Tradicionalmente los algoritmos de clustering se basan en algoritmos de inteligencia artificial, concretamente usando algoritmos de aprendizaje no supervisado, tanto jerárquicos, como de partición (Shi, Z., Pun-Cheng 2019), como puedan ser el K-MEANS (MacQueen 1967), DBSCAN (Ester M, Kriegel HP, Sander J, Xu X 1996), o similares. Estos algoritmos pueden dar buenos resultados, aunque depende de muchos factores: la dispersión de los datos, la convexidad, el nivel de ruido, o datos relativamente aislados de cualquier grupo posible, pero además del resultado, tienen dos problemas adicionales:

- El primero es el alto rendimiento que requieren. La ejecución de estos algoritmos conlleva un procesamiento de cálculo elevado, lo cual hace que sea complejo de aplicar en aplicaciones interactivas en tiempo real con conjuntos con un gran número de datos. Ejecutar algoritmos costosos cada vez que un usuario active un filtro, o seleccione un intervalo temporal es inviable, si se desea que sea usable para el usuario.
- El segundo es que estos algoritmos están diseñados para crear grupos sin información adicional de los objetos. Cuando en el conjunto de datos se puede encontrar mucha información que puede ayudar al proceso. Este problema puede solucionarse mediante preprocesado de la información. Pero implica un retraso adicional que penaliza la usabilidad del sistema.

Por este motivo, estos algoritmos no son usados cuando el conjunto de datos es muy grande y el sistema de visualización es interactivo, ya que el sistema dejaría de ser usable, en los términos actuales. Para sistemas interactivos donde el usuario espera en tiempo real una respuesta, se usan otros algoritmos, como MarkerCluster (Leaver 2012) o SuperCluster (Agafonkin 2016), creados para plataformas web de visualización de datos en mapas. Estos algoritmos son capaces de agrupar datos en tiempo real en conjuntos de medio millón de datos. En las pruebas realizadas con el algoritmo SuperCluster el autor indica que es capaz de agrupar 400.000 datos, en formato GeoJSON (Butler, H., et al. 2016), en menos de un segundo.

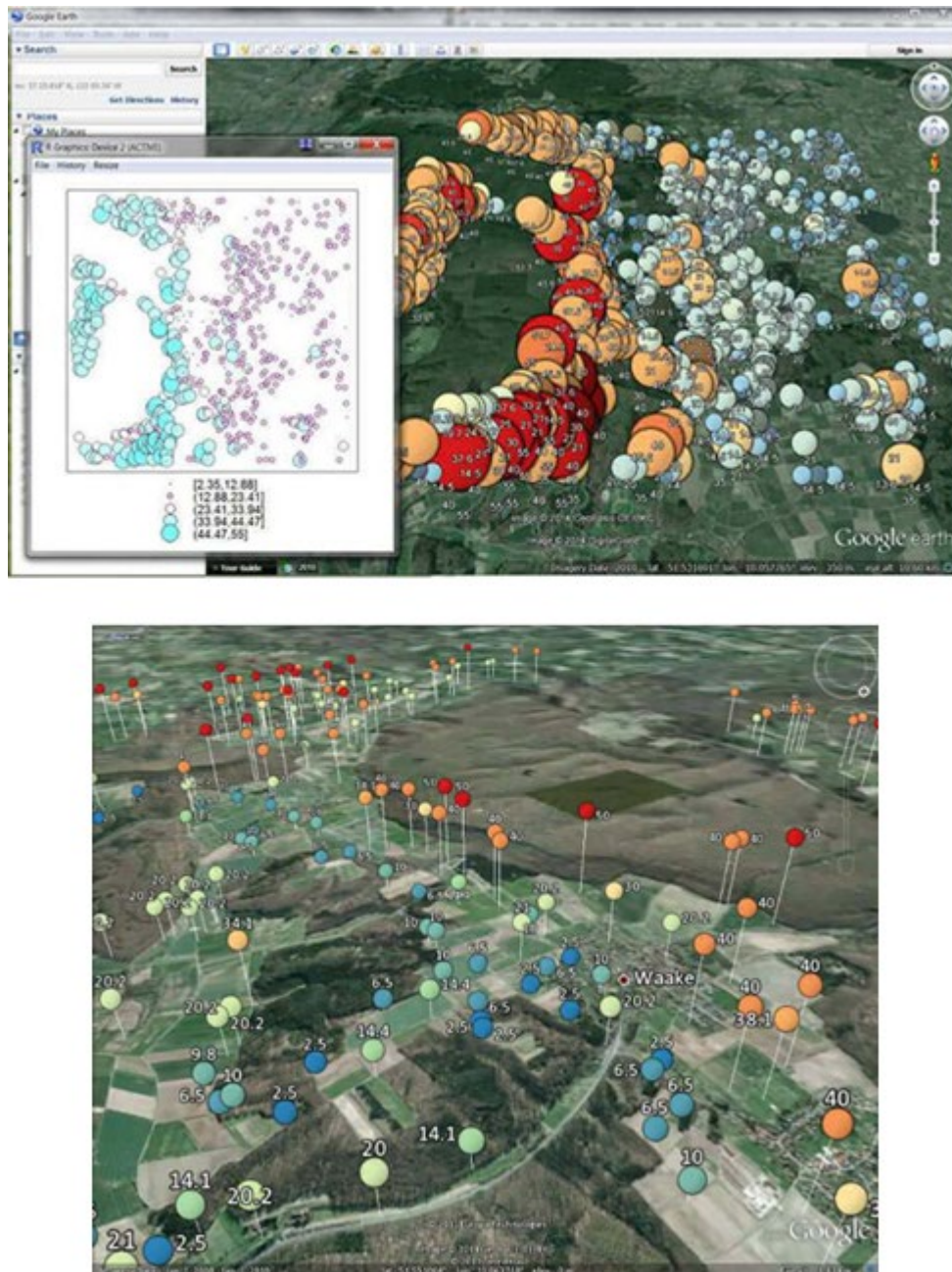
#### *4.2.3. Visualización de relaciones entre objetos*

Visualizar las relaciones es otra dificultad añadida, si también se desea visualizar su fluctuación en el espacio y en el tiempo. De hecho, es una de las actividades más complejas en la visualización de este tipo de datos. En primer lugar, no se pueden obtener resultados similares de los trabajos realizados en la visualización de grafos, o árboles, al menos no de una forma general. En estos trabajos, generalmente los valores son fijos y no fluctúan en función de los valores de otras variables, como el espacio, o el tiempo.

A pesar de la dificultad, existen varios trabajos que permiten visualizar objetos que están relacionados en sistemas de visualización interactivos de datos espacio temporales.

- La forma clásica de representar objetos relacionados es dibujar una línea entre ellos. Un ejemplo de este tipo de representación lo tenemos en los mapas de flujo (Guo 2009). Esta línea puede ser de varios colores y diferentes densidades, con el fin de indicar más información sobre la relación (Scheepens, Willems et al. 2011).
- Otra forma es mediante la representación de los objetos relacionados con una marca gráfica característica (color, grosor, un objeto gráfico, etc.) (Zheng, K., Gu, D., Fang, F., Wang, Y., Liu, H., Zhao, W., Zhang, M., and Li, Q. 2017).
- Combinar las soluciones anteriores con técnicas de agrupamiento, comentadas en la sección anterior. Esta solución suele dar información a nivel global, es decir, no muestra objetos relacionado con uno seleccionado interactivamente, suelen mostrar los datos que tiene un mismo valor agrupados y con un marcador gráfico que los diferencia. Un ejemplo de esta forma de representación (Tomislav Hengl, Pierre Roudier et al. 2015) se encuentra representado en la Figura 9.





*Figura 9 Ejemplo de combinación de mostrar relaciones con técnicas de agrupamiento. Arriba, burbujas de mismo color están relacionadas. Debajo, igual, pero con más elementos gráficos que ayudan a evitar problemas de sobre dibujado. Fuente: (Tomislav Hengl, Pierre Roudier et al. 2015)*

El problema principal que tiene mostrar la información de los datos que están relacionados, es que con demasiada frecuencia se suele representar demasiada información, lo que hace muy compleja la tarea de detectar patrones, o comportamientos fuera de lo común en estas relaciones. Sobre todo, porque en ocasiones, los indicadores de la relacionen terminan por cubrir visualmente una gran parte de los datos, haciendo imposible ver los datos (Andrienko, G., Andrienko, N., Bak, P., Keim, D., Wrobel, S. 2013).

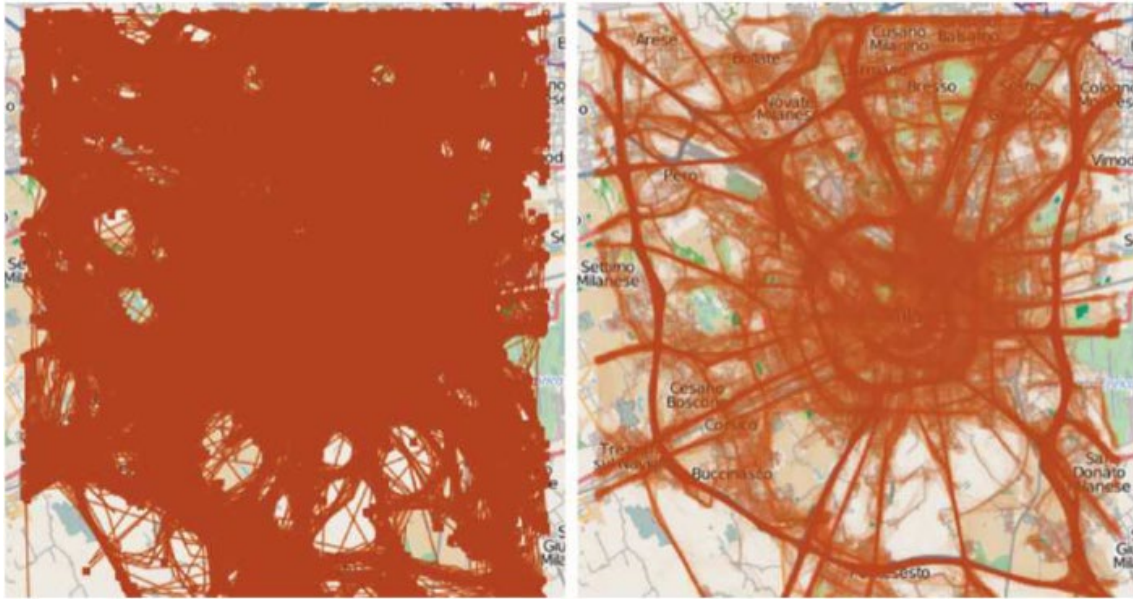


Figura 10 A la izquierda, ejemplo de efectos del sobre dibujado en mapas de trayectoria. A la derecha, solución con reducción de la opacidad al 5%. Fuente:(Andrienko, G., Andrienko, N., Bak, P., Keim, D., Wrobel, S. 2013)

#### 4.2.4. Técnicas de visualización interactivas para la selección del tiempo.

Hasta este momento se han expuesto técnicas de visualización de datos interactivas teniendo en cuenta sobre todo la variable de espacio. En esta sección se analiza la combinación de las técnicas anteriores, pero incluyendo la variable tiempo.

Para analizar cómo fluctúan en el tiempo los datos representados espacialmente existen dos grupos de técnicas. Por una parte, están las técnicas que permiten determinar un momento concreto de tiempo, o un intervalo de tiempo, y el sistema muestra los datos relacionados con ese intervalo de tiempo definido. Por otra parte, existen las técnicas que permiten definir varios intervalos de tiempo y ver al mismo tiempo los datos relacionados en varios intervalos de tiempo.

La primera opción es la más usada, y para facilitar la selección del intervalo de tiempo se han desarrollado avanzadas interfaces de usuario con innovadores controles para poder seleccionar el intervalo de tiempo. Una de las soluciones más utilizadas es la línea de tiempo (Kraak 2005, Lee, Devillers et al. 2014) Son controles generalmente lineales, donde se puede seleccionar un fragmento de forma interactiva. Los más avanzados suelen ser jerárquicos, con varias franjas horizontales, que dividen el fragmento superior en varias zonas y que permiten navegar por la línea de tiempo de forma más rápida e intuitiva (Wang, C., Ma, and Chen 2018b). Existen algunos controles que permiten definir incluso meses y días, de forma visual, como el control Time Wheel (Andrienko, Natalia, Andrienko et al. 2003). En la Figura 11 se observa una herramienta que permite definir los hallazgos arqueológicos en una zona, mediante tres líneas de tiempo. La superior permite acotar el periodo en años al que pertenecen las piezas, la media el periodo cultural y la inferior el momento en el que se hicieron las excavaciones. En la Figura 12 se muestra una captura de una aplicación que permite visualizar zonas donde se han encontrado fósiles de un periodo prehistórico. En esta aplicación se muestra una línea de tiempo interactiva que permite seleccionar un periodo de forma sencilla. La parte superior de la línea muestra un periodo, que se subdivide en tres periodos, cada uno de ellos a su vez, se subdivide en más



periodos. Al seleccionar uno de estos periodos inferiores, se posiciona en la parte superior y permite al usuario seguir navegando. Esta línea permite localizar un periodo en pocos segundos.

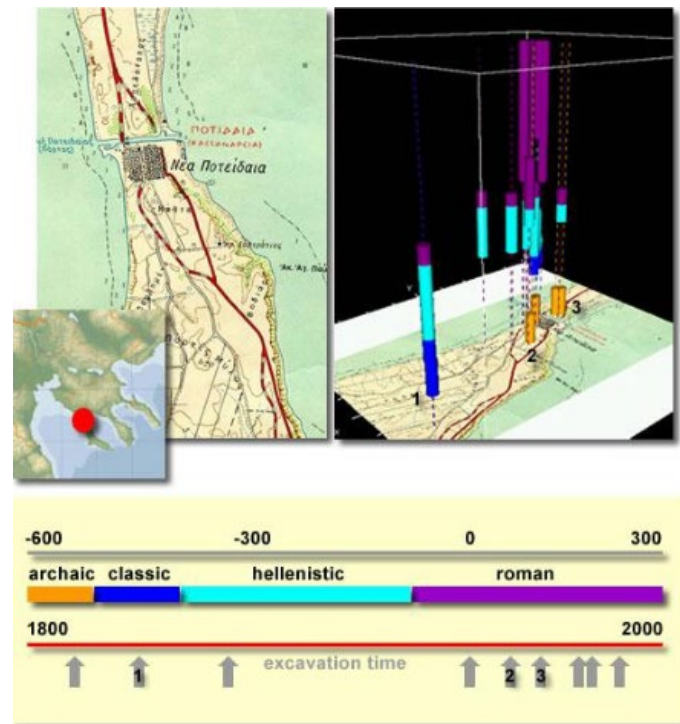


Figura 11 Captura de aplicación con línea de tiempo para visualizar hallazgos arqueológicos. Fuente: (Kraak 2005 )

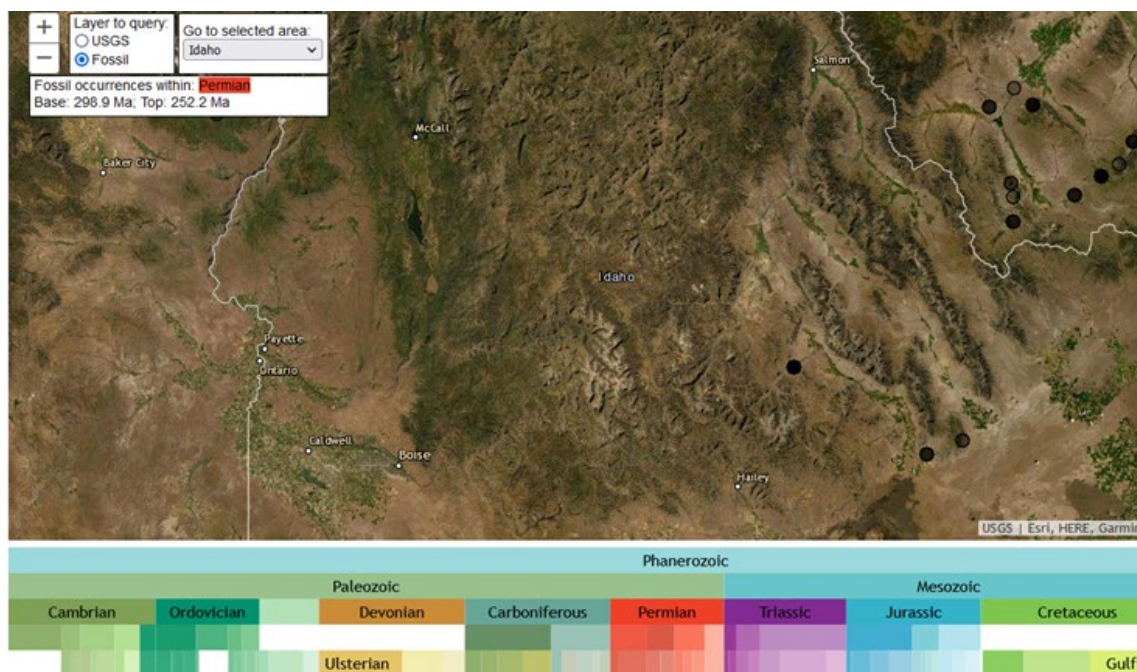


Figura 12 Captura de aplicación con línea de tiempo para visualizar hallazgos de fósiles de un periodo. Fuente: (Wang, C., Ma, and Chen 2018b)

La segunda opción, que permite ver, de forma simultánea, el estado de los datos en diferentes momentos de tiempo es más compleja, tanto en la selección, como en la visualización, sobre todo si es simultánea. Por una parte, se ha de poder determinar qué intervalos de tiempo se desean analizar y por otra, generar técnicas que permitan ver la información de cada momento seleccionado.

Aunque pueden existir situaciones que pueden requerir contrastar dos momentos de tiempo diferentes, lo habitual es analizar la evolución de periodos de tiempo continuos, o sucesivos. Por ello, para visualizar datos en diferentes momentos de tiempo, generalmente se ofrecen interfaces para determinar un punto inicial y un punto final, y si tiene sentido, definir un intervalo de tiempo. De esta forma se ofrecen tantos conjuntos de datos como intervalos haya entre los extremos.

La forma de mostrar la información es diversa, las técnicas más usadas son las basadas en visualizaciones bidimensionales. Donde los datos de cada intervalo pueden reducirse y verse todos de forma continua, o construir fotogramas y ver como una película que muestra la evolución de los datos conforme fluctúa el tiempo, creando el denominado por Andrienko (Andrienko, N., Andrienko et al. 2000) como “Ventana de tiempo”. En la Figura 13 se aprecia una captura de aplicación que utiliza estas técnicas para ver la incidencia de cáncer en la ciudad de Taiwan (Ku, Liaw et al. 2016).

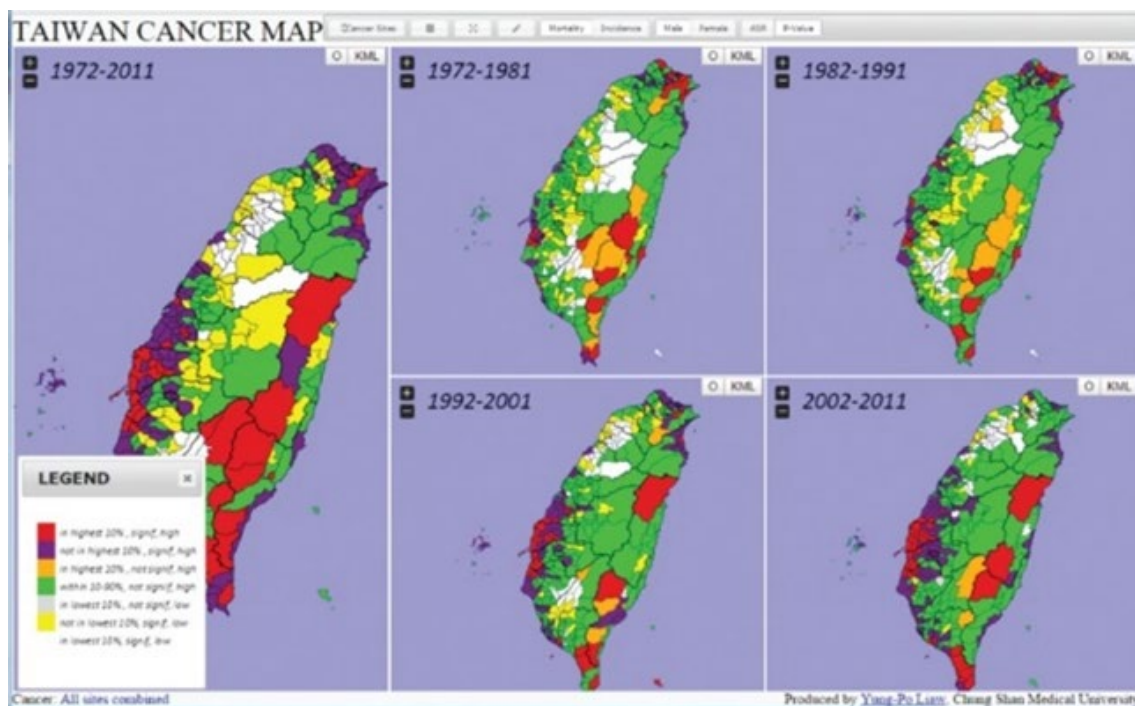


Figura 13 Efecto ventana de tiempo. El mismo escenario, pero con datos relativos a diferentes intervalos de tiempo. Fuente: (Ku, Liaw et al. 2016).

Otras formas de mostrar cómo fluctúa la información en el tiempo es usar visualizaciones en entornos tridimensionales. Esta opción, a priori parece ser mejor, ya que mostrar estos datos de forma simultánea implica visualizar más información, y parece lógico que una dimensión adicional pueda facilitar la visualización. Una solución desarrollada es el uso de los llamados cubos espacio temporales (Bach, Benjamin, Dragicevic et al. Jun 11, 2014). Un cubo espacio temporal representa diferentes capas bidimensionales con toda la información de cada intervalo, una encima de la otra, con la suficiente distancia para poder ver el contenido. Esta opción aporta mucho, siempre que la interacción para poder ver el contenido sea la adecuada (Windhager F., Mayr E., Schreder, G., Smuc M., Federico, P. and Miksch, S. 2016) . La navegación libre por entornos tridimensionales no suele ser muy usable en usuarios sin experiencia en estos entornos. Si la visualización es estática, es complejo ver el contenido si hay varias capas. Esta técnica también se usa para ver la alteración del entorno en el tiempo, por ejemplo, construcción, o destrucción de edificios (Bogucka, Jahnke 2018). También hay trabajos

realizados, sobre todo cuando hay muchos intervalos de tiempo, que usan marcadores gráficos compuestos por espirales tridimensionales (Hewagamage, Hirakawa et al. 1999), donde en cada vuelta de la espiral se muestran los datos en ese punto concreto. Esta opción tiene el inconveniente de que es muy difícil comparar datos que estén distantes en el espacio. En la se Figura 14 muestran dos ejemplos de aplicaciones que usan cubos espacio temporales.

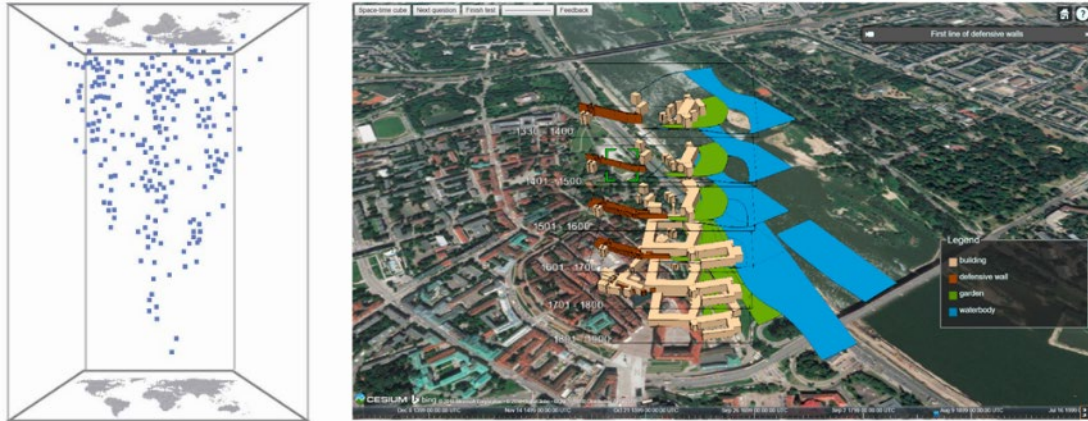


Figura 14 Izquierda, ejemplo de cubo espacio temporal geoespacial. Izquierda, Fuente: (Bach, Benjamin, Dragicevic et al. Jun 11, 2014).. Derecha, captura de aplicación que usa un cubo espacio temporal para ver cambios en edificios. Fuente: (Windhager F., Mayr E., Schreder, G., Smuc M., Federico, P. and Miksch, S. 2016)

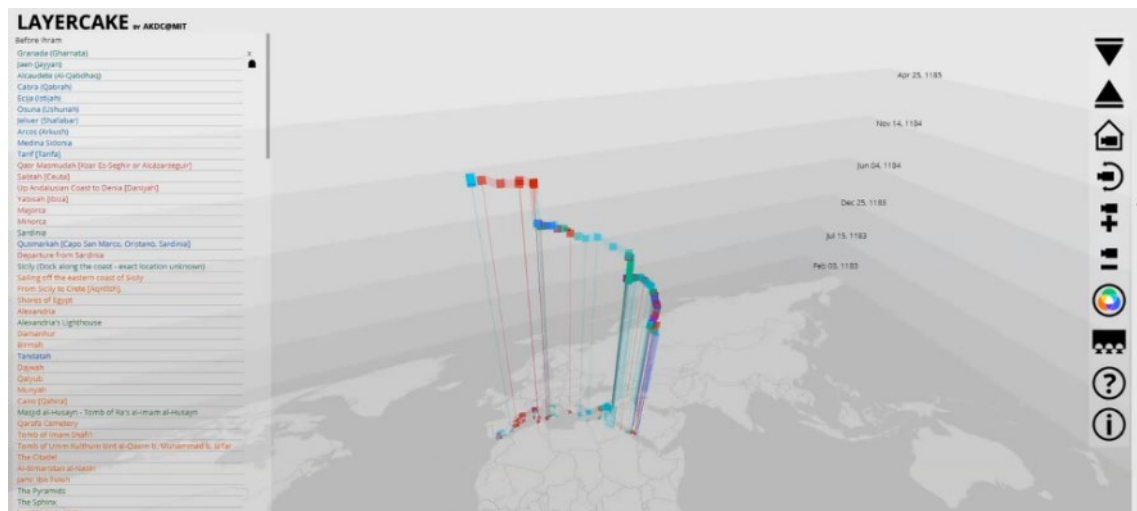


Figura 15 Captura de pantalla de la aplicación LayerCake. Fuente: (Aga Khan Documentation Center at, MIT 2019)

También existen sistemas de visualización que permiten visualizar trayectorias de forma tridimensional. En el entorno tridimensional, las coordenadas x, e y, son las espaciales, generalmente longitud y latitud, o derivadas, y la coordenada z, o altura, viene determinada por el tiempo, de forma que la trayectoria siempre es ascendente. Esta solución es muy similar a la de un cubo espacio temporal, pero aplicando a visualizar trayectorias. La herramienta LayerCake (Aga Khan Documentation Center at, MIT 2019) utiliza está técnica. Visualizar las trayectorias de esta forma es muy espectacular visualmente, y se usa para reflejar la rapidez, o lentitud de movimiento de un objeto en el tiempo. El principal problema que tienen es que su aplicación está limitada a muy pocos objetos. Si se usa esta técnica con varias trayectorias es muy sencillo



llegar a situaciones de sobre dibujado, siendo imposible comprender el contenido. En la Figura 15 se muestra una captura de pantalla de la herramienta LayerCake.

#### 4.3. APLICACIÓN DE ONTOLOGÍAS EN SISTEMAS DE VISUALIZACIÓN.

Generalmente, las herramientas de Big Data procesan cantidades ingentes de información sin la necesidad de disponer de datos con un alto componente semántico. Por ejemplo, dentro del abanico de posibilidades de aplicaciones de Big Data, existen las herramientas de procesamiento de imagen. Estas aplicaciones, analizan el contenido de imágenes mediante soluciones basadas en redes neuronales, buscando patrones en una imagen, tomando como referencia un conjunto de imágenes relativamente grande y correctamente clasificadas (Zerdoumi, Sabri et al. 2018). Probablemente, si durante el procesamiento de las herramientas de Big Data, se tuviera acceso a información semántica de los objetos, el procesamiento sería más rápido. Por ejemplo, a la hora de procesar una imagen de un tejido, si se pueden conocer los motivos que aparecen, que técnicas, o que materiales se usaron para elaborarlo, se podría evitar procesamiento adicional, porque muchos de los datos ya son conocidos, o sirven de ayuda para clasificarla.

Las tecnologías relacionadas con la web semántica y en concreto las ontologías, se usan desde hace décadas para representar complejos modelos de información e inferir conocimiento. Y parece lógico deducir, que deberían de utilizarse junto a las herramientas Big Data, para obtener mejores resultados de forma más eficiente. Sin embargo, el Ontology Summit de 2014 (Gruninger, M., Obrst, L., Kutz, O., Lange, C., Levenchuk, A., Quattri, F., Schneider, T., Spero, S., Thessen, A., Vegetti, M., Vizedom, A. 2021) se manifestó que aunque las ontologías habían tenido un papel clave en las tecnologías de la Web Semántica, sus aplicaciones en nuevas líneas, como la línea de investigación de Big Data, es menor que el esperado. Parece lógico pensar que ambas líneas de investigación hubieran mantenido una mayor relación, pero no ha sido así. En el Ontology Summit de 2014, también propusieron una serie de medidas para incrementar la aplicación de las ontologías en este campo, que han dado buenos resultados en los siguientes años, tal y como quedó reflejado en el Ontology Summit de 2018 (Baclawski, Bennett et al. 2018). En esos cuatro años se habían desarrollado varios proyectos que combinaban las dos líneas, y había varios operativos (Bennett, Baclawski 2017, Konys 2017, Dou, Wang et al. 2015), además de mejorar y evolucionar las herramientas que permiten el acceso al contenido de una ontología (Ong, Xiang et al. 2017, Verhodubs 2017). Si el impulso se mantiene, todo apunta a que una gran parte de la información que maneja la tecnología de Big Data estará soportada por un grafo de conocimiento. Con ello, los avances realizados sobre visualización de ontologías deberían de ser muy útiles para poder visualizar y analizar la información de estos datos resultantes de unir estas dos líneas de investigación. Sin embargo, los principales avances en el campo de visualización de grafos de conocimiento están basados en la visualización de la estructura, y navegar por datos, mediante técnicas de representación de grafos, treemaps, formatos radiales y otras técnicas (Sawsaa, Lu 2014, Lohmann, Link et al. 2015, Dudáš, Lohmann et al. 2018). Los avances realizados en esta línea no tienen generalmente el objetivo de analizar el contenido y descubrir patrones, o relaciones. Por lo tanto, existe una necesidad de poder combinar los avances de visualización de datos, con grafos de conocimiento, pero aprovechando el conocimiento y la estructura del grafo.

En la Figura 16 se muestra una visualización de una ontología mediante WebOWL, una herramienta web que representa el contenido de una ontología de forma interactiva con técnicas de representación de grafos.

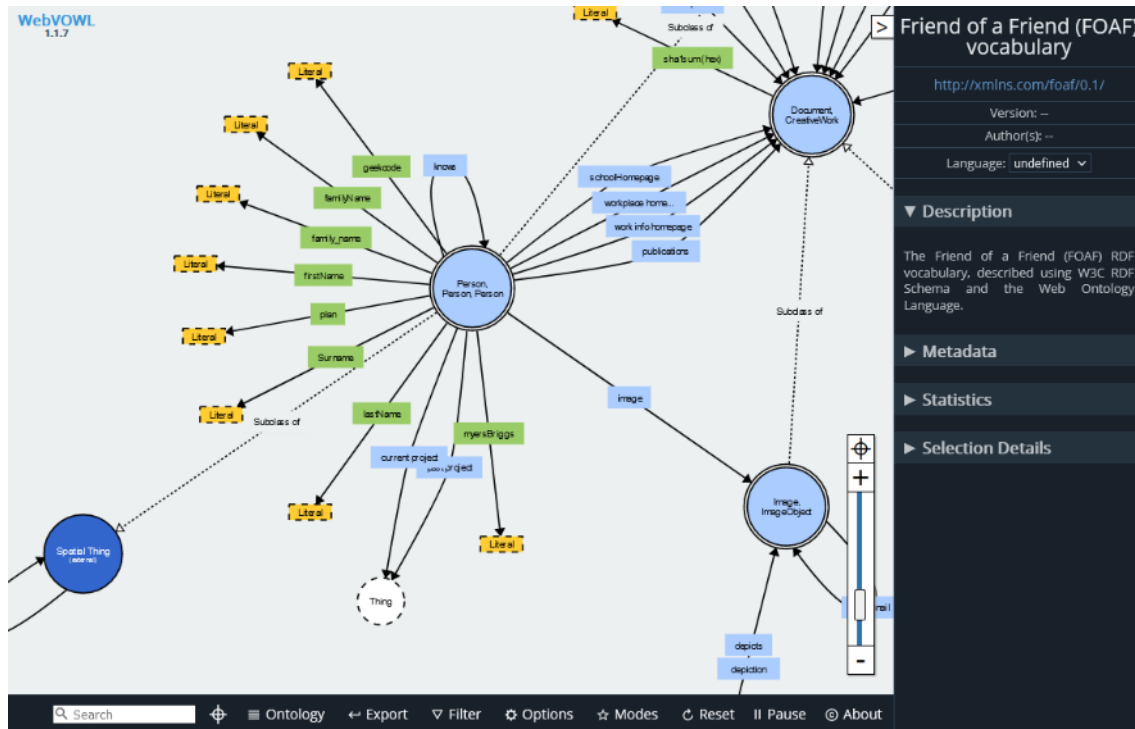


Figura 16 Captura de pantalla de WebOWL, visor de ontologías online. Fuente:(Lohmann, Link et al. 2015)

En la Figura 17 se muestra una combinación de técnicas para representar una ontología, por una parte, una representación 2.5D para y por otra, otra basada en histogramas.

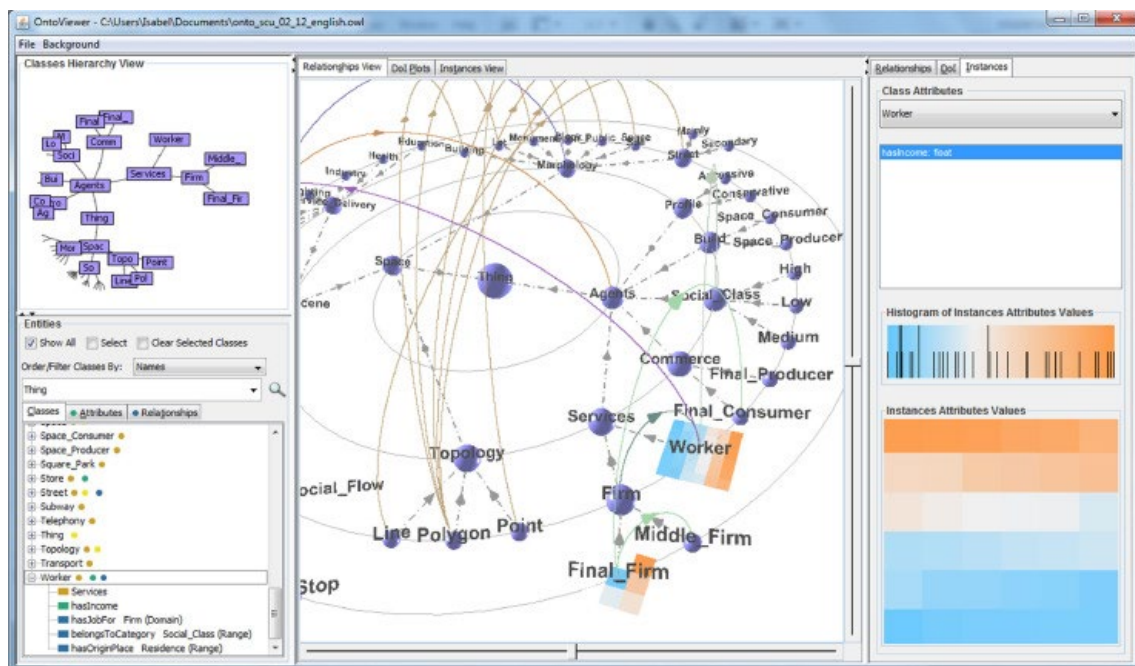


Figura 17 Visualización de ontologías con combinación de técnicas 2.5D con histogramas. Fuente: (Silva, Santucci et al. 2019)

Además, dado que estas herramientas van a ser capaces de procesar ontologías, la definición de cómo se ha de visualizar la información, debería de estar en una ontología. De no hacerse así, los sistemas de visualización seguirán siendo aplicaciones dirigidas, que usarán el contenido de un grafo de conocimiento, como una base de datos, solicitando la información de forma prefijada y visualizándola en trabajos ad hoc. No existiendo diferencia, al menos para este objetivo, entre una ontología, o una base de datos.

#### *4.3.1. Ontologías de visualización.*

Aunque no hay muchos trabajos, hay algunos autores que han detectado esta necesidad y en la última década se han publicado algunas propuestas de utilizar únicamente ontologías para definir cómo se ha de visualizar interactivamente el contenido de una ontología. Los primeros trabajos en este ámbito tienen como objetivo definir una escena en 3D mediante una ontología (Bolívar, H., González, R., Sanjuán, O. 2009, Filho, Vieira, M. d. F. Q. 2014), más adelante se han publicado trabajos más ambiciosos que incluyen ontologías de visualización Visualization Ontology (VISO) (Polowinski, Jan, Voigt 2013) y Visualization Urban Mobility Ontology (VUMO) (Sobral, T., Galvão et al. 2016), además de otros trabajos en los que se manifiesta esta necesidad (Nazemi, Burkhardt et al. 2015, Polowinski, Jan, Voigt 2013, Falconer, Bull et al. 2009) .

#### *Definición de escenas 3D mediante una ontología*

En la primera década de este siglo, la representación gráfica en entornos 3D, hasta hace pocos años limitada a equipos informáticos de alto coste se volvió más accesible. Esta situación, junto a la popularidad de uso de internet, la mejora en las conexiones y en los navegadores web, ha motivado que comenzará en esos años una tendencia, cada vez mayor, a visualizar entornos tridimensionales en plataformas web. Esto último se reforzó con la aparición de APIs (Application Programmin Interface) en lenguajes populares, orientados a este tipo de representación gráfica, como es el caso de Java 3D (Sun Microsystems 2000), que además permitía visualizar estos entornos en plataformas web, también por la publicación de nuevos estándares para definir y representar este tipo de escenarios gráficos en la web, como X3D (Extensible 3D) (Web3D Consortium 2013).

Todo esto unido al auge de los lenguajes de especificación de ontologías, motivó la aparición de trabajos orientados a representar escenarios en 3D mediante una ontología.

Su objetivo no era únicamente representar el aspecto gráfico y los objetos a visualizar, sino también modelizar las características de interacción de la escena, en función de las posibilidades que ofrecían los medios de desarrollo del momento, principalmente asociadas a eventos de movimiento, e interacción con los objetos de la escena.

Las ontologías que seguían este tipo de filosofía, generalmente su definición era casi paralela a la definición del API (Bolívar, H., González, R., Sanjuán, O. 2009). Esto era posible porque la API estaba totalmente orientada a la definición de escenas tridimensionales. Eran muy útiles para definir qué objetos ser representaban en la escena y cómo eran sus características. Lo mismo para cualidades generales de la escena, como la iluminación, el sonido, etc. Un tema más complejo era incluir en la tecnología lo que este tipo de API denominaban Behavior (Oracle 2008), o comportamientos, que asociaban a objetos una serie de acciones, generalmente programas y definidas por desarrolladores.

Sin embargo, en la última década estas tecnologías quedaron obsoletas y debido a que las nuevas tecnologías son más abiertas y no están totalmente dirigidas a la definición de escenarios 3D, no surgieron más ontologías similares, pero adaptadas a las nuevas tecnologías.

### Ontología VISO

El objetivo de la ontología de visualización VISO formaliza como visualizar el conocimiento, pretendiendo ser un base para próximos sistemas de visualización de grafos de conocimiento. Esta ontología está dividida en siete módulos:

- GRÁFICO, que formaliza términos relativos al dominio de la representación gráfica. Contiene conceptos como Graphic Representation, que hace referencia al resultado de visualización, con conceptos descendientes como Map, o Visual Structures (que hace referencia a representaciones más concretas como Pie\_Chart (Gráficos de tarta), o Cone\_Trees (Árboles). Otro concepto clave es Grapic Object, con extensiones para representar objetos y sus relaciones.
- DATA, es la encargada de representar los tipos de datos de las variables y estructuras.
- ACTIVITY, comprende los conceptos relacionados con los aspectos humanos de la visualización, desde tareas, acciones y operaciones.
- SYSTEM, contiene los conceptos para representar el contexto de visualización a nivel de hardware y software.
- DOMAIN y USER es la parte del modelo diseñada para formalizar hechos y restricciones del dominio específico.
- FACTS, permite la formalización de restricciones y ponderaciones sobre elementos del modelo, como relaciones gráficas.

El creador de esta ontología, también ha desarrollado el lenguaje RDFS/OWL Visualization Language (RVL) (Polowinski, Jan Jun 12, 2013). Es un lenguaje declarativo para la visualización de los datos de una ontología. Este lenguaje permite definir un mapeo entre clases, relaciones, e instancias con determinados valores, con conceptos y atributos de la ontología VISO. De esta forma se puede mapear las instancias de una clase que tengan un valor concreto en una propiedad, con una entidad gráfica de VISO. Por ejemplo, las instancias con un valor, se podría mapear con un círculo que es un concepto básico que extiende a GraphicObject, con un determinado color, y si tienen otro valor, se podrían asociar con el mismo concepto, pero con otro color. De esta forma podríamos visualizar, las instancias de ese concepto como círculos con dos colores.

En la Figura 18 se exponen dos ejemplos de visualización, a la izquierda visualización de instancias de una ontología de aminoácidos, mostrando las instancias con una imagen u otra, en función de diferentes valores de sus atributos. A la derecha, un sistema interactivo de navegación y estructura por círculos, con datos de una ontología de plantas.

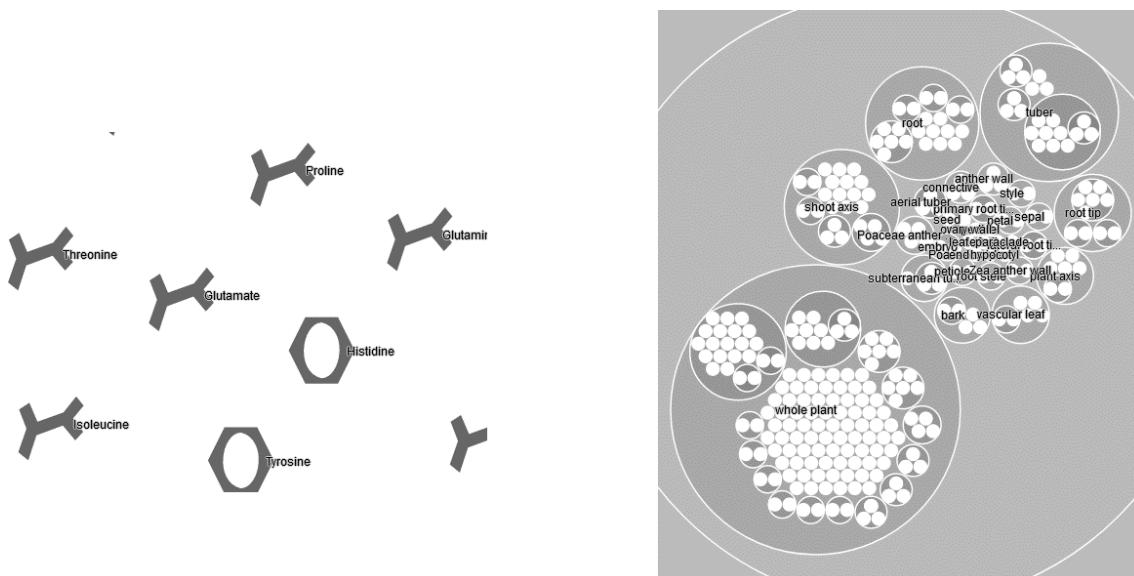


Figura 18 Ejemplos de visualización con VISO y VRL. A la izquierda sobre una ontología de aminoácidos, y a la derecha una ontología de plantas. Fuente: (Polowinski, J. 2015)

El proyecto VISO ofrece mucha documentación en su página web (Polowinski, J. 2015), además de tutoriales y ejemplos de cómo usar VRL con un visualizador que usa la D3.js (Bostock, M., Ogievetsky, V., Heer, J. 2011) como motor de visualización.

El principal problema de este proyecto es el lenguaje VRL. Este lenguaje realmente está construido con OWL y con conexiones a otros lenguajes como SPIN. Tiene una complejidad importante y difícil mantenimiento. Además de que, para su uso, se requiere un conocimiento muy alto de la ontología de dominio para poder hacer visualizaciones de calidad. Esto es así porque la visualización se consigue mapeando conceptos de la ontología de dominio, con conceptos de la ontología VISO, alterando propiedades en función de los valores en atributos y relaciones.

Otro problema adicional, pero menos complejo de solucionar, es que la ontología VISO, los elementos gráficos que existen dentro del módulo GRAPHIC, aunque son genéricos y en principio independientes de las dimensiones del entorno, están centrados principalmente en entornos bidimensionales. Esto implica que si se desea hacer una visualización tridimensional se requiera la extensión y modificación de algunos conceptos, así como la creación de nuevas clases.

#### Ontología VUMO

Visualization Urban Mobility Ontology (VUMO) (Sobral, T., Galvão et al. 2016, Sobral, Thiago, Galvão et al. 2020) es una ontología de visualización dirigida a la visualización semiautomática de la movilidad urbana.

La ontología está compuesta de dos módulos principales:

- Módulo 1. En este módulo se han modelado los conceptos relacionados con la movilidad urbana (tráfico, tickets, flujos de circulación, etc.)
- Módulo 2. Este módulo formaliza conceptos de visualización.

El módulo 2, que contiene los conceptos de visualización permite definir un marco de referencia geográfico, o abstracto, además de definir el tiempo y las dimensiones sobre las que se va a visualizar la información (2D y 3D). En lugar de abstraer diferentes conceptos de visualización,



como escenarios, objetos básicos, etc. En la ontología VUMO se definen dos escenarios de visualización HeatMap y CalendarHeatMap y dos niveles de interacción, uno para zoom (Zoom) y otro para realizar filtros (Filter)

Los autores de la ontología proponen que una vez se haya diseñado un sistema de visualización capaz de visualizar el contenido de los datos expresados en VUMO, sólo habrá que desarrollar dos tareas:

- Mapear datos de cualquier ontología con conceptos de movilidad urbana a los conceptos del Módulo 1.
- Invocar su sistema de recomendación, que en función de la funcionalidad que el usuario final desea realizar, permitirá generar los datos necesarios en VUMO. De esta forma, el sistema de visualización únicamente tiene que procesarlos y visualizarlos. Al hacerlo, gracias al sistema de recomendación, los datos serán los más adecuados para la funcionalidad requerida por el usuario. Aunque si desea realizar más visualizaciones que los definidos en VUMO tendrá que extender la ontología para incluirlos.

En la Figura 19 se muestran unos esquemas de las dos escenas de visualización definidos en el módulo 2 de la ontología VUMO.

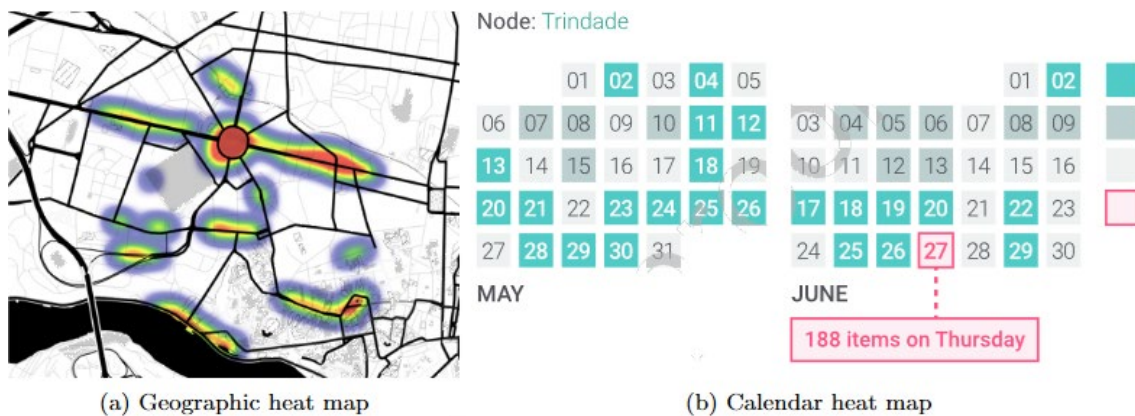


Figura 19 Esquemas de los prototipos de visualización definidos por VUMO. Fuente: (Sobral, Thiago, Galvão et al. 2020)

En este sistema, para visualizar la información es necesario realizar un mapeo del dominio de datos con la ontología VUMO. Si este mapeo se realiza de forma adecuada, se pueden obtener resultados cercanos a las necesidades del usuario de forma automatizada. El problema que tiene generar nuevas visualizaciones es que es un proceso complejo y costoso. Crear una nueva visualización, implica definirla en la ontología, e implementarla en el sistema de visualización. Además, habría que adaptar el sistema de recomendación, que procesa los datos, ya que está muy ligado a las visualizaciones definidas en la ontología. Debido a esto, una nueva visualización conllevaría adaptar, o extender, el sistema de recomendación. En esta ontología no se contemplan aspectos fundamentales de las escenas principales, que tendrían que hacerse desde el sistema de visualización, como por ejemplo decidir cómo representar diferentes tipos de datos, o como representar las relaciones entre ellos, o si se pueden hacer filtros y sobre qué, o mostrar los mapas de calor mediante otra técnica, como clusters numéricos, etc.

Esta situación tiene la dificultad añadida de que no hay ninguna implementación del sistema de visualización. Por lo que habría que implementarlo, o esperar a que la comunidad lo desarrolle, a ser posible en formato de código abierto para poder ampliarlo.

#### 4.4. DESARROLLOS WEB PARA LA VISUALIZACIÓN DE DATOS DEL PATRIMONIO CULTURAL

En esta sección se expondrán una selección de aplicaciones de visualización de datos espacio temporales de patrimonio cultural. Todas las aplicaciones expuestas son accesibles desde una plataforma web y son de acceso público.

Y todas ellas, aunque sea a nivel de objetivo y funcionalidad último, están alineadas con los desarrollos de esta investigación. Se ha creído interesante tenerlas en cuenta, por analizar el estado del arte en este tipo de soluciones y en el mismo ámbito y plataforma.

Además de describirla, se considerarán los siguientes aspectos:

- Número de datos.
- Si utiliza técnicas de agrupamiento, y cómo.
- Si usa líneas de tiempo.
- Si permite ver relaciones entre los objetos.
- Si tiene un nivel de interacción a nivel de navegabilidad, zoom y filtrado.
- Si los datos se procesan a partir de un grafo de conocimiento.
- Si se trata de un sistema abierto.
- Si permite visualizar la fluctuación de datos en el tiempo y cómo, un momento concreto, o varios de forma simultánea.

##### 4.4.1. *The Museum of the World.*

El Museo Británico (The Trustees of the British Museum 2021), junto al Google Cultural Institute (Google 2021b), ha desarrollado una plataforma web para visualizar un conjunto seleccionado de sus colecciones, “The Museum of the World” (British Museum, Google Cultural Institute 2015). El objetivo del proyecto es conseguir que el conocimiento y la cultura de la humanidad estén disponibles de forma libre.

El proyecto presenta objetos de todo el mundo y a lo largo del tiempo con una interfaz innovadora, muy similar al mástil de una guitarra. En la Figura 20 se muestra una captura de pantalla donde se puede apreciar el aspecto de la interfaz. Cada pista hace referencia a un continente, que tiene un color concreto al principio, color que se mantiene en todos los puntos que están ubicados en ese continente.

Conforme avanza la pista, cada bloque de trastes, si mantenemos el símil del mástil de guitarra, se corresponde con un siglo. La interfaz de la aplicación permite:

- Desplazarse de forma interactiva, mediante botones en la interfaz, el ratón y el teclado, hacia delante y hacia atrás en el tiempo, moviendo el punto de vista del observador, sobre las pistas.
- Si se selecciona un objeto, aparece una ventana flotante con información del objeto, como el nombre, una fotografía, aunque sería deseable contar con datos fundamentales, como la fecha de su descubrimiento y el tipo de objeto que es. Aparece un botón que al seleccionarlo muestra una ventana emergente con toda la información del objeto.
- Filtrar por cinco tipos de objetos mediante un menú situado a la derecha.
- A su vez, al seleccionar un objeto, se muestran unas líneas, que sirven para ayudar a establecer un orden cronológico, no para indicar que están relacionadas. Afectan a todos los objetos de la interfaz.

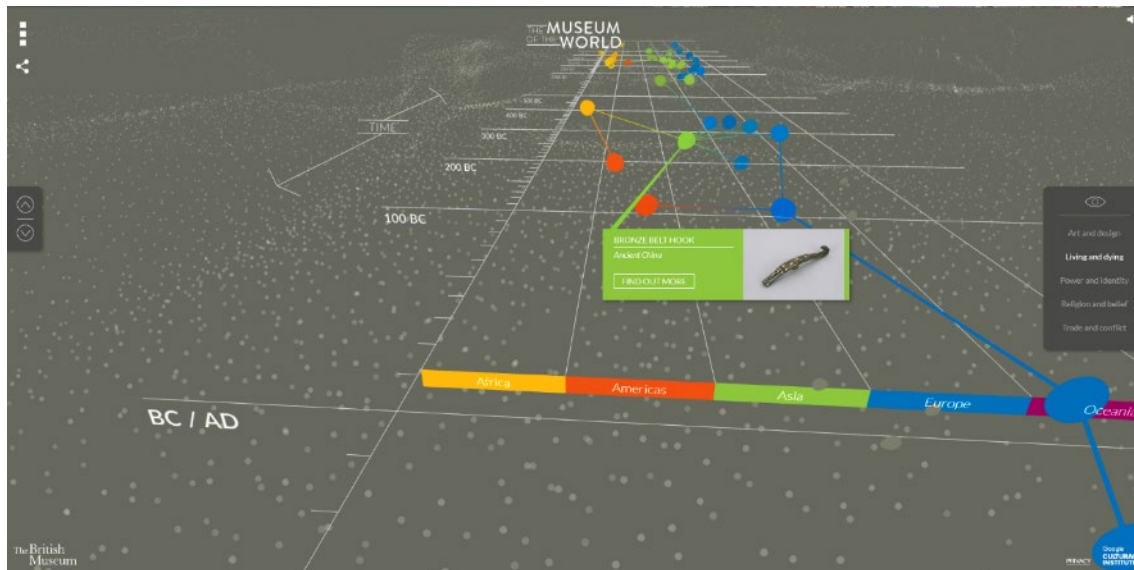


Figura 20 Captura de pantalla de la aplicación *The Museum of the World*. Fuente: (British Museum, Google Cultural Institute 2015)

El conjunto de datos es pequeño, comprende un total de 319 objetos (Lash 2017).

Los resultados finales del proyecto están disponibles mediante una aplicación web, pero el desarrollo del sistema no está publicado en código abierto. No existe una versión pública del software para poder ejecutarlo con otros datos. Tampoco está descrito el API para introducir nuevos datos, ni como enlazar la visualización con el dominio de datos.

La ubicación espacial es muy poco precisa, ya que únicamente relaciona el objeto con el continente. Esto en determinados contextos de uso puede ser una ventaja, pero en otros un inconveniente importante.

#### 4.4.2. Cronobook

Cronobook (Cronobook 2021) es una plataforma web que permite a los usuarios colocar fotos de calles, plazas, etc. Indicando la fecha de cuándo fueron tomadas, además de indicar su posición y una descripción.

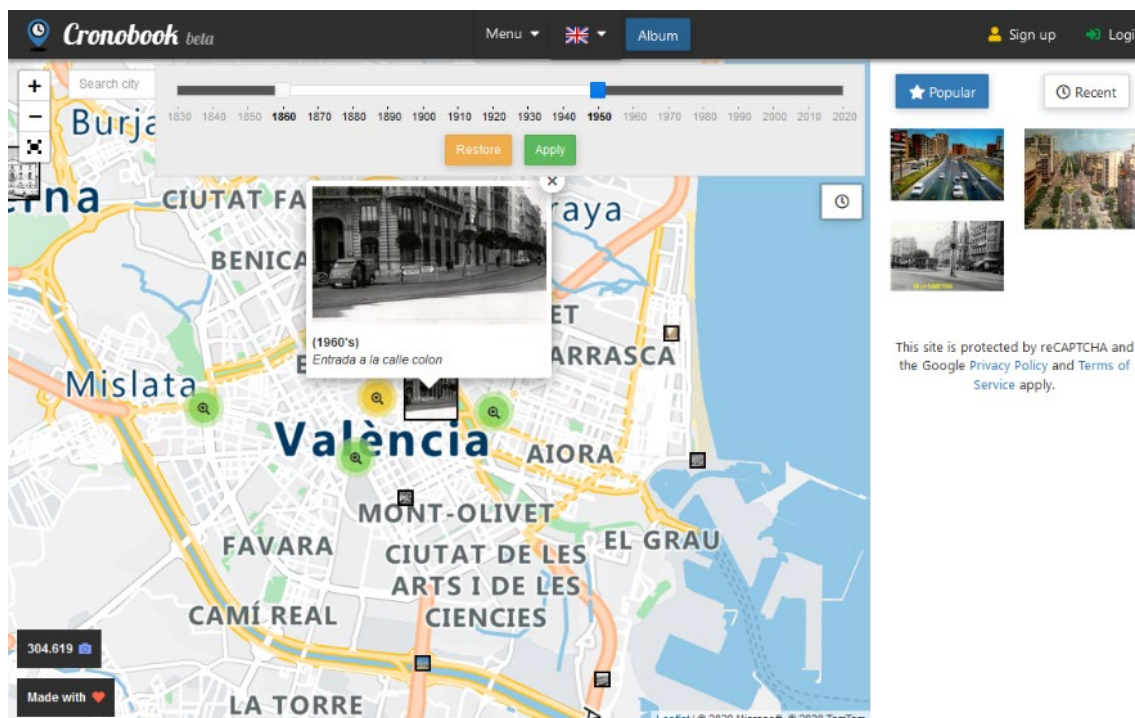
En la actualidad cuenta con 304.619 fotografías. Se trata de un mapa bidimensional, y utilizada la herramienta Leaflet (Agafonkin 2020), con mapas de Mapbox (Mapbox 2021) y OpenStreetMap (OSMF 2021). La herramienta permite navegar por el mapa, utilizando los recursos de la herramienta Leaflet. Organiza los datos en clusters, utilizando el algoritmo de markercluster (Leaver 2012).

Permite realizar varios filtrados sobre los datos, pero no permite ver relaciones entre ellos. El resultado de filtrar los datos se muestra en una ventana aparte, donde aparecen las fotografías en modo collage.

La herramienta muestra una línea de tiempo, que posibilita filtrar los datos asociados a un intervalo de tiempo sobre el mapa. En la Figura 21 puede verse una captura de pantalla de la aplicación web donde se ve el mapa, los clusters, una ventana con la fotografía y datos, así como la línea de tiempo desplegada.

No ofrece ninguna API, y no hay ninguna publicación que indique la tecnología usada para mantener los datos, aunque no parece que la información esté soportada por un grafo de conocimiento.

El sistema mantiene internamente un histórico de las visitas realizadas por los usuarios, permitiendo ver las más populares de la región actual del mapa, así como las recientemente consultadas por el usuario.



*Figura 21 Captura de pantalla de la aplicación Cronobook. Fuente:(Cronobook 2021)*

#### *4.4.3. EAMENA Database*

Endangered Archaeology in the Middle East & North Africa (EAMENA) Database (Zerbini 2018) es una base de datos geoespacial de objetos arqueológicos de Oriente Próximo y del Norte de África. La base de datos maneja más de 338.000 referencias. El sistema está soportado por el software Arches (Getty Conservation Institute, WMF 2021), aplicación multiplataforma, de código abierto, que permite integrar y exportar datos de patrimonio cultural, además de visualizarlos y filtrarlos en una plataforma web.

La aplicación web (Arcadia Fund, EAMENA Project, British Council 2018), muestra inicialmente todos los resultados en una lista de datos paginada, junto a un mapa bidimensional que agrupa los datos en clusters. El agrupamiento no es muy correcto, ya que existe un solapamiento importante de clusters, al agruparse en función del tipo de objetos. En la Figura 22 se muestra una captura de pantalla de la aplicación.

El sistema permite cambiar la opacidad del marcador de los clusters, pero sigue siendo complejo averiguar la cantidad de datos que hay en una zona hasta que no se hace un zoom adecuado.

La aplicación permite filtrar los datos por todos los campos de la base de datos, además de elegir entre mapa satélite, o político.



Los datos están estructurados en un grafo de conocimiento basado en el modelo CIDOC-CRM, y aunque en la página web no se especifica un API, es de asumir que debe de existir, por la filosofía del software Arches en el que está soportado.

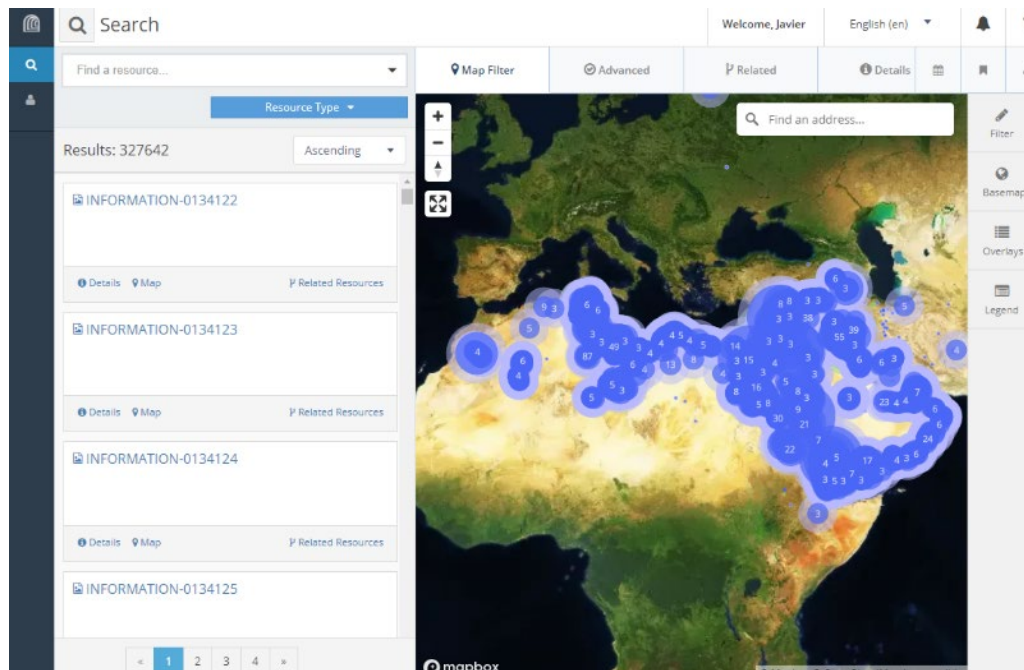


Figura 22 Captura de pantalla del sistema de visualización de EAMENA Database. Fuente: (Arcadia Fund, EAMENA Project, British Council 2018)

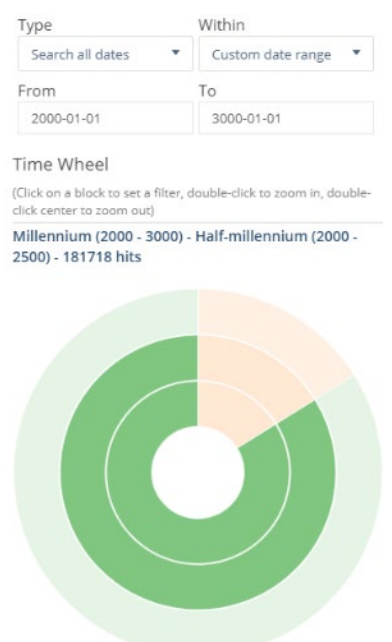


Figura 23 Rueda de tiempo interactiva de EAMENA Database. Fuente: (Arcadia Fund, EAMENA Project, British Council 2018)

Aunque no tiene una línea de tiempo embebida en el mapa, la aplicación consta de dos sistemas para filtrar el tiempo. Por una parte, permite introducir una fecha origen y otra final, por otra parte, ofrece una rueda de tiempo interactiva, con periodos concéntricos, que permite seleccionar un periodo de forma sencilla.

En la Figura 23 se puede ver una captura de pantalla del sistema de selección de tiempo. La aplicación no permite ver datos en varios momentos de tiempo diferente.

El sistema permite mostrar relaciones de los objetos con entidades de la base de datos, pero no con objetos representados en el mapa. Esta información se muestra en una ventana diferente. Por ejemplo, la mayoría de referencias que representan yacimientos arqueológicos tienen asociada una fotografía realizada por la Royal Air Force (RAF). Muchos de estos objetos tienen únicamente una relación con la RAF, ya que es una entidad de la base de datos.

#### 4.4.4. Chronas

Chronas (Dietmar 2021b) es un proyecto creado gracias al crowdfunding organizado por su creador a título personal (Dietmar 2021a). El trabajo tiene fines educativos, consistente en un mapa bidimensional interactivo que permite visualizar todo tipo de marcadores históricos (castillos, batallas, etc.), además de visualizar por áreas las fronteras políticas de las regiones, religiones, población, etc. Todo ello vinculado a un momento del tiempo, definido por una línea de tiempo que interactivamente permite hacerse más grande para poder realizar mejor la selección. En la Figura 24 se muestra una captura de pantalla del proyecto Chronas, en la que se aprecia como filtrar las capas, además de ver diferentes marcadores y definiciones de áreas.

Sobre estos datos, formados por más de 114.000 marcadores, y cerca de 50 millones de definiciones de áreas. Se pueden realizar filtros por varios criterios, además de texto libre. Además, el sistema visualiza en la zona superior, imágenes de los datos relacionados con el criterio activo. Los datos son adquiridos del grafo de conocimiento de Wikidata (Pellissier Tanon, Vrandečić et al. 2016)

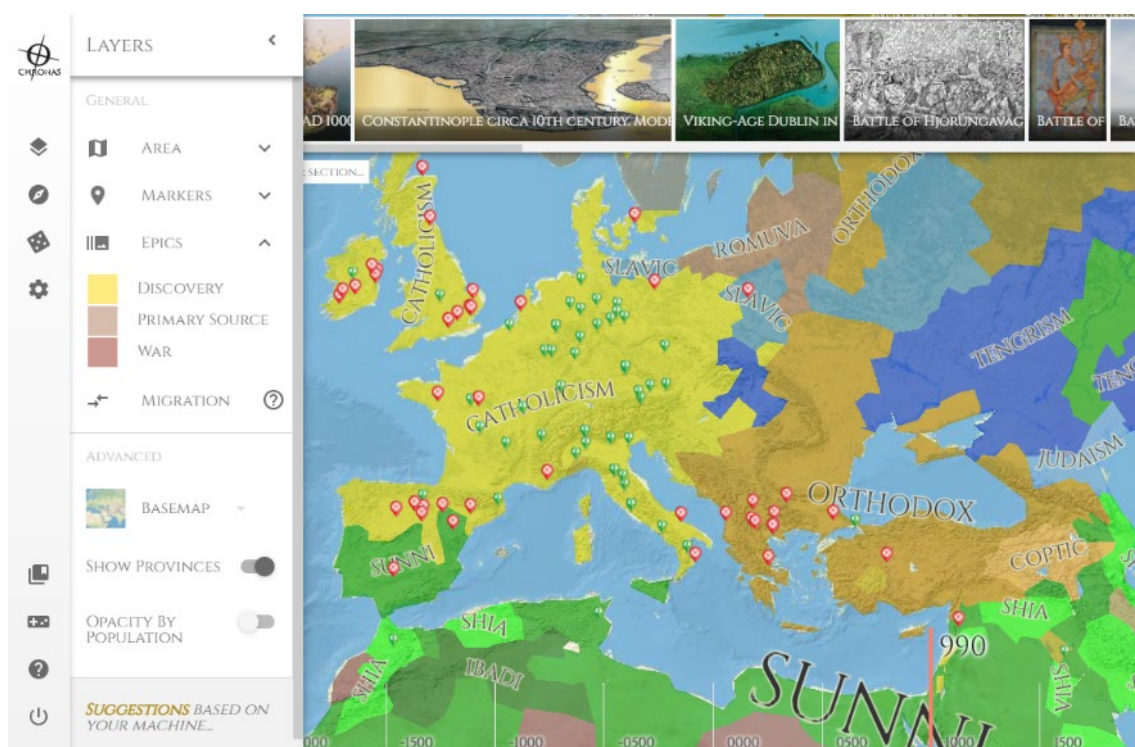


Figura 24 Captura de pantalla de la aplicación Chronas. Se aprecian marcadores de dos tipos, frontera de las creencias religiosas de la época, definidas por la línea de tiempo inferior. Fuente:(Dietmar 2021b)

La aplicación está desarrollada sobre React (Meta Platforms 2022) y el sistema de visualización usa las librerías Leaflet (Agafonkin 2020). No utiliza agrupamientos, ya que sólo muestra los datos en un momento concreto del tiempo y la resolución del mapa no es muy alta, dado que,

debido a los fines del proyecto, no es necesario. El proyecto es de código abierto y tanto el código (Dietmar 2021c), como la API (Aumann 2021) están disponibles públicamente.

#### 4.4.5. Sanborn Maps Navigator

Este proyecto (Quian 2020b) permite ver mapas antiguos de diferentes barrios de poblaciones de EEUU, realizados en los dos últimos siglos por la compañía Sanborn. Aunque la colección oficial es de más de 50.000 mapas, sólo hay alrededor de 32.000 mapas disponibles de forma online, obtenidos a través de la Biblioteca del Congreso de EEUU. Además, tiene una base de datos de periódicos de la época, cuya portada se muestra aleatoriamente.

Se trata de un mapa de EEUU bidimensional, sobre el cual, se dibujan zonas, con diferente intensidad de color, indicando el número de mapas que tiene. Para ello se usa la técnica de cloropetas (Calvo Palacios, Pueyo Campos 1989) .

Los datos se desconoce si están soportados en un grafo de conocimiento, pero hay publicada una API (Trent, Library of Congress 2020) para poder acceder a ellos, también está publicado el código fuente del proyecto (Quian 2020a), en código abierto.

Aunque no posee línea de tiempo, es interesante como hace posible que no sean necesarios mediante la técnica de cloropetas. En la Figura 25, se puede observar una captura de pantalla de la aplicación.

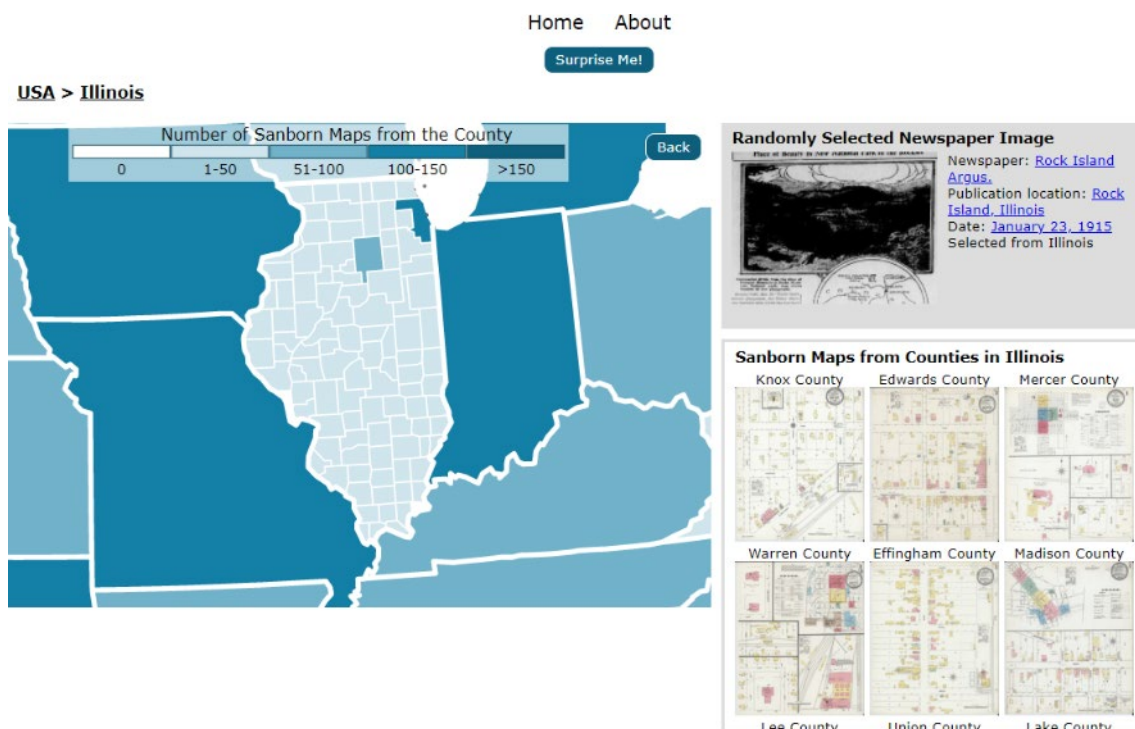


Figura 25 Captura de pantalla de Saborn Maps Navigator. Fuente: (Quian 2020b)

#### 4.4.6. Jewish Cultures Mapped

El Centro de Estudios Culturales de Israel, Daat Hamakom, creó en 2017 una aplicación para ver hechos culturales judíos en la historia (Daat Hamakom 2017).

Aunque no se muestra la cantidad total de “hechos”, se publica en la página del proyecto, que los “hechos” relacionan 200 personas y 65 organizaciones, en 355 lugares. Los “hechos” pueden suceder en un momento puntual del espacio, o en un intervalo de tiempo, como puede ser la estancia de una persona en una ciudad.

La aplicación no usa clusters, porque tiene dos modos de búsqueda y visualización, el primero es a través de una línea de tiempo, combinada con el mapa. La línea de tiempo muestra en el tiempo todos los hechos de la base de datos seleccionados, inicialmente son todos. Seleccionado un hecho en la línea de tiempo, se selecciona en el mapa, y viceversa. Desde el mapa, es posible acceder a más información sobre el objeto. La línea de tiempo, aunque interactivamente puede hacerse más grande, para facilitar su manejo y ver mejor los datos, no determinar un momento en el tiempo, filtrando los datos de ese momento, únicamente permite ver y seleccionar eventos.

La otra forma de buscar y visualizar los datos es activando filtros, que sustituyen la línea de tiempo, por una lista de lugares, personas, organizaciones, y al seleccionar el elemento, se visualizan en el mapa los eventos relacionados con esa persona, organización, etc. Los filtros no son incrementales, por lo que la capacidad de filtro es muy limitada.

La aplicación muestra relaciones con el evento seleccionado, uniéndolos con líneas.

En la se muestra una captura de pantalla de la aplicación. En esta captura puede verse la línea de tiempo y como muestra las relaciones entre eventos.

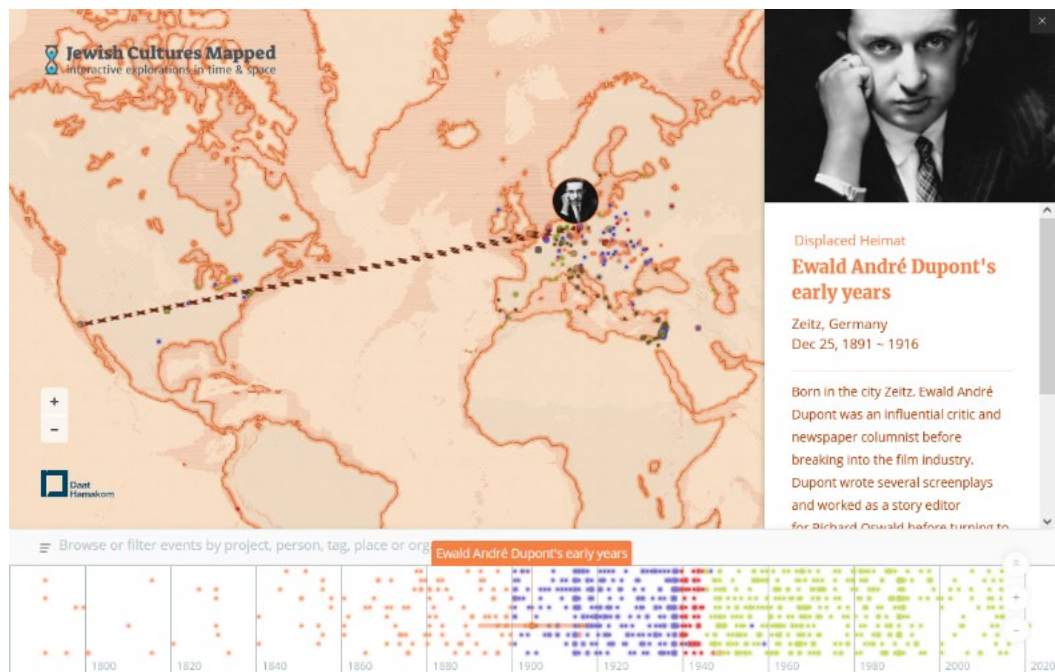


Figura 26 Captura de pantalla de la aplicación Jewish Cultures Mapped. Fuente:(Daat Hamakom 2017)

No aparece publicado si los datos están en un grafo de conocimiento, o una base de datos, así como tampoco se publica ni el código fuente del proyecto ni una API de consulta. Únicamente se publica un proceso para introducir nuevos datos en el sistema.



#### *4.4.7. Paleontology Information Explorer*

La universidad de Idaho creó en 2017 un prototipo para visualizar datos sobre los fósiles de EEUU, soportados por una ontología (Wang, C., Ma, and Chen 2018b). El prototipo (DCS. Univ. Idaho 2019) consiste en un mapa bidimensional, sobre el cual se visualizan con uso de marcadores simples la posición de los fósiles. Aunque no utiliza clusters, dado el limitado conjunto de datos, alrededor de 350 fósiles, este trabajo es muy interesante por el hecho de desarrollar una ontología propia para representar la información, con el objetivo de unificar ontologías temporales geológicas. Según describen en su publicación, existen varias ontologías para representar este tipo de datos temporales, cuya notación cambia en numerosas ocasiones según el país. Los autores, han diseñado un grafo de conocimiento, que unifica estas ontologías, que puede descargarse públicamente (Xiaogang 2021).

Además, también han desarrollado una línea de tiempo jerárquica, e interactiva, que permite navegar de forma sencilla por los períodos geológicos y seleccionar uno para poder visualizar en el mapa estos datos. La herramienta no permite la visualización de relaciones entre los objetos, ni tampoco permite ver los datos de varios periodos de tiempo de forma simultánea.

El código fuente del proyecto puede descargarse públicamente (Xiaoang 2020).

En la Figura 12 se expone una captura de pantalla del piloto desarrollado en la que puede verse claramente la línea de tiempo, y sus diferentes periodos relacionados jerárquicamente.

#### *4.4.8. Basados en Google Maps*

Una solución muy popular para visualizar datos espacio temporales en un mapa bidimensional, y también tridimensional es el uso de las herramientas de Google Maps (Google 2021c).

Estas herramientas, son siempre accesibles públicamente, pero a partir de un número de usos, muy superior al de un usuario medio, pero generalmente no válido para una aplicación con múltiples usuarios tiene un coste económico.

Las herramientas de Google Maps permiten visualizar mapas de todo el mundo en dos y tres dimensiones, usando fuentes de imágenes, en tiempo real. Además, tienen una API que permite dibujar marcadores, utilizando diseños propios, y visualizar clusters.

En la actualidad existen varias herramientas que usan esta solución para representar datos de patrimonio cultural. Si tienen un conjunto muy reducido de datos, simplemente los muestran en la herramienta, pero cuando tienen un número importante, han de introducir sus propios filtros, como es el caso del Atlas Cultural de Australia (Stadler 2021), que permite filtrar entre diferentes tipos de objetos de patrimonio cultural, incluso limitando el tiempo en el que se crearon, o como el explorador del patrimonio cultural de la ONG Cyark (Cyark 2021), que muestra una barra de tiempo, que usa un histograma el conjunto de datos que tiene en cada momento del tiempo, además de permitir seleccionar los datos de un momento del tiempo.

Hay varias aplicaciones que usan estas herramientas, debido a su popularidad, pero no muestran relaciones entre los objetos, ni tampoco los datos de los objetos en varios intervalos de tiempo de forma simultánea. Sin embargo, al ser herramientas de Google, están integradas con el resto de herramientas de esta plataforma, que son muy útiles, como es el caso de Street View, que permite realizar un paseo virtual por el mapa.

En la Figura 27 se muestran dos capturas de pantalla con la interfaz de las dos herramientas comentadas en esta sección.

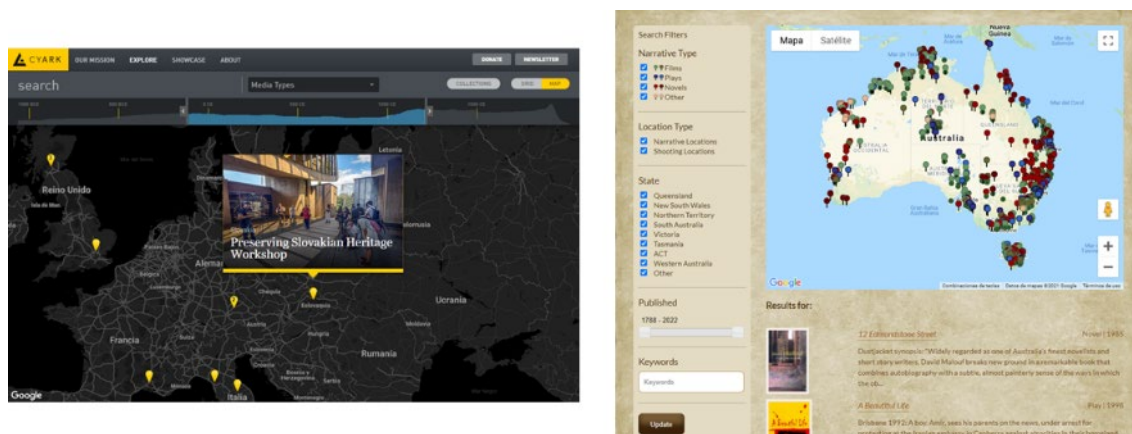


Figura 27 Captura de pantalla del Explorador de Patrimonio de Cyark (izquierda) y Atlas Cultural de Australia (derecha). Fuentes respectivamente: (Cyark 2021, Stadler 2021)

#### 4.4.9. Resultados

En esta sección se han expuesto varias soluciones representativas de plataformas web, de visualización interactiva de datos de patrimonio cultural. Tras analizar los parámetros expuestos al principio de la sección, se procede a recoger los resultados obtenidos en la Tabla 3 Resumen de los datos obtenidos a partir del análisis de las aplicaciones de visualización web de patrimonio cultural.

El significado de los títulos presentes en la columna de la tabla es:

- Nombre: Nombre de la aplicación
- Datos: Número de datos.
- Grafos: Si usa, o no, grafos de conocimiento para soportar la información.
- Línea: Si usa líneas de tiempo.
- Grupos: Si usa técnicas de agrupamiento, o clustering, para mostrar datos.
- Relación: Si permite ver las relaciones entre los diferentes objetos.
- Abierto: Si el código fuente de la aplicación está disponible, así como la API para interactuar con los datos.
- Dimensión: Si usa una representación en 3D, 2D, o 2.5D.
- Tiempo Simultáneo: Permite ver datos existentes en varios momentos en el tiempo de forma simultánea.

En el caso de que se desconozca el dato se indica mediante el símbolo de interrogación.

Tabla 3 Resumen de los datos obtenidos a partir del análisis de las aplicaciones de visualización web de patrimonio cultural.

| Nombre              | Datos | Grafos | Línea | Grupos | Relación | Abierto | Dimensión | Tiempo Simultáneo |
|---------------------|-------|--------|-------|--------|----------|---------|-----------|-------------------|
| Museum of the World | 319   | ?      | SI    | NO     | SI       | NO      | 2.5D      | SI                |

|                   |         |    |    |    |    |    |         |    |
|-------------------|---------|----|----|----|----|----|---------|----|
| Cronobook         | 304.619 | NO | NO | SI | NO | NO | 2D      | NO |
| EAMENA            | 338.000 | SI | SI | SI | NO | SI | 2D      | NO |
| Chronas           | 114.000 | SI | SI | NO | NO | SI | 2D      | NO |
| Sanborn Maps N.   | 32.000  | ?  | NO | SI | NO | SI | 2D      | NO |
| Jewish C.M.       | ?       | ?  | SI | NO | SI | NO | 2D      | NO |
| Paleontology I.E. | 350     | SI | SI | NO | NO | SI | 2D      | NO |
| Google Maps       | ?       | ?  | SI | SI | NO | ?  | 2D / 3D | NO |

Debido a que los datos de las aplicaciones que usan Google Maps pueden ser muy variados. La cantidad de datos, el uso de grafos de conocimiento, así como que se publiquen el código fuente del proyecto y la API, se ha preferido marcar esta información como desconocida. Únicamente se han introducido datos de las características relacionadas con la herramienta Google Maps.

#### 4.5. CONCLUSIONES

En este capítulo se ha realizado un estudio sobre las técnicas de visualización de datos, haciendo especial énfasis en el campo de visualización de datos espacio temporales.

Aunque la visualización de datos es un campo muy activo, hay varias líneas en las que hay que trabajar para obtener mejores resultados:

- Una mejor aplicación de las técnicas de visualización en tres dimensiones. En la actualidad la realidad virtual y la realidad aumentada ha experimentado un crecimiento muy alto, debido a la mejora y el abaratamiento de los dispositivos (Sirohi, Agarwal et al. 2020) Aunque está cambiando la tendencia, en la actualidad, hay muy pocas técnicas de visualización tridimensionales que no se usen sobre todo por la espectacularidad visual.
- La visualización de las relaciones de los objetos entre sí, es un tema en el que aún se necesita realizar mucho trabajo. Generalmente, las soluciones desarrolladas tienen una funcionalidad muy reducida, además de que no hay un número importante de trabajos que incluyan este tema. Este tipo de visualizaciones permite ver cómo se relacionan entre sí objetos, que tengan varias características en común. Lo normal en visualización es seleccionar objetos mediante filtros, de esta forma se encuentran objetos que pertenecen a un grupo que tienen una serie de características similares (tiempo, técnica, lugar, etc.) Sin embargo, en ocasiones, es interesante detectar relaciones entre estos objetos, por ejemplo, que tengan un material poco habitual, o que hayan sido desarrollados por el mismo autor. De esta forma se puede visualizar, junto con el espacio y tiempo, que objetos de ese grupo tienen esa característica. Esta tarea tiene la dificultad adicional de que generalmente visualizar estas relaciones introduce mucho “ruido” gráfico en la escena y añade nuevos elementos a la interfaz de usuario, que hace más complejo su uso.
- El mismo problema aparece a la hora de visualizar datos temporales de forma simultánea. Excepto el uso de cubos espacio temporales, que no es muy frecuente encontrarlo en la

mayoría de aplicaciones desarrolladas, apenas hay más técnicas que permitan realizar esta tarea.

También se ha estudiado la visualización de datos que están en grafos de conocimiento. Generalmente, este dato es secundario, ya que la forma de obtener los datos es realizando consultas al grafo de conocimiento y que estos sean procesados por el sistema de visualización. Esta forma de proceder no toma partido de las posibilidades del grafo de conocimiento, y estos sistemas son usados como si fueran bases de datos. Sin embargo, hay algunos investigadores que piensan que se podrían obtener más ventajas utilizando las posibilidades de un grafo de conocimiento. En su trabajo usan otro grafo que representará las diferentes posibilidades de visualización, cuyos conceptos han de mapearse con los del grafo cuyo contenido se desea visualizar. Defienden el uso de una ontología de visualización, que represente cómo se ha de visualizar el contenido. En esta corriente hay dos visiones. Una que pretende que la definición y visualización se lleve a cabo de forma completa en la ontología. Modelizando todas las posibilidades de visualización y usando la ontología para indicar cómo visualizarla, como es el caso de la ontología VISO y el lenguaje RVL. Otra forma de usar la ontología de visualización es la definida por los autores de la ontología VUMO. En este caso la ontología de visualización representa conceptos muy complejos de visualización, que se mapean con la ontología de datos. En la primera opción, el sistema de visualización sería genérico, y sólo habría que ampliarlo si se amplía el lenguaje RVL, o la ontología de visualización. En la segunda, habría que hacerlo de nuevo, o mejorarlo para visualizar nuevas formas de representación.

Para terminar, en este capítulo se han analizado diferentes plataformas web de representación de datos de patrimonio cultural, ya que es uno de los objetivos de esta tesis. De este estudio se extraen las siguientes conclusiones:

- Aunque no es algo generalizado, hay varias implementaciones que toman sus datos de grafos de conocimiento, en algunos casos basados en CIDOC-CRM, otros de Wikidata, y algunos creando su propio grafo, a partir de un modelo propio.
- Los últimos desarrollos tienden a publicar el código fuente de su trabajo y la API en un repositorio público para que pueda ser accedido públicamente.
- Apenas hay soluciones que usen entornos tridimensionales para mostrar la información, generalmente usan mapas bidimensionales.
- La mayoría de ellos utilizan técnicas de agrupamiento de datos, pero hay muy pocas soluciones que muestren relaciones entre los objetos que están representando, y los que lo hacen usan técnicas muy básicas, como dibujar líneas entre ellos. Esta solución, como hemos visto en este capítulo, no es muy válida para realizar análisis en conjuntos importantes de datos.
- Generalmente, este tipo de aplicaciones permiten filtrar los datos de un periodo de tiempo, bien mediante un formulario, o con el uso de líneas de tiempo, algunas de ellas utilizando técnicas interactivas muy avanzadas. Pero estas soluciones no suelen mostrar datos en diferentes periodos de tiempo de forma simultánea, a pesar de que hay técnicas sencillas que dan opción a ellos. Esto probablemente no se lleve a cabo, porque no está soportado por las herramientas que usan para la representación.
- Muchas de estas plataformas manejan cantidades importantes de datos, cercanas al medio millón de datos.

## CAPÍTULO 5. TECNOLOGÍAS DE INTEGRACIÓN

### 5.1. RESUMEN

En este capítulo se realiza un estudio de herramientas software relacionadas con los objetivos de esta tesis. Se estudian aquellas que permiten crear, mantener y acceder al contenido de un grafo de conocimiento que pueda soportar grandes cantidades de información. Además, se analizarán algunas herramientas para visualizar, y enlazar datos contenidos en un vocabulario.

Para los fines de esta tesis, es igualmente necesario, el análisis de plataformas software de visualización que permitan integrarse sobre aplicaciones web. Estas plataformas deben de permitir una comunicación total con la aplicación web, además de representar visualmente la información con los requerimientos necesarios y permitir al usuario final una interacción adecuada.

Otro tipo de soluciones que se estudiarán en este capítulo son aquellas relacionadas con la gestión y tratamiento, e integración de vocabularios.

### 5.2. HERRAMIENTAS DE GESTIÓN DE GRAFOS DE CONOCIMIENTO

Cada vez hay más aplicaciones en el mercado que usan grafos de conocimiento, hay estadísticas recientes que cuantifican la existencia de 9960 conjuntos de datos, que soportan un total de 150 billones de tripletas (Ali, Saleem et al. 2020). Las aplicaciones que usan estos grafos comprenden desde temas domésticos, turísticos, energéticos y varios temas transversales (Fensel, Şimşek et al. 2020). Pero por muy bien diseñado que este un grafo de conocimiento, al igual que sucede con las bases de datos relacionales, o cualquier sistema de gestión de datos, si el motor, o sistema que se encarga de encargarse de mantenerlo y de atender las consultas no es muy bueno, el rendimiento tampoco lo será.

Debido a esto, en los últimos años han mejorado y proliferado nuevos sistemas de gestión software de grafos de conocimiento (Atemezing, Amardeilh 2018).

Los motores de RDF se pueden dividir en dos clases, los centralizados, que almacenan los datos en un solo nodo, y motores distribuidos, que permiten hacerlo en varios nodos. La complejidad de los grandes conjuntos de datos ha hecho que los centralizados sean muy poco eficientes para satisfacer la cada vez más grande demanda de consultas, en lo referentes a almacenamiento, capacidad de cálculo y procesamiento (Wylot, Hauswirth et al. 2018).

Analizar los diferentes motores RDF existentes en el mercado es una labor que queda fuera de los objetivos de esta tesis. Por ello, se analizarán únicamente los sistemas más populares, que han sido evaluados en artículos científicos en los que se realiza una comparativa de su eficiencia (Ali, Saleem et al. 2020, Salehpour, Davis 2020, Atemezing, Amardeilh 2018, Wylot, Hauswirth et al. 2018) y que han obtenido los mejores rendimientos.

A continuación, se expondrá de forma detallada las características de los principales motores, recabando información sobre ellos, que se recapitula en una sección de resultados.

### *5.2.1. Virtuoso*

Virtuoso Universal Server (OpenLinkSw 2021) es un motor de base de datos híbrido que presenta un sistema que aúna bases de datos virtuales, texto libre, OWL, RDF, XML, bases de datos relacionales, junto con las funcionales de un servidor de archivos y un servidor de aplicaciones.

Este sistema, mediante un único proceso, implementa varios protocolos. Está diseñado para obtener el máximo provecho del soporte de concurrencia y múltiples procesadores del sistema operativo (Erling 2012).

El acceso a las bases de datos relaciones se proporciona mediante ODBC y JDBC. Proporciona un repositorio DAV, y puntos finales de los protocolos SOAP y WS en varios lenguajes de programación.

Almacena los datos RDF en bases de datos relacionales desde el inicio del modelo, sin embargo, por temas de rendimiento ha desarrollado motores RDF personalizados.

Además de implementar SQL, para realizar consultas sobre bases de datos relaciones, también soporta el lenguaje de consultas SPARQL, su extensión GeoSPARQL, y la versión SPARQL compatible con JENA. Además, también soporta razonamiento OWL.

La escalabilidad es una de sus principales características, permitiendo migrar a un sistema de altas prestaciones con varios servidores replicados y balanceo de carga de forma relativamente sencilla.

Existen varios trabajos publicados en lo que se evalúa su rendimiento (Salehpour, Davis 2020, Atemezing, Amardeilh 2018, Jovanovik, Spasić 2018), y aunque hay áreas en las que otras alternativas son mejores, en general, el rendimiento es muy bueno.

Esta tecnología ha sido usada para soportar los grafos de conocimiento de muchos de los principales proyectos de Linking Open Data Project, como son DBPedia, Geonames, o Freebase.

### *5.2.2. RDF4J*

RDF4J (RDF For Java) (Eclipse Foundation 2022) es el nuevo nombre que toma el proyecto OpenRDF Sesame tras ser soportado por la empresa Aduna en 2016.

Es un marco de trabajo de código abierto para el mantenimiento, consulta y análisis de datos en RDF. Puede integrarse en una aplicación standalone, o como servidor, que ofrece un API para la manipulación y consulta de los datos en RDF y OWL.

Ofrece un API de interfaz apilable, que permite añadir diferentes funcionalidades.

Soporta los lenguajes de consulta SPARQL y su extensión GeoSPARQL.

También ofrece razonamiento OWL.

### *5.2.3. StarDog*

Stardog (StarDog Union 2014, W3C 2015) es un motor de base de datos RDF propietario, que soporta razonamiento OWL y es conocido por la gran velocidad de las consultas en SPARQL.

La velocidad de sus consultas y su potente sistema de validación de restricciones (Pérez-Urbina, Sirin et al. 2021) , junto con las diversas posibilidades de escalabilidad que ofrece son sus principales características.

Hay varios trabajos publicados (Escobar, Roldán-García et al. 2020, Tchouangouem, Pauwels et al. 2019) sobre grafos de conocimiento que usan este motor, que generalmente ofrecen comparativas con otros sistemas, en los que muestran que sus resultados en velocidad y validación son mejores que el resto.

También soporta la realización de consultas en SQL.

### 5.2.4. Jena + Fuseki

Jena (Apache Foundation 2021b) y Fuseki (Apache Foundation 2021a) son dos herramientas de código abierto desarrolladas por Apache Software Foundation.

Jena se desarrolló en HP Labs en el 2000, y en el 2009 se cedió el proyecto a Apache Foundation.

La arquitectura de Jena ofrece:

- Motor de Inferencia sobre RDF y OWL.
- Motor de consultas SPARQL.
- API para mantenimiento de ontologías en RDF y OWL.

Jena está desarrollada en Java por lo que funciona en todos los sistemas operativos.

Fuseki es un servidor de SPARQL, de código abierto, desarrollado en Java, que puede ejecutarse como servidor y como aplicación web.

Está completamente integrado con Jena, por lo que los dos juntos constituyen una plataforma que permite el acceso web a grafos de conocimiento.

### 5.2.5. Resultados

Tras analizar los datos de las diferentes plataformas, se observa que, con la excepción de poder realizar consultas en SQL, y que sean sistemas propietarios, o de código abierto, tienen las mismas características principales:

- Permiten el acceso y manipulación de datos en OWL y RDF.
- Soportar consultas en SPARQL, ofrecen razonamiento y soportan SHACL.
- Todos tienen más de 15 años, están activos y han publicado una nueva versión hace menos de un año. Cuentan con una comunidad de desarrolladores importantes, bien por ser de código abierto, y/o porque lo usan en el desarrollo de proyectos.

Sin embargo, en el rendimiento las diferencias son importantes. Analizando los resultados de una comparativa reciente (Nguyen ,H., Mau Quoc, Martin Serrano, Han Nguyen Mau, John G. Breslin, Danh Le Phuoc 2019), donde se analizan las herramientas expuestas en esta investigación se observa que hay una notable diferencia entre las plataformas propietarias y las de código abierto, basadas en Java. En la Figura 28 se muestra una tabla, con datos de este estudio, donde tras realizar 11 consultas diferentes, se analiza para cada plataforma el coste de procesamiento de la consulta (parsing), el de optimización y el de ejecución.

En los resultados se aprecia claramente que Virtuoso es la plataforma que en general mejor rendimiento consigue, seguido de Stardog, RDF4j y Jena en último lugar. Las principales diferencias respecto a Virtuoso se hallan en las consultas que contienen subconsultas.



| Query | Jena          |              |              | Virtuoso      |              |              | RDF4j         |              |              | Stardog       |              |              |
|-------|---------------|--------------|--------------|---------------|--------------|--------------|---------------|--------------|--------------|---------------|--------------|--------------|
|       | Pars.<br>(ms) | Opt.<br>(ms) | Exe.<br>(ms) | Pars.<br>(ms) | Opt.<br>(ms) | Exe.<br>(ms) | Pars.<br>(ms) | Opt.<br>(ms) | Exe.<br>(ms) | Pars.<br>(ms) | Opt.<br>(ms) | Exe.<br>(ms) |
| Q1    | 10            | 1814         | 48304        | 1             | 75           | 4            | 9             | 542          | 1210         | 2             | 101          | 9            |
| Q2    | 7             | 105          | 53           | 1             | 70           | 12           | 6             | 1100         | 127          | 1             | 94           | 42           |
| Q3    | 15            | 1320         | 48833        | 1             | 500          | 74           | 16            | 2500         | 226414       | 1             | 3085         | 11410        |
| Q4    | 13            | 1840         | 48305        | 2             | 560          | 27           | 15            | 1432         | 309528       | 1             | 3115         | 10883        |
| Q5    | 14            | 1410         | 48739        | 1             | 50           | 4            | 15            | 90           | 14           | 2             | 73           | 66           |
| Q6    | 13            | 163          | 10           | 1             | 48           | 0            | 16            | 813          | 356          | 1             | 80           | 64           |
| Q7    | 20            | 1731         | 48415        | 2             | 72           | 4            | 21            | 1430         | 227404       | 1             | 204          | 3            |
| Q8    | 10            | 159          | 373          | 1             | 65           | 4            | 8             | 1100         | 868          | 1             | 71           | 41           |
| Q9    | 16            | 1774         | 48388        | 1             | 110          | 8            | 12            | 2113         | 151653       | 1             | 105          | 136          |
| Q10   | 22            | 1677         | 48470        | 2             | 635          | 8            | 21            | 1546         | 144141       | 2             | 3093         | 11201        |
| Q11   | 5             | 37           | 410          | 1             | 15           | 0            | 4             | 20           | 4            | 1             | 46           | 67           |

Figura 28 Comparativa de Rendimiento en herramientas de gestión de grafos de conocimiento. Fuente: (Nguyen ,H., Mau Quoc, Martin Serrano, Han Nguyen Mau, John G. Breslin, Danh Le Phuoc 2019)

### 5.3. HERRAMIENTAS DE VISUALIZACIÓN DE DATOS ESPACIO TEMPORALES SOBRE PLATAFORMAS WEB

Después de analizar en el capítulo anterior las diferentes herramientas de visualización web de datos de patrimonio cultural, se observa que la visualización de mapas sobre plataforma web es un punto generalizado. Esto es importante, sobre todo si se desea visualizar espacialmente, con un mínimo de precisión, la ubicación de estos datos. Obviamente, no es el único elemento a tener en cuenta. Independientemente que el sistema visualice mapas, o no, se debe de permitir que exista una interactividad, navegar por el espacio visualizado, además de representar elementos, a ser posible en dos y tres dimensiones (iconos, relaciones, fotografías, etc.)

La representación de mapas en plataformas web, ha evolucionado mucho desde los mapas estáticos, pero con alto contenido, de Google Maps en 2005, hasta la actualidad (Köbben, Kraak 2020). En la actualidad hay infinidad de aplicaciones web que usan mapas en todos los campos, desde turismo para reservar hoteles (Booking 2021), o restaurantes (Tripadvisor 2021), ver datos de patrimonio cultural (Stadler 2021), hasta para ver la evolución en tiempo real de la COVID 2019 (John Hopkins. U. Medicine. 2021). La representación de mapas en la web suele estar basada en el uso de fragmentos de diferente resolución (tiles), que se van obteniendo bajo demanda, de diferentes proveedores de mapas. Esta evolución ha provocado que surjan nuevos estándares para transferir datos geográficos, como son GeoJSON (Butler, H., Daly, M., Doyle, A., Gillies, S., Hagen, S., and T. Schaub 2016), o TopoJSON (Bonativa, L., Bostock, M. 2018).

En esta rápida evolución, los estándares de W3C, como HTML 5, JavaScript y WebGL han tenido un papel importante (Boulos, Warren et al. 2010, Köbben, Kraak 2020). JavaScript, desde el navegador web, en la parte, cliente, aporta la interactividad y capacidad de procesamiento (Evans, Romeo et al. 2014). Cuando se trata de gráficos de alto nivel, WebGL (Web Graphic Language) (The Khronos Group Inc. 2021) es un estándar para representar gráficos que aprovecha la Unidad de Procesamiento Gráfico (GPU), presente en los ordenadores modernos y también en muchos teléfonos móviles (Evans, Romeo et al. 2014). La presencia de estos sistemas ha reducido las necesidades de recursos en la parte servidor y permite que un terminal cliente pueda ejecutar funcionalidades avanzadas de geoprocesamiento y visualización que hasta hace pocos años sólo estaban asequibles en ordenadores de alta gama (Kulawiak, Dawidowicz et al. 2019). De hecho, en la actualidad, generalmente, más que el rendimiento, o la velocidad, en la mayoría de los casos preocupan otros factores, como el requerimiento, o no, de instalación de



plugins, la dificultad de programación, o que su uso implique adquisición de una licencia con coste (Zunino, Velázquez et al. 2020).

Debido a la importancia de la representación de mapas en la web, donde cada vez se requiere más velocidad, más interacción y más elementos de representación, han surgido varias librerías basadas en Javascript para su representación (Zunino, Velázquez et al. 2020). Las más utilizadas son: Leaflet (Agafonkin 2020), MapBox GL JS (Mapbox 2021) y OpenLayers (OpenLayers 2021).

Además de estas librerías diseñadas específicamente para la representación e interacción con mapas, existen otras librerías, que permiten visualizar gráficos e interactuar con ellos desde un navegador web. Algunas son muy antiguas y han caído en desuso, como pueden ser Java3D, VRML (Virtual Reality Modeling Language), o X3D, generalmente por la necesidad de plugins, dificultad de programación, o sus limitaciones. Otras librerías en cambio, son muy utilizadas como WebGL, D3.js, three.js, o Unity3D.

Seguidamente, se detallan cada una de las librerías más utilizadas, con el objetivo de obtener datos sobre determinadas características, que se resumirán en una última sección de resultados y permitirá realizar una comparativa entre ellas.

### 5.3.1. WebGL

WebGL (The Khronos Group Inc. 2021) es un estándar web abierto, multiplataforma y libre de derechos que ofrece un API basado en OpenGL ES (Khronos Group 2022), a través del elemento Canvas (The Khronos Group Inc. 2021) de HTML 5. La API de WebGL está basada en componentes usados para transformar vértices, o coloreado de píxeles (shaders), con construcciones muy similares semánticamente a la API de OpenGL (Open Graphic Language) ES.

La última versión de WebGL, expone la API de OpenGL ES 3.0.

Este API está soportada por los principales navegadores, en sus versiones para los diferentes sistemas operativos. Safari, Chrome, Edge y Firefox son miembros del grupo de trabajo de WebGL.

Lo más interesante de esta API, además de su potencia y posibilidades gráficas, es que funciona en los principales navegadores sin necesidad de plugins externos, ya que está implementado en el propio navegador. Por lo que es una alternativa ideal para representar una escena en 3D en una aplicación web. Además de funcionar correctamente, tiene un buen rendimiento. En la Figura 29 se muestra el soporte de esta plataforma en los principales navegadores y sistemas operativos.

| IE   | Edge  | Firefox | Chrome | Safari  | Opera   | Safari on iOS | Legend |  |  |  |
|------|-------|---------|--------|---------|---------|---------------|--------|--|--|--|
|      |       | 2-3.6   | 4-7    | 3.1-5   | 10-11.5 |               |        |  |  |  |
|      | 12-18 | 4-23    | 8-32   | 5.1-7.1 | 12.1-18 | 3.2-7.1       |        |  |  |  |
| 6-10 | 79-97 | 24-95   | 33-97  | 8-15.1  | 19-82   | 8-15.1        |        |  |  |  |
| 11   | 98    | 96      | 98     | 15.3    | 83      | 15.3          |        |  |  |  |
|      |       | 97-98   | 99-101 | 15.4-TP |         | 15.4          |        |  |  |  |

| Opera Mini | Android Browser | Opera Mobile | Chrome for Android | Firefox for Android | UC Browser for Android | Samsung Internet | QQ Browser | Baidu Browser | KaiOS Browser |
|------------|-----------------|--------------|--------------------|---------------------|------------------------|------------------|------------|---------------|---------------|
|            | 2.1-4.4.4       | 12-12.1      |                    |                     |                        | 4-15.0           |            |               |               |
| all        | 97              | 64           | 97                 | 96                  | 12.12                  | 16.0             | 10.4       | 7.12          | 2.5           |

Figura 29 Soporte de WebGL en diferentes navegadores y sistemas operativos. Fuente: (Deveria, Lennart 2022)

Lo más negativo de esta API es que es de bajo nivel, por lo que hacer aplicaciones complejas requiere contar con un conjunto de librerías que permitan desarrollar con rapidez. Además, al estar ejecutándose en una aplicación web, la gestión de memoria la realiza JavaScript, cuyo garbage collector no se ejecuta hasta que finaliza el proceso de dibujo del fotograma (frame). Esto implica planificar con cuidado el uso de recursos que requieran memoria, ya que su ocupación no podrá ser liberada hasta finalizar el dibujo de la representación gráfica.

Como se describe a continuación, muchos sistemas establecen una capa de alto nivel sobre WebGL, permitiendo realizar desarrollos más rápidos, pero basándose en WebGL.

### 5.3.2. D3.JS

D3 (Bostock, M., Ogievetsky, V., Heer, J. 2011) es una librería Javascript de código abierto, diseñada para la visualización de datos mediante el estándar web SVG, utilizando el elemento Canvas de HTML5.

Esta librería ofrece un enfoque de manipulación del DOM basado en los datos, lo cual conlleva una libertad de diseño de interfaces y aprovecha la potencia de los navegadores modernos.

D3 funciona en la mayoría de navegadores y es una alternativa muy potente para la representación de grafos y datos interactivos. El problema de usar SVG (Scalable Vector Graphics) es que no ofrece un API para la representación de gráficos tridimensionales. Aunque hay extensiones de código abierto (Nieke 2021) para la inclusión de este tipo de gráficos, están diseñadas para la representación de objetos sencillos.

El rendimiento en tiempo real es más bajo que WebGL. En una prueba comparativa realizada recientemente (Lindberg 2020), se realizó en D3.JS y en WebGL el renderizado de conjuntos de datos de 1000 nodos en movimiento, que incrementaba en número de forma constante. Los resultados mostraron que D3.JS con SVG estaba muy por detrás de WebGL.

### 5.3.3. *Three.js*

Three.js (Williams 2012) es una librería JavaScript de código abierto que permite la creación de gráficos 3D y 2D en navegadores web. Permite la representación de escenas tridimensionales mediante WebGL y bidimensionales mediante SVG.

La librería fue publicada en 2010, inicialmente desarrollada en ActionScript, publicada en el repositorio público Github (Mrdoob 2022), y posteriormente traducida a JavaScript.

Three.js parte de la creación de una escena, sobre la que se pueden agregar objetos, con materiales, así como luces, cámaras, y la aplicación de shaders accediendo a las capacidades de OpenGL Shading Language (GLSL). Además, lo realiza explotando la unidad de procesamiento gráfico (GPU) del sistema hardware en el que se ejecuta.

Al estar basado en WebGL, utiliza el elemento Canvas de HTML5 para representar gráficamente.

Resumiendo, es una librería de código abierto, que permite representar escenas con la calidad y rendimiento que pueda obtenerse con WebGL, de forma mucho más sencilla que en WebGL y programando en JavaScript. Ofrece la opción de añadir plugins, que facilitan el desarrollo, descargándolos directamente de la página principal, o de diversos repositorios públicos, como Github. Estos plugins suelen ser de código abierto, aunque también existen de distribución comercial. No existe ninguna página comercial genérica para su adquisición.

### 5.3.4. *Unity3d*

Unity [(Unity Tech. 2022b)] es una herramienta orientada al desarrollo de videojuegos en 2D y 3D, propietaria de la empresa Unity Technologies. La primera versión de la herramienta fue liberada en 2005.

Es una herramienta diseñada para el desarrollo en C# o Javascript, aunque principalmente se suele utilizar con C#.

Es una herramienta que permite generar el código desarrollado en varias plataformas (Windows, Mac, Linux, Android, iOS, Wii). Utiliza los motores gráficos OpenGL y Direct3D para los sistemas basados en PC, OpenGL ES para dispositivos móviles y web.

La herramienta se diseñó con el objetivo de “democratizar” el desarrollo de videojuegos, haciendo que fuera accesible al mayor número de personas posibles. Para ello posee dos tipos de licencias, Unity Personal, que es completamente funcional, gratuita y únicamente requiere la publicación del logotipo durante el arranque y que se tenga ingresos superiores a los 100 mil dólares, y Unity Pro, que requiere un pago. El objetivo lo ha cumplido y en la actualidad es una de las plataformas de desarrollo de videojuegos con más usuarios, en 2020 contaba con más de un millón y medio de desarrolladores, y dos billones de usuarios finales (Hollister 2020).

Inicialmente, la herramienta tenía su propio plugin para el desarrollo de páginas web, pero desde su versión 5, en 2014, utiliza WebGL para la representación de gráficos en un navegador web. En el caso de aplicaciones web, la herramienta ofrece una integración dinámica completa con los módulos de la página web, mediante JavaScript.

AssetStore (Unity Tech. 2022a) es una página web que funciona como un market de Assets, o complementos software que ofrecen multitud de servicios. En la actualidad hay más de 11.000 assets, marcados en la categoría de cinco estrellas en el portal. Al igual que los markets de las plataformas de dispositivos móviles, algunos son gratuitos y otros ofrecen diferentes fórmulas de pago.

Permite la integración con todo tipo de plataformas, como SMARTTVs y consolas de videojuegos, además de usar todo tipo de dispositivos de realidad virtual y aumentada, como gafas, guantes, etc.

### 5.3.5. Leaflet

Leaflet (Agafonkin 2022) es una librería JavaScript de código abierto diseñada para la visualización web de mapas interactivos. Su primera versión se publicó en 2011.

Funciona en los principales navegadores y en los diferentes sistemas operativos de escritorio Linux, Windows y Mac OS, así como en los usados en dispositivos móviles, iOS y Android.

A través de datos proporcionados en formato GeoJSON (Geographical Javascript Object Notation) y la asignación de un proveedor de texturas de los mapas, es capaz de visualizar la información suministrada en un mapa interactivo.

Soporta la representación de datos mediante clusters y mapas de calor con la instalación de plugins adicionales.

La librería ofrece en su página web más de 50 plugins para realizar todo tipo de funcionalidades de interacción y representación, además de que hay varios publicados en repositorios públicos como Github.

Es una librería ligera, menos de 50 Kb, cuyos resultados son más que aceptables. La principal limitación es que está diseñada para representar datos en 2D. Aunque algunos de los plugins permiten representar elevaciones y otros gráficos básicos en 3D mediante la librería D3.JS.

### 5.3.6. OpenLayers

OpenLayers (OSGF 2021) es la librería Javascript de código abierto para la visualización de mapas interactivos que lleva más tiempo publicada. La primera versión se publicó en 2005, y fue realizada por Metacarta. Desde 2007 está incluida en los proyectos de Open Source Geospatial Foundation.

Al igual que Three.JS utiliza WebGL sobre el elemento Canvas de HTML5 para renderizar los gráficos.

Soporta la representación de datos mediante clusters y mapas de calor sin plugins.

Al estar basada en WebGL y JavaScript funciona en la mayoría de navegadores y sistemas operativos. Soporta todo tipo de fuentes de datos geográficas, sin plugins. El principal problema que tiene esta librería es el tamaño, necesita más de 1Mb para poder ejecutarse.

### 5.3.7. Cesium.JS

Cesium.JS (Cesium GS. Inc. 2022) es una librería JavaScript de código abierto bajo licencia Apache 2.0, diseñada para crear mapas 3D interactivos mundiales. En la actualidad tiene más de un millón de descargas y se ha convertido en un referente para crear aplicaciones con mapas 3D en la web.

Permite el uso de Teselas 3D (OGC 2018), para diseñar e interactuar con edificios 3D, fotogrametría y nubes de puntos, así como incorporar modelos 3D, gestionar capas de terreno, y además ofrece una visualización en 4D basándose en el tiempo.

La renderización de los gráficos 3D se realiza a través de WebGL, esto asegura su funcionamiento en la mayoría de navegadores y sistemas operativos.

Con esta librería se han realizado proyectos que manejan cantidades ingentes de datos, como un proyecto de visualización 3D basado en nubes de puntos (Green, C. G., and Villa, I. 2019) , realizado por UBER Engineering, en el que demuestra que es capaz de representar nubes de puntos formadas por más de 355 millones de datos.

El problema de Cesium es que todos los elementos 3D están ligados al hecho de representar un mapa, además de tener limitaciones en torno al API que ofrecen para los mapas. Por lo que no se puede hacer una representación de escenas 3D si no se representa un mapa en la escena.

#### 5.3.8. Resultados.

En la Tabla 4 se muestran las diferentes características de las herramientas de visualización de datos espacio temporales en web.

*Tabla 4 Características de librerías de visualización de datos espacio temporales en web.*

| Librería   | Escenas 3D/2D | Solo Mapas | Código abierto | Multiplataforma | Plugins | Bajo nivel |
|------------|---------------|------------|----------------|-----------------|---------|------------|
| WebGL      | Si            | NO         | SI             | SI              | NO      | SI         |
| D3.JS      | 2D*           | NO         | SI             | SI              | SI      | NO         |
| Three.JS   | SI            | NO         | SI             | SI              | SI      | NO         |
| Unity3D    | SI            | NO         | NO**           | SI              | SI      | NO         |
| Leaflet    | 2D            | SI         | SI             | SI              | SI      | NO         |
| OpenLayers | 2D            | SI         | SI             | SI              | SI*     | NO         |
| Cesium.JS  | 2D*           | SI         | SI             | SI              | SI      | NO         |

\* Estas librerías tienen plugins para mostrar modelos y escenas 3D, pero están muy limitadas.

\*\* No es de código abierto, pero ofrece una licencia gratuita a cualquier entidad con menos de 100.0000 \$ de beneficio.

Después de analizar las herramientas más utilizadas para representar datos espacio temporales en la web, a través de representaciones en escenas de dos y tres dimensiones, se ve que en entorno 3D y 2D las principales alternativas son las basadas en WebGL, en concreto Tree.JS y Unity3D. Si se requiere visualizar la información únicamente en entornos de dos dimensiones, las de tres dimensiones serían perfectamente válidas, aunque hay varias alternativas muy potentes, especialmente aquellas diseñadas para visualizar datos sobre mapas.

#### 5.4. CONCLUSIONES

En este capítulo se han expuesto dos grupos principales de tecnologías.

Por una parte, aquellas que dan soporte para la creación, mantenimiento y acceso a los datos de un grafo de conocimiento y por otra aquellas herramientas que permiten visualizar datos interactivos en una aplicación web.

Tras realizar el análisis se puede concluir que las capacidades tecnológicas actuales permiten satisfacer plenamente los objetivos buscados en esta investigación.

Hay una gran variedad de herramientas de soporte de grafos de conocimiento capaces de soportar una gran cantidad de datos, más que suficiente para las necesidades del patrimonio cultural. Además, de estas herramientas, muchas de ellas son de código abierto, o tienen licencias que permiten su uso y escalabilidad. De hecho, de las evaluadas, y para los fines de este trabajo, la plataforma Virtuoso parece la más conveniente, dado que:

- Es capaz de funcionar en varios sistemas operativos.
- Es escalable.
- Hay una versión gratuita con limitaciones de soporte y de escalabilidad que siempre puede ampliarse cambiando la licencia.
- Tiene un rendimiento similar a las mejores herramientas.
- Tiene varias formas de mantenimiento de la información y permite el acceso mediante servicios web, y mediante búsquedas en SPARQL.

De todas formas, el uso de una plataforma u otra, sería independiente para poder realizar la investigación. Ya que la mayoría permiten el soporte de grandes cantidades de información y el acceso a su información a través de la web con SPARQL.

En el campo de la representación gráfica el tema es un poco más delicado. Por una parte, se han expuesto herramientas que permiten mostrar mapas interactivos, ya que al tratarse de datos espacio temporales, en muchas ocasiones habrá un mapa como base. Sin embargo, no hay que representar únicamente escenas con mapas, por lo que estas herramientas podrán ser consideradas si algún día separamos la visualización de mapas, del resto de visualizaciones, a nivel de herramienta, si no se optará por herramientas más genéricas.

Dentro del apartado de herramientas genéricas, hay también un número importante de posibilidades, pero las mejores desde el punto de vista de este trabajo, son Three.js y Unity3D. Ya que permiten representar datos en 3D y 2D en un entorno multiplataforma, con una gran calidad y rendimiento, integrándose en una aplicación web. La principal diferencia entre una y otra herramienta, es que la primera es de código abierto, y la segunda, aunque no lo es, posee un tipo de licencia que hace que sea de acceso gratuito para muchos desarrolladores que quieran usarla. Otra diferencia de estas herramientas, es que Three.js tiene varios plugins y extensiones, pero no se puede comparar este número, ni generalmente la respuesta de sus desarrolladores, con los de Unity3D. Unity3D posee extensiones para trabajar con varios tipos de dispositivos y permite trabajar con plataformas como SMART TV, consolas, etc., que no están disponibles en Three.js.

### SECCIÓN III. PROPUESTAS Y CONTRIBUCIONES

## CAPÍTULO 6. MODELO DE REPRESENTACIÓN DE ESCENAS DE VISUALIZACIÓN DE DATOS

### 6.1. RESUMEN

El contenido del capítulo 6 de esta investigación está enfocado a definir un modelo de visualización de datos espacio temporales de patrimonio cultural. Por ello, con la información obtenida del análisis de diferentes sistemas de visualización en el capítulo 4, se procede a enumerar y detallar los diferentes objetos visuales y procesos que conforman una escena de visualización de este tipo de datos. Los objetos visuales hacen referencia a aquellos objetos, que generalmente poseen una representación visual en una escena, por ejemplo, los marcadores para representar los datos, o grupo de datos. Los procesos comprenden acciones que suceden en la escena, generalmente están relacionadas con interacciones que el usuario puede realizar, pero también pueden estarlo a eventos que suceden en el sistema, como el inicio, la actualización de datos, cambio de escenas, etc.

También se exponen con más profundidad, los problemas y puntos a mejorar de los sistemas analizados en el capítulo 4, junto con una, o varias propuestas de solución, o mejora, de los mismos. Algunos de estos problemas son típicos del dominio de datos del patrimonio cultural, como los relacionados con la incerteza, aunque no son exclusivos del dominio, otros son más comunes y también están presentes en la mayoría de dominios, como la visualización de relaciones.

A modo de conclusión del capítulo, se describen todas las características que definen un modelo de representación para este tipo de sistemas de visualización, que contemple los objetos visuales definidos en el principio del capítulo, así como las soluciones, o mejoras, a los problemas expuestos previamente. Con esta información, el siguiente paso sería diseñar e implementar una ontología que representara este modelo, con el objetivo de poder definir un sistema de visualización.

### 6.2. ELEMENTOS DE UN SISTEMA DE VISUALIZACIÓN DE DATOS DE PATRIMONIO CULTURAL

En los sistemas de visualización de datos espacio temporales analizados en el capítulo 4, tanto en aquellos que estaban relacionados con el dominio de patrimonio cultural, como los que no, existen una serie de objetos visuales y procesos en común, y otros que sólo aparecían en los que visualizaban datos de patrimonio cultural.

A continuación, basándose en dicho análisis, se exponen una serie de conceptos que van a permitir representar un sistema de visualización, tanto los datos y la forma de visualizarlos, como las posibles interacciones a realizar en el sistema, de acorde a la siguiente clasificación:

- **Objetos visuales:** Objetos gráficos que pueden reconocerse visualmente, generalmente estos elementos están relacionados con un escenario, o escena visual sobre la que se representan.
- **Procesos de interacción:** Acciones que se permiten realizar en la escena, ya sea para facilitar la interacción con el usuario, como con otras aplicaciones, o eventos del sistema de representación. Generalmente estos procesos implican consecuencias en la representación de la escena.



### 6.2.1. Objetos visuales

Los objetos visuales con los que se va a representar el contenido del grafo de conocimiento se basan principalmente en dos conceptos: Escena y Elemento Visual.

Por Elemento Visual, se hace referencia a los datos a representar, generalmente una instancia, o grupo de instancias de un concepto del grafo de conocimiento con la información.

Con la Escena se referencia el contenido visual global que se va a representar. Este concepto contiene unas propiedades que la definen, pero además contiene relaciones con todos los elementos visuales que se van a representar en ella.

Seguidamente, antes de enumerar los diferentes objetos visuales y entrar en más detalle, se describirán una serie de consideraciones sobre las dimensiones de un objeto visual y su ubicación en la escena, también sobre las diferentes posibilidades de representación de dicho objeto y sus relaciones con otros objetos, todo ellos teniendo en cuenta las dimensiones espaciales de la escena.

#### Localización de un objeto en la escena

Tal y como hemos visto en los sistemas analizados, como no podría ser de otra forma, la escena podrá mostrarse gráficamente en un espacio bidimensional, o tridimensional. Se podrán referenciar y agrupar objetos, manejando varias dimensiones, incluso cambiar la visualización de acorde a otra dimensión, como pueda ser el tiempo. Sin embargo, en la representación gráfica, todo objeto, o grupo de objetos, requerirá dos, o tres coordenadas, para representarlo en la escena a visualizar, independientemente del dispositivo que se utilice.

Lógicamente, si se va a representar contenido con datos espacio temporales, el espacio y el tiempo serán siempre propiedades clave en los objetos a representar. Sin embargo, pueden existir otras propiedades, cuyo valor en una instancia determinada, puede decidir dónde y cómo va a representarse dicho contenido. Estos valores estarán asociados a un dominio, cuya dimensión debe de estar claramente determinada. Por ejemplo, en el caso del espacio pueden ser valores de longitud, y latitud, que suelen usarse para representar el dato en un mapa, pero también puede ser únicamente un continente, como el caso del British Museum (British Museum, Google Cultural Institute 2015), donde además, el tiempo es un parámetro necesario para asociar la posición del objeto en el espacio de la escena, como también pasa con el sistema de LayerCake (Aga Khan Documentation Center at, MIT 2019)

Por lo tanto, los objetos a representar en la escena pueden tener una serie de propiedades que determinan su posición en la escena. Pero la escena, va a representarse mediante un espacio bidimensional, o tridimensional, donde la zona a visualizar va a ser limitada. Por lo que habrá que definir las dimensiones de la escena, las dimensiones de los valores de los objetos, y también una forma de mapear los valores de dichas propiedades de los objetos a las dimensiones espaciales de la escena.

En la Figura 30 se muestra, con la ayuda de varios esquemas de posibles escenas, como un objeto y sus propiedades se visualiza en diferentes tipos de escena. Las propiedades usadas son datación, y lugar de producción, definido por su longitud y latitud. A la izquierda se muestra una escena en 2.5 dimensiones, similar al caso de British Museum, donde cada dato se visualiza con ayuda de dos dimensiones, la coordenada X sería el resultado de una función que a través de la longitud y latitud del objeto devolvería una variable numérica, asociada a un continente. La parte inferior es un mapa bidimensional tradicional, donde la coordenada x, sería la latitud, y la coordenada “y” la longitud. Por último, a la derecha se trata de un mapa en tres dimensiones,

similar al caso de LayerCake, donde las coordenadas  $x$ ,  $y$  y  $z$ , se tratarán igual que el bidimensional tradicional, pero la coordenada  $z$ , de altura, vendría determinada por la datación del objeto.

El mapeo entre las dimensiones del objeto y las coordenadas en las dimensiones de la escena, puede ser muy diverso. Por ejemplo, en el capítulo 10, se describe el proyecto de Arxiu del Disseny de Valencia, dónde incluso propiedades definidas por el usuario, como “confort” o “elegancia”, que reciben valores dinámicamente definen la posición en escenas de dos y tres dimensiones.

En la Figura 30, también se realiza el hecho de que la escena tiene una parte que se está visualizando en un momento determinado, que es finita, marcada más oscura en la figura, y otra parte que es el resto de la escena, que puede ser finita, o no.

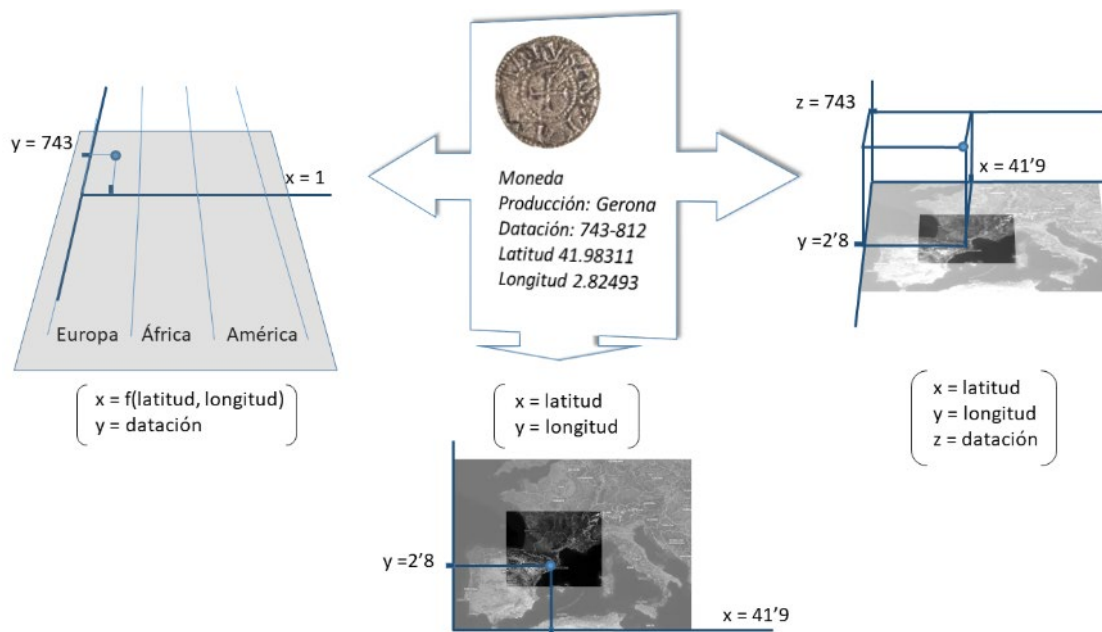


Figura 30 Diferentes formas de representar un objeto en escenas de varias dimensiones y usando diferentes propiedades del objeto para mapear su posición en las dimensiones de la escena, que además tiene una parte visualizada (en negro) diferenciada de su totalidad. Fuente: Elaboración propia.

### Representación de un objeto.

La visualización de un objeto es una de las características que más puede atraer la atención y en ocasiones es una propiedad diferenciadora entre diferentes sistemas de visualización. Sin embargo, es una tarea relativamente sencilla y poco compleja, generalmente, a nivel técnico.

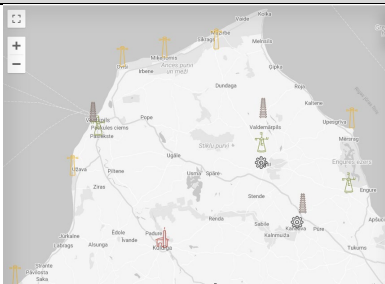
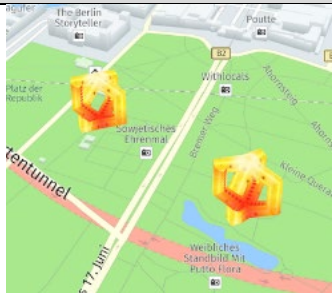
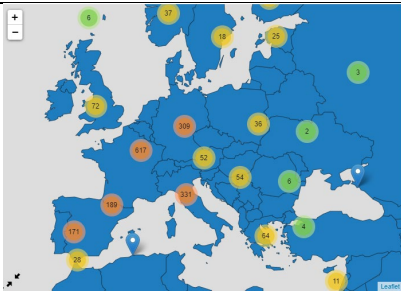

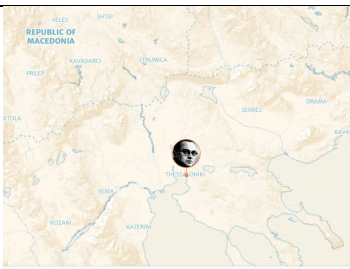

El objeto ha de corresponderse con un concepto de la ontología en la que está basado el grafo de conocimiento a representar.

En el caso de escenas bidimensionales la representación de un objeto se suele realizar mediante una imagen, o mediante formas geométricas con colores (círculos, cuadrados, etc.), que es una solución muy habitual en las aplicaciones basadas en Google Maps (Google 2021c). Estas formas geométricas pueden ir rellenas además de por colores, por imágenes relacionadas generalmente por el tipo de objeto, o con el objeto en concreto (Daat Hamakom 2017).

En las escenas tridimensionales, la representación es totalmente análoga a las dos dimensiones, pero transformando el objeto en su correspondiente representación tridimensional. Por ejemplo, si en dos dimensiones se representa un objeto mediante una imagen en el caso más

genérico, en el entorno tridimensional sería un modelo 3D. Igual que se usa un círculo en entornos bidimensionales, su análogo sería una esfera en el entorno tridimensional. En la Tabla 5 se enumeran los diferentes tipos de representaciones básicas, en escenas bidimensionales, junto a un ejemplo, y su correspondiente representación en escenas de tres dimensiones.

*Tabla 5 Diferentes tipos de representaciones de objetos en la escena. Para cada tipo, además de visualizar un ejemplo se expone su transformación de escena 2D a escena 3D.*

| 2 dimensiones                                 |  | 3 dimensiones   |  |
|---|--|---|--|
| Tipo  | Ejemplo  | Tipo  | Ejemplo  |
| Imagen  |  <p>Fuente: (Interreg., Stonia-Latvia 2018, Android 2022)</p> | Modelo 3D   |  <p>Fuente: (Android 2022)</p>                                   |
| Figura geométrica básica rellena con colores  |  <p>Fuente: (Council of Europe 2021)</p>                     | Poliedro básico relleno con colores.                                      |  <p>Fuente: (Akihiko Kusanagi 2022, Council of Europe 2021)</p> |
| Figura geométrica básica rellena con imágenes |  <p>Fuente: (Daat Hamakom 2017)</p>                         | Poliedro básico donde todas, o una de las caras tiene aplicada un imagen. |  <p>Fuente: Elaboración Propia</p>                             |

A la hora de representar el objeto es importante que su representación gráfica tenga un aspecto uniforme y constante, de acorde a la visualización de la escena. Para ello, el tamaño de dicha representación será un porcentaje del tamaño de la escena a visualizar. De esta forma,

independientemente de lo cerca, o lejos que esté el punto de visión respecto al objeto, sus proporciones serán siempre las mismas.

Suele ser habitual que los objetos de un mismo tipo tengan una representación gráfica similar, por ejemplo, las pinturas se pueden representar con un cuadrado, las esculturas con un círculo, etc. Sin embargo, es habitual que pueden existir subtipos, y en ocasiones estos subtipos se representan con cambios en la representación gráfica, generalmente con el color, o con la imagen. De esta forma, las pinturas impresionistas pueden ser cuadrados azules y las pinturas cubistas cuadrados blancos, o pueden ser cuadros con la imagen del autor, o del mismo cuadro. Es decir, la figura o modelo del tipo se mantiene, pero cambian el color, o una imagen que se usa con dicha representación gráfica en función de otras propiedades. Este cambio, ligado a un subtipo, suele estar asociado al valor de una propiedad del objeto. En el ejemplo propuesto, la propiedad de los objetos de tipo "Pintura" podría ser el autor, o el tipo de corriente pictórica en la que están catalogados. Es interesante destacar, que en los trabajos revisados no se han tenido en cuenta ninguna de las normas ISO para gráficos de información pública

Existen normas ISO para el diseño de gráficos de información pública (ISO 2007), o diseño de símbolos en documentación técnica (ISO 2010). Sin embargo, en ninguno de los trabajos revisados durante la investigación se ha declarado el seguimiento expreso de estas directrices. Es aconsejable usar este tipo de directrices utilizadas en señales de tráfico y otros indicadores, así como aplicar varios tests recomendados por la misma agencia de estandarización (ISO 2014b).

#### [Características de las propiedades del objeto a representar.](#)

En el modelo, el objeto a representar es una instancia de un concepto perteneciente a un dominio en el. Dicho objeto tendrá varias propiedades, algunas de las cuales podrán ser usadas por el sistema de visualización para mostrar información, pero otras han de tener un valor si quieren ser representadas. Este tipo de propiedades que son obligatorias son aquellas que hacen referencia a las coordenadas espaciales y temporales del objeto.

En los objetos del dominio del patrimonio cultural, como se ha podido constatar durante la investigación, el tiempo y el espacio son propiedades fundamentales, que también definen al objeto. Además, un objeto de este dominio puede tener varias propiedades temporales y espaciales con diferente significado. En la Tabla 6 se muestran varias propiedades, tanto espaciales, como temporales, que los archivos de un museo suelen conocer, cuando es posible, que no es siempre. Algunos de los ejemplos más utilizados son, el lugar de producción, su ubicación actual y el lugar de procedencia, que es el último lugar donde estuvo antes de llegar a la ubicación actual. Sin embargo, puede haber muchos más, y con diferente significado, por ejemplo, restauraciones, reconstrucciones, cambio de propiedad, etc.

Además, sólo con los datos de ejemplo de la Tabla 6 ya se observa la heterogeneidad de la introducción de la información. Por ejemplo, para explicar un lugar, en ocasiones se cita una ciudad y entre paréntesis el país, en otra el nombre de un museo y la ciudad. Otro ejemplo es el usado para referenciar un momento del tiempo, a veces son dos años, separados por un guión, en otras sólo pone un año, y en otras se usa una frase con meses y un año.

*Tabla 6 Ejemplos de las diferentes propiedades temporales y espaciales asociadas a un objeto de patrimonio cultural.*

| <b>Nombre</b>              | <b>Descripción</b>                              | <b>Ejemplo : Cuadro de Guernica</b>    |
|----------------------------|---|--|
| Lugar de producción        | Lugar espacial donde se produjo.                | Se produjo en París (Francia)          |
| Momento de producción      | Momento del tiempo cuando se produjo.           | Se produjo entre Mayo y Junio de 1937. |
| Lugar de Exposición I      | Lugar donde estuvo expuesto                     | MOMA, Nueva York                       |
| Tiempo de Exposición I     | Tiempo durante el cual estuvo expuesto          | 1940 - 1981                            |
| Lugar de Exposición II     | Lugar donde estuvo expuesto                     | Casón del Buen Retiro, Madrid          |
| Tiempo de exposición II    | Tiempo durante el cual estuvo expuesto          | 1981 - 1992                            |
| Ubicación                  | Lugar dónde está en la actualidad               | Museo Reina Sofía, Madrid              |
| Tiempo en ubicación actual | Tiempo que está expuesto en la ubicación actual | 1992 -                                 |

Estos datos, que son sólo pequeños ejemplos de la heterogeneidad que puede darse, sirven para indicar la necesidad de usar un formato más uniforme, utilizando, si fuera posible algún tesaurus utilizado para clasificar los diferentes momentos del tiempo, o referencias espaciales. Para el caso de ubicaciones espaciales, sería de gran ayuda que también se usaran coordenadas numéricas, referenciando el sistema de coordenadas usado. Como hay que definir un mapeo entre las dimensiones de la escena gráfica y las dimensiones de la escena a representar, la conversión entre las coordenadas numéricas espaciales y las de la escena implica realizar un sencillo cálculo.

Lo que no hay duda es que un objeto puede tener varias propiedades espaciales y temporales que deben de ser identificadas de alguna forma especial, ya que estos valores pueden afectar a su representación.

Aunque menos importantes, otras prioridades también deben de ser identificadas en el objeto, porque también se suelen usar en la representación, como son: imagen, o imágenes del objeto, y etiqueta del objeto. Además, en la representación, también se pueden usar otras propiedades con fines informativos, como puedan ser el autor, nombre, la descripción, enlace con información adicional, etc.

### Niveles de detalle

En algunos sistemas, la escena es estática, no permite navegación, permitiendo mostrar los datos mediante una imagen, o un escenario tridimensional estático. Esta particularidad, es independiente de que el conjunto de datos que se visualizan sea también estático, o no. Si el conjunto de datos no es muy grande, o se desea mostrar una información, mediante un mapa de calor, o simplemente una visualización global. Esta forma de representar los datos es cada vez más poco habitual. La evolución de los sistemas de visualización tiende a que el usuario pueda interactuar con él, navegando por la escena, ya sea un mapa, una imagen, o un escenario tridimensional.

En los diferentes modelos de navegación por la escena, dispositivos e interfaces de usuario pueden ser muy varios y diversos. Este tema, depende mucho del tipo de usuarios que van a utilizar el sistema, así como de su contenido. Es un tema que sería interesante considerarlo, pero queda fuera del alcance de esta investigación. Lo que sí que se incluye son las consecuencias de esta navegación. A efectos prácticos, la navegación por la escena implica una actualización de sus contenidos conforme la cámara, o el punto de visión del usuario se mueve por ella. Una escena, tiene que tener la posibilidad de definir diferentes niveles de detalle. Estos niveles de detalle están directamente relacionados con la navegación, por qué conforme más se aproxime la cámara a una zona, si se quiere mejorar la visualización habrá que mejorar la calidad de esa parte.

El anterior razonamiento se entiende perfectamente con un mapa. Si la cámara está en un punto del espacio que permite ver un país entero, no tiene sentido tener que representar calles, o pequeñas regiones, con sus nombres, habría tanta información visual que sería imposible distinguir nada. Por ejemplo, sólo en España hay cerca de 900.000 calles (INE 2022). Con este conjunto de datos, en una resolución FULL-HD, de 1920x1080 píxeles, se dispondría de 2 píxeles de media por calle. Por ello, para trabajar con mapas, se suelen usar diferentes capas de información, donde a cada sección de la capa superior le corresponden varias secciones de la capa inferior con mayor resolución de información. En la Figura 31 se muestra un esquema de este proceso. A menor distancia, en este caso altura, del punto de visualización a los objetos a visualizar, se actualizan los objetos con mayor nivel de detalle, o mejor resolución de información, en la zona que está más cerca.

En el ejemplo del mapa se trata de una escena bidimensional, donde generalmente se muestran imágenes de mayor resolución de una zona más reducida. Mejorar el detalle de una zona puede implicar la aparición de nuevos objetos, o cambiar la forma de representarlos. En el caso de escenas tridimensionales es muy similar. En ese caso, a menor distancia, los objetos tridimensionales se visualizan con más detalle, generalmente con más vértices, o mejorando la resolución de las texturas que se aplican sobre ellos.



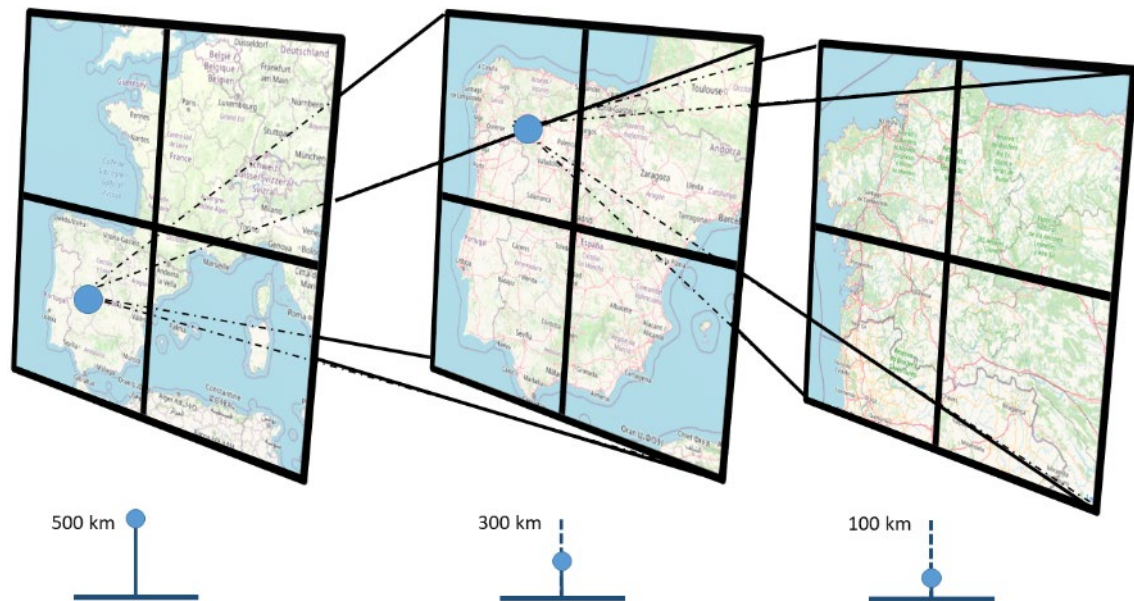


Figura 31 Los contenidos de la escena se actualizan por otros con mejor resolución conforme disminuye la distancia del punto de observación. El tamaño de la escena en el dispositivo de visualización se mantiene.

Una forma de poder contemplar esta actualización de la escena es mediante la definición de un número de capas de detalle. Asociando este número a la escena, el sistema puede ofrecer medios interactivos para modificarlo, generalmente mediante los controles de navegación. Esto conlleva actualizar la escena de acorde al nivel de detalle activo. La demarcación del número de niveles también determina el máximo nivel de detalle que se puede alcanzar en la escena. En el caso de que este nivel no actuara a nivel global, sino que fuera a nivel individual de cada objeto de la escena, este número tomaría un valor nulo, o cero. En este último caso la mejora del detalle de cada elemento se calcularía de forma particular en cada objeto. La activación de un nivel de detalle, u otro, suele ser transparente para el usuario, pero en ocasiones puede ser interesante ofrecer una interfaz para definir una mayor, o menor calidad en la visualización. Esto puede realizarse definiendo la capa de detalle en el momento de partida.

#### Línea de tiempo.

Tal y como se ha descrito en el capítulo 4 de esta memoria, la línea de tiempo es un elemento de la interfaz de la escena gráfica, que está presente en varios sistemas de visualización de datos espacio temporales. El objetivo es poder acotar los objetos de la escena visual, relacionados con un intervalo de tiempo. De esta forma, igual que navegando por la escena, se define una región del espacio, mediante esté control se define una región de tiempo sobre la que se desea visualizar información.

Mediante este control se establece un intervalo temporal, ya sea de forma dinámica, o preestablecida por el sistema. Una vez fijado, el contenido de la escena se puede actualizar de dos formas:

- Mostrando los datos cuya propiedad temporal indique pertenencia a ese intervalo completo, o a un fragmento del mismo, cuyo tamaño es fijado por la granularidad del intervalo, que está prefijada por el sistema. Por ejemplo, si el intervalo elegido comienza en el siglo X, y termina en la actualidad, siendo la granularidad de un siglo, se podrán visualizar los datos de cada siglo, desde el X, hasta el XXI. Ejemplos de este tipo de controles son mostrados en la Figura 11 y en la Figura 12 .

- Mostrando de forma simultánea, los datos pertenecientes a varios fragmentos consecutivos del intervalo, de forma simultánea. Por ejemplo, si el intervalo elegido es desde el siglo X, hasta el momento actual, se podrían ver todos los datos del intervalo, en cada siglo, de forma simultánea. Ejemplos de sistemas que permiten la visualización simultánea de los datos en varios momentos del tiempo pueden observarse en la Figura 13, también en la Figura 14 y en la Figura 20.

Esta definición de intervalos temporales y visualización de datos pertenecientes a dicho intervalo, permite analizar la fluctuación de la información con el transcurso del tiempo.

En los datos de patrimonio cultural, el tiempo es un dato muy importante, y es muy habitual que un concepto del dominio de datos tenga varias propiedades temporales. Por ejemplo, un utensilio expuesto en un museo, puede tener una fecha de producción, y otra de cuando se encontró. También podrían almacenarse otras fechas, como las fechas con los cambios de propiedad de dicho objeto a lo largo de su historia, o fechas de restauración, etc. Esto implica que sea fundamental definir qué propiedad temporal del objeto es la que está relacionada con la línea de tiempo.

Además, hay varias formas de representar los diferentes tipos de línea de tiempo y de seleccionar el inicio y fin del intervalo:

- Estática y lineal (Kraak 2005), un ejemplo puede verse en la Figura 11.
- Jerárquica y dinámica (Wang, C., Ma, and Chen 2018a), de la que se muestra un ejemplo en la Figura 12.

#### *6.2.2. Agrupamiento de objetos.*

En el caso de que el conjunto de datos a visualizar sea muy grande es muy probable que varios de los datos estén ubicados en posiciones espaciales cercanas. Cuando esto sucede, aunque el número de datos no sea muy grande, suele provocar un solapamiento gráfico de su representación en la escena, ya que se están posicionando en la escena varios objetos gráficos en el mismo lugar, o en lugares espacialmente muy próximos. Un ejemplo de este problema se muestra en la Figura 32.

En el caso del ejemplo de la Figura 32 el solapamiento se consigue al alejarse en extremo de la escena. Es un caso extremo, pero válido como ejemplo visual. En este sistema de visualización realmente el solapamiento gráfico de los marcadores de los objetos no es un problema, ya que tiene muy pocos objetos, alrededor de 50, y están lo suficientemente espaciados en la ciudad, que es el espacio de visualización del sistema. Sin embargo, en un conjunto con más datos, este problema suele presentarse con mucha facilidad. Volviendo al mismo sistema, si en lugar de trabajar con 50 objetos, fueran 5000, sería muy fácil que los objetos se solapasen, además de que prácticamente no dejarían ver el mapa de la ciudad, que es la base de la escena.





Figura 32 Ejemplo de solapamiento de objetos cuando no hay agrupación. Al alejar el punto de visión (izquierda) sólo se distingue una nube de marcadores que no aporta información. Fuente: (Helsingin kaupungin museo 2015)

Para resolver este problema se recurre al agrupamiento de objetos. Esta agrupación se puede hacer de varias formas, pero las más utilizadas son las siguientes:

- Agrupación en torno a un marcador representante. Consiste en ir agrupando los diferentes objetos de una región, para eliminarlos de la escena y sustituirlos por un marcador representante, que suele ubicarse en el centro de la región definida por el grupo, punto espacial en el que generalmente no hay ningún objeto real, o bien sobre la posición del objeto del grupo más cercano a dicho centro, estando pues situado sobre la posición de un objeto real. Este proceso se repite en todos los niveles de detalle, de forma, que en un nivel X de máximo, los marcadores representantes y los grupos se conforman a partir de objetos del conjunto de datos, pero en un nivel X-1, con menor detalle, tanto los marcadores, como los grupos se definen partiendo de los marcadores representantes del nivel X, y así recursivamente hasta llegar al nivel de menor detalle.
- Mapas de calor. Se crean las agrupaciones de datos, de forma similar a la anterior. Se asocian regiones de la escena a cada grupo, o grupos, en función de su densidad. Cada zona se le asigna un color, que suele tener más intensidad en función del número de datos que contiene, o incluso cambiar de color, en función del número de datos. Esta forma de agrupamiento es estática generalmente. Una vez las regiones están definidas y coloreadas, no se modifican, aunque se acceda mediante la navegación a un nivel de detalle diferente. De esta forma, permite ver la distribución de objetos en la escena de forma sencilla, pero no permite analizar cada dato de forma independiente.

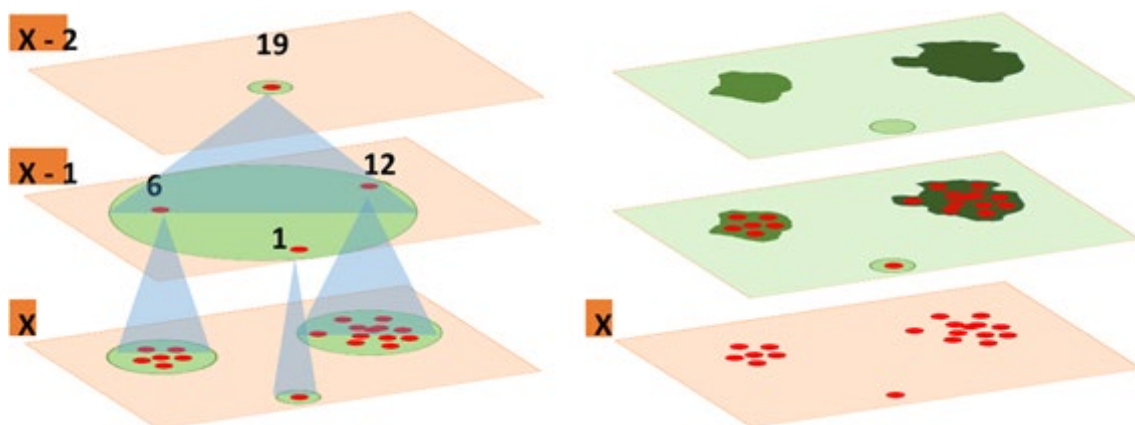
En la Figura 33 se muestra un esquema de los dos tipos de técnicas de agrupamiento. En la parte izquierda la de marcadores representantes. En el nivel de más detalle X, se visualizarán todos los objetos. En el nivel de detalle X-1, sólo se verían 3 objetos, donde cada uno es un marcador representante de un grupo del nivel inferior. En el nivel de menor detalle (X-2) se mostraría en la escena un único objeto, que es un marcador representante de los 3 marcadores del nivel X-1. La parte derecha tiene un esquema de cómo se obtiene un mapa de calor. Tras agrupar los puntos, se marcan las zonas que los contienen con mayor intensidad de color en función del

número de puntos que contiene. Generalmente, sólo se visualizan las regiones de color en toda la navegación. Algunos sistemas de visualización que usan esta técnica de agrupamiento permiten visualizar los objetos independientes al llegar a un nivel alto de detalle. En cualquier caso, esta técnica se usa principalmente para visualizar una distribución global de los datos en la escena.

La técnica de agrupación de objetos con marcadores representantes, para aportar información visual adicional, suele aplicar visualizaciones especiales sobre estos. Estas modificaciones se aplican de forma aislada, o conjunta. Algunas de las más utilizadas son:

- Cambiar el tamaño del marcador representante en función del número de objetos que representa. Esta técnica tiene el problema de que marcadores representantes pueden solaparse con otros marcadores, al ocupar un espacio importante de la escena en el caso de que tengan muchos objetos.
- Aplicar un color sobre el marcador representante, en función del número de objetos que contiene, generalmente se aplica un color para un rango de valores. De esta forma, al ver la escena se sabe dónde hay agrupaciones grandes de objetos rápidamente.
- Indicar en el marcador un número que define el número de objetos que contiene. Esta técnica aporta mucha información, pero es más costoso ver la distribución de los datos a nivel global.

Lo habitual es utilizar la técnica de aplicar un color sobre el marcador representante, además de incluir el número de datos. De esta forma se obtiene información a nivel global, además de poder saber el número de objetos que hay en cada grupo.



*Figura 33 Esquema de técnicas de agrupación de objetos (clustering) más usadas. A la izquierda, con marcadores representantes y a la derecha mapas de calor. Fuente: Elaboración propia.*

Los sistemas de visualización suelen agrupar los objetos independientemente del concepto, por ejemplo, en un conjunto de datos de obras de arte, pueden agrupar cuadros con esculturas. Esta forma de proceder tiene la ventaja de que la escena queda más clara, pero el inconveniente de que se pierde información, se desconoce en un grupo la cantidad de elementos que hay para cada tipo, o concepto.

En función del conjunto de datos y de la finalidad del sistema puede ser más interesante diferenciar las agrupaciones por concepto, incluso aplicar varios tipos de agrupación de forma simultánea, mapas de calor y marcadores representantes. Por ello, lo mejor es que la escena tenga la posibilidad de definir una técnica de agrupamiento y todas sus características (marcador

representante, técnica visual sobre el marcador, etc.), pero además que exista la posibilidad de definir, de forma opcional, sobre cada concepto, su técnica de agrupamiento y sus características.

De esta forma, por ejemplo, se podría realizar un sistema de visualización que mediante mapas de calor mostrará el número de restauraciones de obra de arte, y con marcadores representantes diferenciados mostrará la distribución de cuadros y esculturas. Estas combinaciones pueden parecer caprichosas a priori, pero en ocasiones su uso puede ayudar a detectar patrones de comportamientos de los objetos desconocidos, o que de otra forma serían muy costosos de obtener. Sin embargo, hay que tener mucho cuidado, cuando se abusa de diferentes técnicas de agrupamiento, ya que se introducen demasiados elementos diferentes en la escena y esto hace que sea mucho más complejo su análisis. Lo habitual es que la escena tenga una técnica de agrupamiento para todos los objetos, que podría ser compatible con agrupamiento de objetos por mapa de calor. Por ejemplo, usar dos mapas de calor en una misma escena puede ser muy difícil, y generalmente imposible de analizar. En la Figura 34 se muestra una captura de pantalla de la herramienta Maptive (Maptive 2022), en la que se combinan dos técnicas de agrupamiento, en este caso con el mismo concepto.

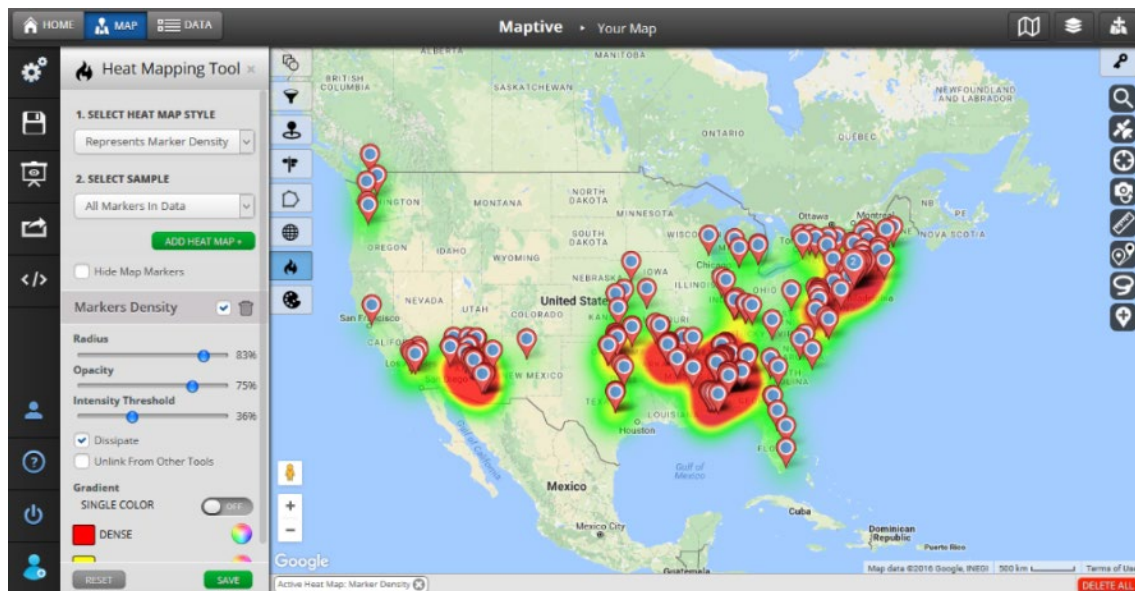


Figura 34 Ejemplo de combinación de mapa de calor y agrupamiento por marcadores. Fuente: Maptive (Maptive 2022)

### 6.2.3. Procesos de interacción

Una vez se han expuesto los aspectos visuales de los elementos de la escena, se procede a explicar procesos de interacción que pueden implicar la actualización de parte de la escena gráfica, o su totalidad. Estos procesos forman parte del modelo, y por lo tanto están relacionados con los conceptos y propiedades de los elementos.

#### Navegación.

Existen un conjunto de procesos relacionados con la navegación por la escena. Estos afectan a dos aspectos clave de la representación de la escena, los niveles de detalle y la agrupación de objetos. Estos procesos son los que permiten modificar el punto de visión en la escena, que son:

- Zoom, donde el punto de visualización se aproxima, o se aleja de la escena.
- Panning, o deslizamiento, en el que el punto de visualización se desplaza sobre un plano.

- Rotación, en el que el punto de visualización rota sobre un eje. Este proceso de navegación tiene sentido principalmente en escenas tridimensionales. En escenas de dos dimensiones sólo puede realizarse en un eje perpendicular al plano de visión, lo que haría girar la escena en contra, o a favor de las agujas del reloj.
- Movimiento a una zona de la escena. Cambio de posición del punto de observación, de forma directa, o interpolada. Generalmente se usa para acercarse a un objeto, o conjunto de objetos.

Cada escena debe de poder definir las posibilidades de navegación y especificar si un cambio generado por estos procesos implica un cambio de nivel de detalle de la escena. En las escenas bidimensionales el proceso de zoom suele conllevar este cambio. En las escenas tridimensionales este cambio está vinculado con el proceso de zoom y también con el de panning, o desplazamiento.

#### Filtrado.

Un sistema de visualización, sobre todo cuando tiene un conjunto de datos importante, suele ofrecer la posibilidad de filtrado de sus datos. Esto tiene varias ventajas:

- Permite visualizar en la escena los objetos en los que el usuario está realmente interesado. Los datos que no son objeto de estudio del usuario generalmente aportan ruido y dificultan la visualización, mediante el filtrado estos datos dejan de visualizarse.
- Mejora el rendimiento del sistema, al menos a nivel gráfico. Al tener que representar y trabajar con menos datos, la productividad de los procesos mejora. Esto se aprecia sobre todo en conjuntos de datos muy grandes, con varios cientos de miles, o millones de datos.

Estos procesos, conllevan dos tipos de acciones para el desarrollador:

- Ofrecer una interfaz de usuario para que pueda realizar el filtrado. Esta interfaz debe de permitir un filtrado incremental.
- Una actualización de la escena a nivel global. El filtrado de datos no se reduce únicamente a eliminar de la escena de visualización unos objetos. También implica actualizar agrupaciones, tamaños, colores, número de datos, representación de relaciones, así como los efectos de la línea de tiempo. A todos los efectos es como un cambio en el conjunto de datos que suele conllevar una actualización global de la escena.

Todos los conceptos deben de definir cuáles de sus propiedades son susceptibles de ser filtradas.

El sistema de filtrado puede tener tres opciones para presentar los posibles valores sobre los que filtrar:

- Que estén previamente definidos, a nivel de propiedad, y por lo tanto aparecerá un desplegable, o una lista con las diferentes opciones por las que buscar.
- Que el sistema obtenga todos los posibles valores del conjunto de datos y que automáticamente muestre un desplegable, o una lista con los valores a seleccionar.
- Que no ofrezca ningún valor y que el usuario tenga que introducir los valores manualmente.

#### Visualización de relaciones.

En un sistema de visualización de datos espacio temporal suele ser casi directo, o muy sencillo visualizar qué objetos están relacionados con otros en función de sus coordenadas espaciales, o temporales. Sin embargo, además de estas propiedades, también puede ser interesante saber

qué objetos están relacionados con otros en función de otras propiedades, y visualizar de alguna forma dicha relación.

Tal y como se describió en la sección “Visualización de datos” del capítulo 4, las técnicas más utilizadas para mostrar relaciones entre los datos de un sistema de visualización, son las siguientes:

- Marcar los objetos relacionados con un color, o marcador gráfico característico.
- Unir los objetos relacionados entre sí mediante líneas, o arcos, cuyo grosor, o intensidad de color, puede variar en función del número de datos que están relacionados, en el caso de que sean agrupaciones.

En la Figura 35 se muestra un esquema de la aplicación de dichas técnicas. El objetivo es ver los objetos relacionados con el objeto B, a través de una propiedad (azul), así como los objetos relacionados con A, a través de otra propiedad (verde). Los objetos, como las agrupaciones están marcados en rojo. En el caso superior se usa la primera técnica para representar relaciones, consistente en usar un marcador gráfico con los objetos relacionados. Cuando no están agrupados, cada objeto tiene un círculo con el color verde, o azul, en función de que tenga, o no, el mismo valor para dicha propiedad. Al agruparse, en la parte de la derecha, dicho círculo se dibuja sobre la agrupación, con mayor, o menor grosor en función del número de objetos relacionados. En el caso inferior se usa la técnica de dibujado de líneas, o arcos, entre los objetos relacionados. En el caso de agrupaciones, la línea sigue estando, con mayor grosor conforme exista un mayor número de objetos relacionados.

Las dos técnicas son aceptables siempre que el conjunto de datos sea reducido, aunque la segunda técnica permite visualizar las relaciones entre objetos de una forma más clara.

Los problemas de la técnica que usa marcadores gráficos para identificar los objetos relacionados son:

- En el caso de que se visualicen varios tipos de relaciones, puede ser costoso ver los objetos relacionados. Hay que tener en cuenta que un mismo objeto puede estar relacionado con dos, o tres objetos a la vez, lo que implica aplicar varios marcadores al mismo objeto. Esto puede suceder con poca frecuencia a nivel individual, sin embargo, la frecuencia puede ser muy alta en el caso de las agrupaciones.
- Si hay objetos relacionados que no están en la parte de la escena que se está visualizando, habrá que incluir marcadores especiales en la escena para indicar que hay más objetos relacionados, aconsejando usar zoom, o panning para encontrarlos, lo que aportaría más ruido en la escena de visualización.

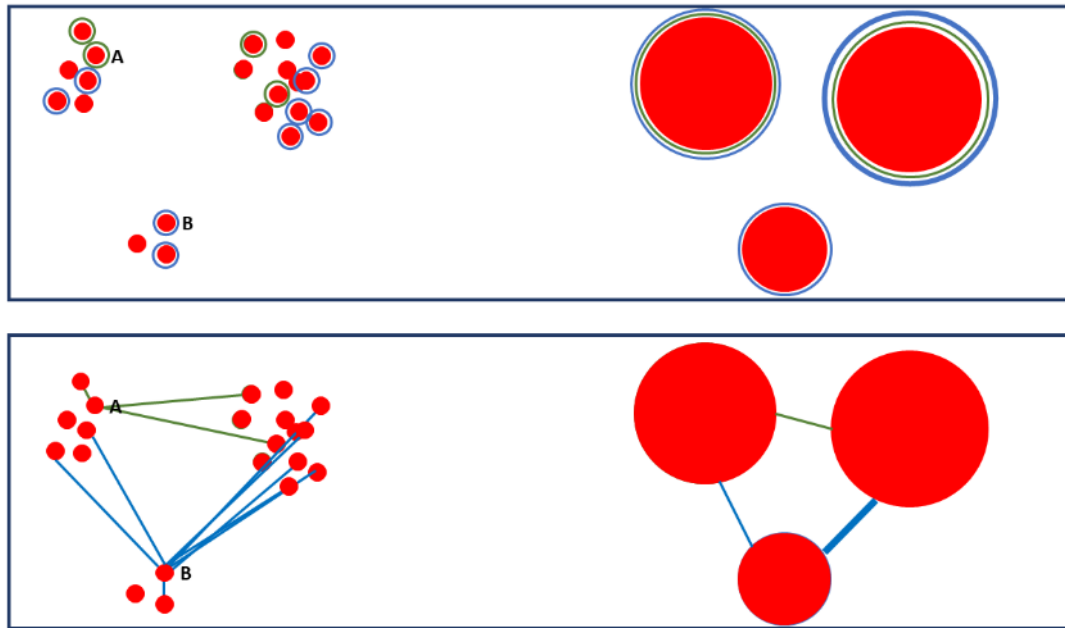


Figura 35 Esquema de aplicación de técnicas para mostrar los objetos relacionados con los objetos A y B, a través de diferentes propiedades (marcadas en azul y verde). En la parte superior se usa la técnica de marcar los objetos relacionados gráficamente y en la parte inferior unirlos mediante líneas, o arcos. Fuente: Elaboración propia.

Los problemas de la técnica de dibujado de líneas, o arcos, para identificar los objetos relacionados son:

- El dibujado de líneas, o arcos, ocultan una parte importante de la escena, impidiendo ver otros elementos de la escena. El problema se agrava conforme haya más datos relacionados.
- Al agrupar los objetos es muy sencillo conocer qué agrupaciones están relacionadas entre sí. Sin embargo, puede ser muy costoso ver qué objetos en concreto están relacionados, ya que para hacerlo hay que deshacer las agrupaciones. Si una vez deshechos los grupos, no están en la misma parte de la escena que se está visualizando, obliga a navegar por la escena para poder averiguarlo.

En cualquier caso, es obvio que es muy complejo usar este tipo de técnicas cuando los conjuntos de datos son muy grandes. Si hay muchos objetos relacionados, hay una gran parte de la escena oculta por los marcadores de la relación, además de que se somete a una carga importante al sistema gráfico, debido a la incorporación de nuevos elementos gráficos que representar.

La aplicación de estas técnicas suele estar condicionada a que el número de objetos sea menor que un número dado, gestionado por el sistema, lo que obliga al usuario a realizar filtros de los datos para poder aplicarlas.

#### Selección temporal

La selección temporal, tal y como se ha descrito en la sección anterior, en el apartado “Línea de tiempo”, requiere de dos procesos: definir un intervalo temporal sobre una propiedad del objeto, y establecer el tipo de visualización, simultánea, o puntual. En el caso de que el tiempo sea un factor para la representación gráfica del objeto en la escena el segundo proceso no se lleva a cabo, ya que el tiempo siempre se visualiza de forma simultánea.

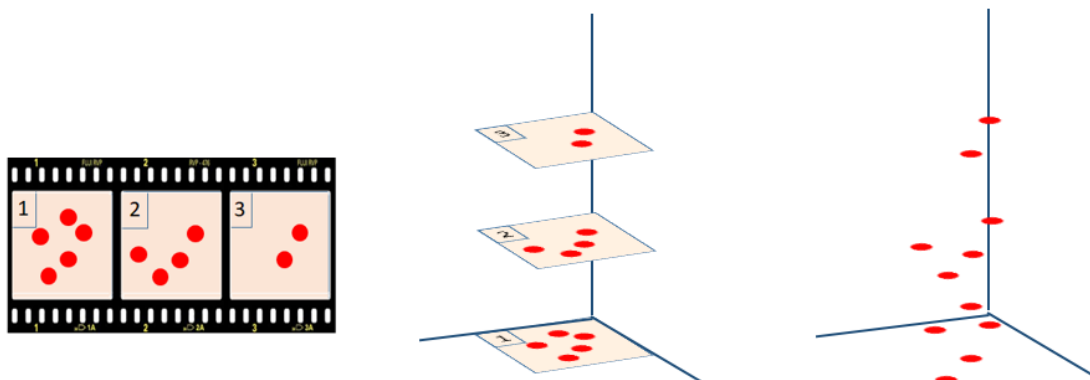


El primer proceso simplemente va a realizar un filtrado, de forma que sólo serán visibles en la escena aquellos objetos cuyo valor de la propiedad temporal esté dentro del intervalo fijado.

En el caso de aquellas escenas donde el tiempo no forma parte de la representación gráfica, si la visualización es puntual, la escena no se ve afectada de forma adicional, sin embargo, si es simultánea sí que se ve afectada y de forma compleja. En este caso, generalmente se entra en un modo de navegación especial, donde el usuario suele tener menos interacción, donde se seguirá uno de los siguientes patrones:

- Una visualización bidimensional, que permite ver la fluctuación del tiempo como si fuese un vídeo, donde cada elemento de la escala de tiempo se corresponde con un fotograma. En este caso, el usuario puede detener la reproducción, retroceder, avanzar, e incluso reducir, o incrementar la velocidad de reproducción. Realmente no es una visualización simultánea, ya que no se pueden visualizar dos fotogramas a la vez, pero dada la rapidez de la visualización se ha incluido en esta categoría, debido a que permite ver los cambios cercanos en el tiempo de forma sencilla y rápida.
- Una visualización tridimensional, donde mediante un hipercubo, el usuario visualiza los datos en cada momento de la escala temporal. El hipercubo puede ser sencillo, donde los datos se muestran de forma continua, o incluir secciones, o cortes, donde sólo se visualizan los datos de cada sección. En este caso, la navegación se simplifica, permitiendo acercarse, o alejarse del hipercubo, además de desplazarse hacia un extremo, u otro del eje del tiempo.

En la Figura 36 se representan dos esquemas para facilitar la comprensión de los diferentes modos de visualización de tiempo de forma simultánea.



*Figura 36 Esquemas de visualización simultánea. A la izquierda se simboliza el modo video, a la derecha dos hipercubos, el primero con capas, y el segundo sencillo. Fuente: Elaboración propia.*

### 6.3. PROBLEMAS DE VISUALIZACIÓN

Como se ha comentado en el capítulo 4, en la visualización de datos espaciotemporales hay varios problemas para los que no hay una solución concreta, sino generalmente varias, siendo una más adecuada que el resto en función de varios factores.

A continuación, se exponen muy brevemente los problemas anteriormente expuestos, para describir varias posibles soluciones, para solucionar este problema, o al menos reducir sus



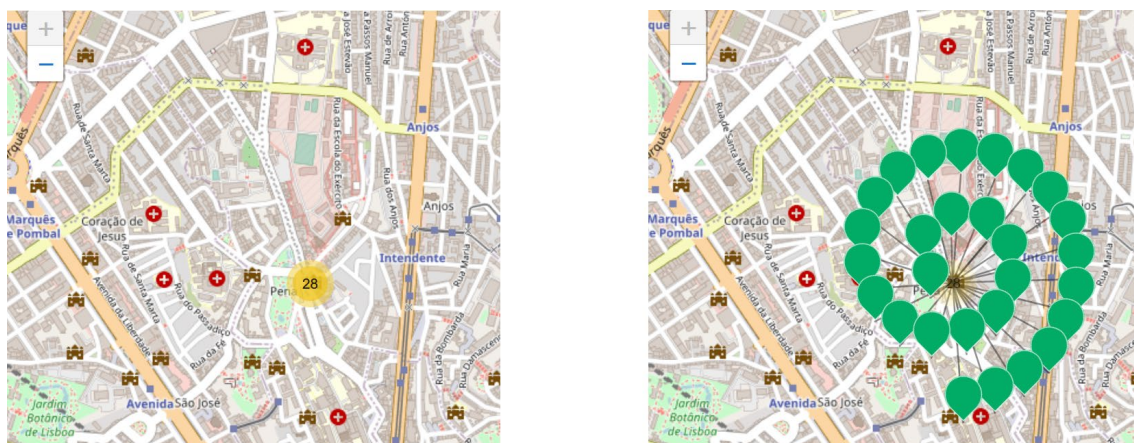
consecuencias. Las diferentes alternativas, así como sus propiedades habrá que incluirlas en la escena visual.

### 6.3.1. Agrupamiento de objetos

El motivo de usar el agrupamiento es que es una técnica muy útil y necesaria cuando hay un conjunto de datos grande, pero hay situaciones en las que el agrupamiento no soluciona el problema de visualizar muchos datos. Se trata del caso particular en el que todos los objetos de un conjunto están en el mismo punto espacial. Cuando esto sucede, el agrupamiento siempre es el mismo, o al menos siempre tiene agrupado a los objetos de ese conjunto. Por lo tanto, independientemente del nivel de detalle activo, todos esos objetos estarán siempre agrupados. Además, aunque el agrupamiento se disuelva, visualmente se verá un único punto, ya que están todos solapados.

Este problema suele suceder con frecuencia, en el ámbito de patrimonio cultural, sobre todo al representar gráficamente la ubicación actual del objeto. Esto es así, porque este patrimonio suele estar en unos pocos puntos de una ciudad, generalmente museos y colecciones privadas. Los cuales, además, suelen tener miles de objetos en su fondo, y la ubicación es la misma, la dirección del museo.

El principal problema de tener varios objetos localizados en el mismo lugar espacial es que hace que sea imposible acceder a la información de un objeto en concreto, o conocer cuántos, o principalmente, que puntos de ese conjunto están relacionados con otros puntos del conjunto de datos global. Revisando las técnicas de visualizar relaciones, comentadas en la sección anterior, en el apartado “Visualización de relaciones”, si en el grupo de datos que están todos ubicados en la misma posición, existen objetos relacionados con otro objeto de la escena, no se podrá conocer cuáles son. Se podrá saber si hay, y si son muchos, o pocos, pero no qué objetos son los relacionados.



*Figura 37 Ejemplo de disposición de grupo en espiral. Usada cuando los objetos están en una misma posición espacial. Fuente: Captura de pantalla de la página web del Proyecto Semap.*

Ante este problema no hay una solución perfecta y coherente con el sistema de visualización. Sin embargo, existen medidas que pueden ser útiles para mejorar la situación y ofrecer al usuario más información. Todas las acciones parten de la base de que el agrupamiento está compuesto por un conjunto de datos en el que todos los elementos están en la misma posición espacial. En ese caso se usa un marcador especial, que lo diferencia de otros marcadores, y al interactuar con dicho marcador suceden las siguientes acciones:

- Visualización de datos en espiral. Los datos del conjunto se disponen en espiral sobre el centro del grupo, y se unen con líneas, u otros marcadores con el fin de indicar su relación y distinguirlos de otros objetos de la escena. Esta solución es válida únicamente en el caso de que el número de objetos sea reducido. El número depende mucho del tipo de escena, pero si el número de datos es muy grande, ocupa gran parte de la escena al desplegarse y es complejo para el usuario manejar la información con esta distribución. En la Figura 37 se expone una captura de pantalla del proyecto Semap en la que se usa la disposición de grupo en espiral. En este ejemplo, el grupo tiene un total de 28 objetos que se muestran con la forma de una espiral.
- Visualización de una lista con los datos del grupo. Esta opción permite manejar un número de datos mayor, aunque tampoco demasiado grande, consiste en mostrar en la escena una ventana con la lista de los objetos del grupo. Cada elemento de la lista se corresponde con un objeto, en el que se incluye su nombre, además de una imagen con su marcador, o color, y algún marcador, generalmente color para indicar los objetos con los que está relacionado. La lista puede estar categorizada, incluso soportar una jerarquía, que tiene que estar vinculada con alguna propiedad de los objetos.
- Visualización gráfica e interactiva de los datos del grupo, mediante alguna técnica interactiva alternativa, agrupándolos por jerarquías. En esta solución, se requiere mostrar una ventana, que puede formar parte, o no de la escena. En esta ventana se mostraría una distribución dinámica y jerárquica de los datos, lo que implica realizar agrupaciones en función de alguna propiedad. Un ejemplo de este tipo de visualización alternativa se muestra en la Figura 38, en ella se muestran varios objetos categorizados en círculos siguiendo una estructura jerárquica, en la que un objeto puede pertenecer a varios grupos. Al poderse navegar entre los diferentes grupos y subgrupos de forma interactiva soporta un gran número de datos. El inconveniente de esta visualización es que se pierde la representación global. Este impacto negativo puede ser menor si los diferentes grupos y subgrupos tienen marcadores gráficos en concordancia con el sistema de visualización.

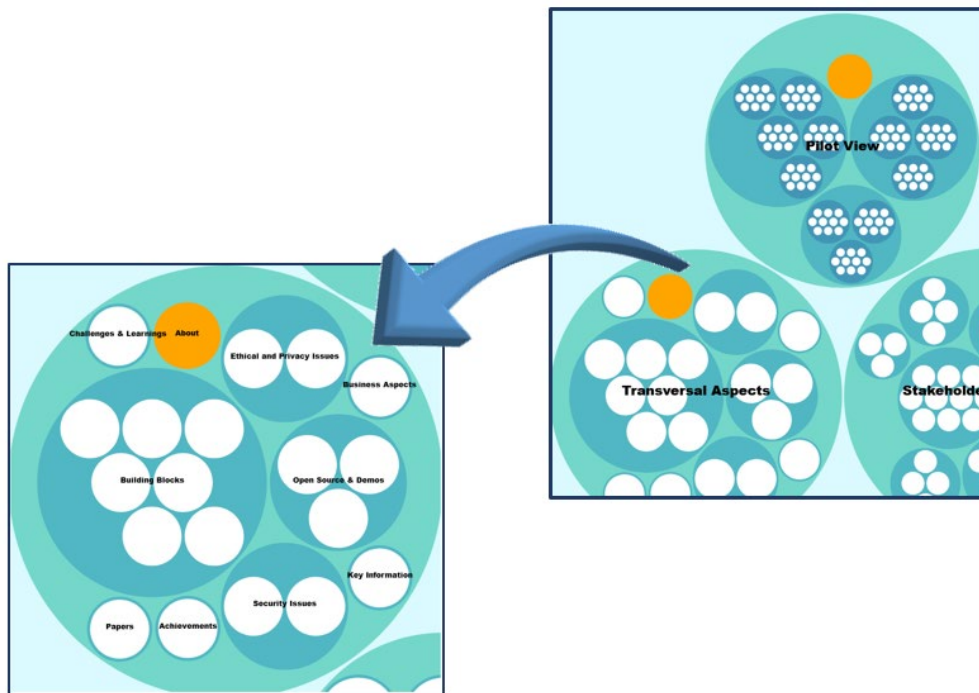


Figura 38 Ejemplo de navegación jerárquica interactiva sobre objetos categorizados. Fuente: Adaptado de Bigmedilytics Blueprint (Bigmedilytics 2021)

### *6.3.2. Representación de las relaciones de un objeto.*

La visualización de objetos relacionados con uno previamente seleccionado, a través de las diferentes propiedades de los objetos es un tema complejo. La aplicación de técnicas de visualización para mostrar esta información suele limitarse a conjuntos de datos dónde el número de objetos, o de relaciones, a representar no es muy grande.

Estas técnicas son muy útiles para realizar varias actividades de investigación, educativas, etc., pero requieren conocer uno, o varios objetos, con una serie de restricciones, generalmente caracterizadas por unos valores concretos en alguna, o varias de sus propiedades.

Por lo tanto, implican un número de interacciones importantes, además de disponer de un objetivo concreto. Sin embargo, cuando se quieren descubrir patrones de comportamientos poco habituales, estas técnicas no son muy útiles. Precisamente por ser poco habituales, generalmente se desconoce qué propiedades, o que valores buscar, así como sobre que objetos partir en esta búsqueda de información.

En este ejercicio, sería ideal poder ver en cada objeto de la escena, cuantos objetos de la misma, o que porcentaje de ellos, tienen una relación en función del valor de sus propiedades. Si se consigue mostrar esta información de forma visual, se podría descubrir de forma relativamente rápida, objetos que siguen un comportamiento poco habitual. Un ejemplo muy habitual de objetos que cumplen esto son los objetos que tienen muchas, o muy pocas relaciones con el resto de objetos. Por ejemplo, si fuera posible visualizar las relaciones entre los diferentes objetos de la escena, se podría apreciar que generalmente los objetos tienen un número similar de relaciones con otros en algunas propiedades. Pero si al visualizar se observa que un objeto no está relacionado con ningún objeto a través de esa propiedad, o está relacionado con muchos más objetos que la mayoría, se trataría de un objeto que no se comporta igual que los demás. Este objeto puede ser interesante analizar, y partiendo de él, ya se pueden aplicar las técnicas generales para descubrir el motivo de esta anomalía. Para que esto fuera posible, habría que disponer de un marcador especial sobre el objeto, que indicara para cualquiera de las propiedades relacionables del objeto, que porcentaje de objetos de la escena tienen el mismo valor.

Esto es complejo por dos motivos, primero obtener esta información es costosa, ya que hay que analizar la información de todos los objetos entre si, lo que implica un coste cuadrático. Además, visualizar esta información de forma gráfica, clara y que aparezca en cada objeto, es un reto importante. Es mucha información y además hay que incluirla en todos los objetos que se están visualizando. Por ejemplo, si un objeto tiene cuatro propiedades por las que buscar objetos que estén relacionados, tiene que indicar las cuatro propiedades y además cuatro porcentajes.

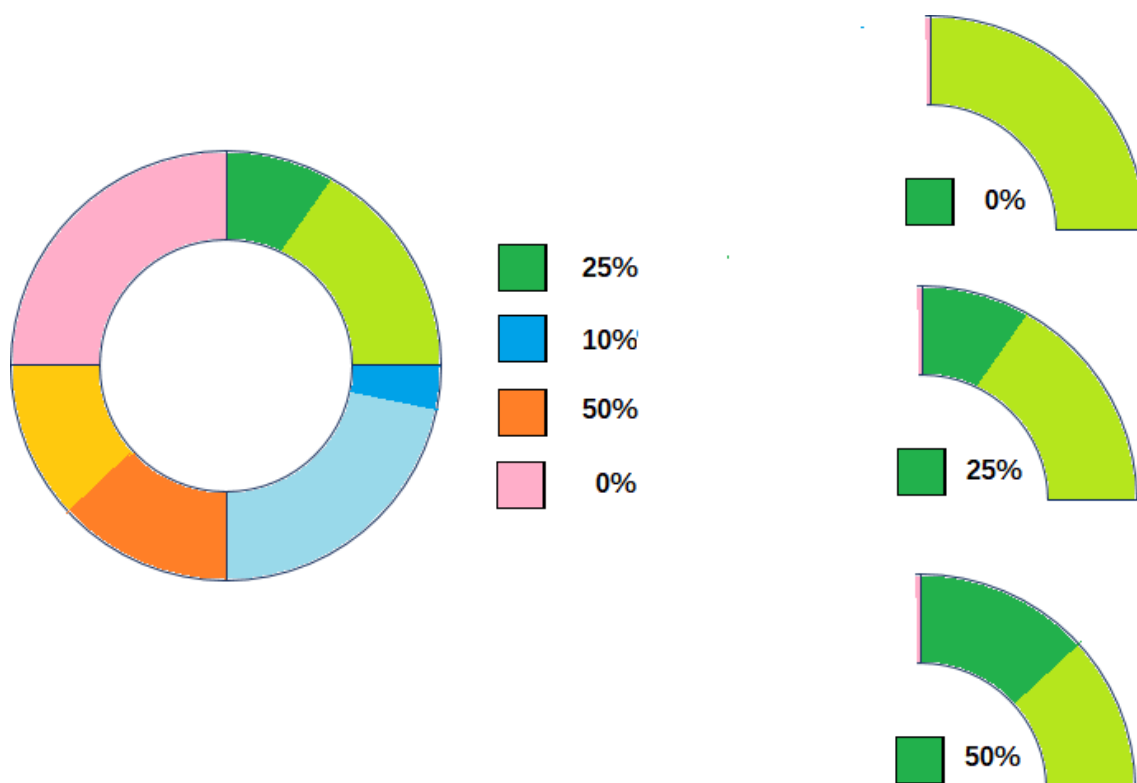


Figura 39 Esquema de estructura en anillo para indicar el porcentaje de objetos relacionados. Fuente: Elaboración propia.

Para solucionar este problema se propone una estructura en anillo, que se pondría idealmente alrededor del objeto, o lo más cerca posible. Esta estructura estaría dividida en tantas zonas como propiedades relacionables tiene el objeto. Cada zona estaría rellena de un color claro, y en función del porcentaje de objetos la zona se rellenaría del mismo porcentaje del color de relleno, pero en su versión oscura. En la Figura 39 se muestra un esquema de la estructura descrita. En este caso el objeto tendría 4 propiedades relacionables, que se corresponden con los colores verde, azul, naranja y rosa. En la escena habría un 25% de objetos con el mismo valor en la propiedad verde, un 10% con el mismo valor en la propiedad azul, y un 50% en la naranja. Además, se muestra que no hay ningún objeto con la propiedad marcada con el color rosa. En la figura, en la izquierda, también se muestra un esquema de cómo se visualizará la banda del círculo correspondiente a la propiedad marcada con el color verde.

### 6.3.3. Incerteza

La incerteza es una realidad que es inherente a los datos de patrimonio cultural. Con una frecuencia importante, este tipo de datos pueden presentar varios valores en sus propiedades. Existen varios tipos de incerteza en función de las propiedades que se desconozcan, de objeto, geoespacial, temporal, de tipo, etc. (Windhager, F., Federico et al. 2019). En la Tabla 7 se enumeran los diferentes tipos de incerteza y las causas a las que es debida.

Es complejo usar una representación lo suficientemente visual para ver que es un dato con incerteza, pero que no introduzca demasiado ruido en la escena. Además, el problema puede agravarse si un mismo objeto presenta incerteza en varias propiedades.

Tabla 7 Tipos de incerteza en los objetos de patrimonio cultural y sus motivos.

| Incerteza    | Motivos   |
|--------------|---|
| De objeto    | Generalmente por problemas en catalogación.   |
| Temporal     | Por diferentes criterios a la hora de atribuir el tiempo y falta de pruebas documentales.                               |
| Geoespacial  | Por diferentes criterios a la hora de atribuir el espacio y falta de pruebas documentales.                              |
| De tipo      | Principalmente debidas a problemas de clasificación, uso de diferentes vocabularios, uso de sistemas propietarios, etc. |
| Proveniencia | Falta de información de la trayectoria espacio temporal que ha seguido un objeto desde su producción.                   |

Por ello, basándose en las ideas de los autores citados en este apartado, se usará el color para indicar la incerteza, pero para no incorporar más elementos al marcador, simplemente se dispondrá sobre el marcador una máscara que hará que el color sea más claro. Esta sería la propuesta genérica, pero ha de existir la opción de que se pueda especificar un marcador específico para indicar la incerteza. En la Figura 40 se muestra un ejemplo gráfico de cómo resaltar la incerteza de un objeto haciendo más claro el marcador. En el caso de objetos 3D, tal y como se muestra a la izquierda de la figura se cambiarían las propiedades de los materiales del objeto para que fuese más claro. En el caso de objetos 2D, como aparece a la derecha, se trataría simplemente de cambiar los valores de brillo y contraste para conseguir el mismo efecto. Con esta propuesta, al mantener el marcador no habría que hacer importantes cambios visuales, ni se introduciría más complejidad en la escena. Además, el color más claro, junto al oscuro, suele aportar la idea de que el objeto tiene algún problema asociado, ya que parece que esté descolorido, o desgastado.

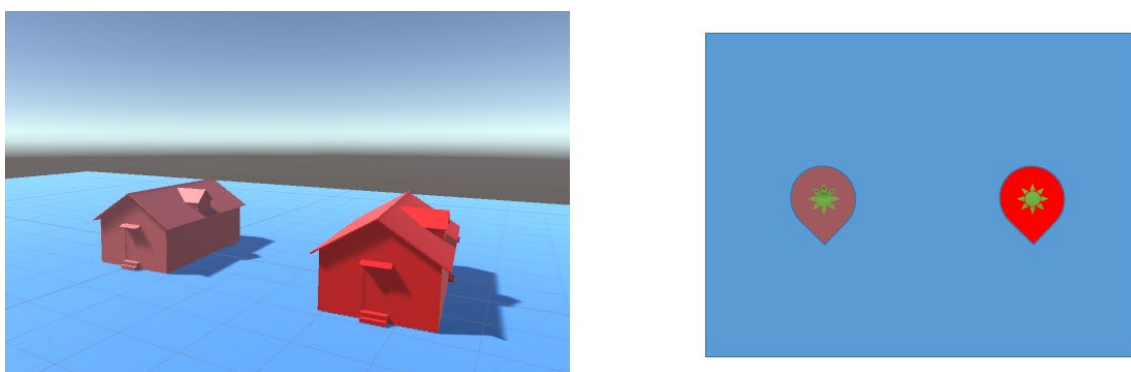


Figura 40 Ejemplo de cambio de marcador en entornos 3D (izquierda) y en entornos 2D (derecha) para indicar que es un objeto con incerteza.

Una opción igualmente válida es representar el objeto teniendo en cuenta todas sus posibilidades, e indicando de forma más sutil que este objeto tiene incerteza, dando opción a visualizar información sobre las otras alternativas, incluso visualizarlas.

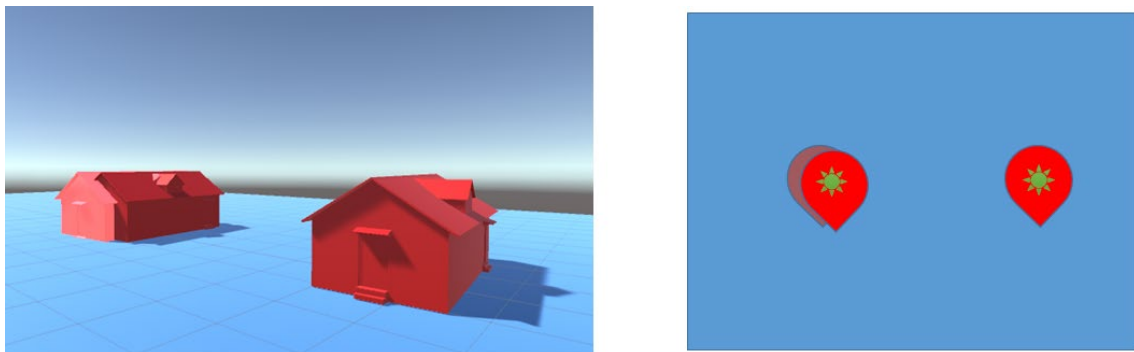


Esta alternativa tiene la ventaja de que se está representando la información que se conoce sobre los datos. De esta manera cuando se realicen filtrados de datos, así como relaciones, el objeto aparecerá en todas las posibles localizaciones, pero puede ser que esté duplicado.

Por ejemplo, en un conjunto de textiles antiguos, existe incerteza espacial en un objeto textil, cuya producción está fijada en Francia, pero también es probable que sea China, y para su elaboración se ha utilizado la técnica de “Espolinado”. Al filtrar objetos realizados con dicha técnica aparecerá dicho objeto ubicado en China, y también en Francia. De esta forma, el usuario vería que hay objetos en las dos regiones, aunque se trate del mismo. Si sólo se visualizará el objeto en Francia, seguramente no asociaría China como una región donde se han fabricado objetos así, cuando hay una probabilidad de que sí que existan.

Esta alternativa, al igual que la anterior tiene efectos negativos y positivos, pero de cualquier forma es necesario que se le informe al usuario mediante algún marcador que es un objeto con incerteza. Y en este caso, puede que este objeto esté representado en la escena con otro, u otros marcadores.

Siguiendo con la filosofía de la propuesta anterior, se propone una representación muy similar a la del objeto, para aprovechar las ventajas. En este caso sería replicar la técnica del marcador con un color suavizado, pero representarlo ligeramente desplazado del original. De esta forma se induce la idea de dualidad, ya que ese objeto tiene una, o más réplicas. En la Figura 41 se muestra un ejemplo donde se reproduce esta técnica. En la izquierda, en una escena tridimensional, a la izquierda de la figura, se observan dos objetos, uno con incerteza, donde se ve otro objeto 3D igual, pero ligeramente desplazado y con los materiales modificados. En la parte derecha de la figura se representan dos marcadores en una escena bidimensional, uno tiene incerteza, y otro no. El que tiene incerteza tiene detrás del objeto, otro objeto similar, desplazado, y con los valores de brillo y contraste modificados.



*Figura 41 Ejemplo de cambio de marcador en entornos 3D (izquierda) y en entornos 2D (derecha) para indicar que es un objeto con incerteza replicado.*

Un problema diferente, pero de consecuencias similares, es el de la granularidad, tanto espacial, como temporal. Si en el sistema de visualización, el tiempo y el espacio tienen una escala que permite utilizar escalas más pequeñas de tiempo que las del objeto, representarlo es un problema. Por ejemplo, a nivel espacial, si un objeto sólo se conoce que es de Francia, no hay problema para representarlo, si el sistema maneja países, o continentes, pero si es un mapa donde los objetos están geolocalizados, con sus coordenadas de latitud y longitud, con un error inferior a 1 Km, ¿en qué parte de Francia se representa este objeto? Aunque se elija un lugar, la capital, o el centro de la región, hay que indicar al usuario que ese objeto no está correctamente geolocalizado, como el resto de objetos de la escena. Además, es recomendable elegir varios lugares representativos de la región, para evitar solapamiento de objetos. Con la dimensión

temporal el problema es el mismo, si el sistema maneja una escala de años, o de cuarto de siglo, ¿dónde se representa, o a qué intervalo se asocia un objeto que se ha podido producir en un intervalo de dos siglos?

Este tipo de incerteza, derivada de la granularidad, hay que reflejarla como la incerteza inicial. El usuario, cuando busque más información sobre el objeto, para averiguar el motivo de la incerteza ya verá el problema analizando los valores de las propiedades.

#### 6.4. ASOCIACIÓN DE LA VISUALIZACIÓN CON DATOS DE DOMINIO

Una parte fundamental de este trabajo consiste en definir cómo asociar los objetos a representar con la información del grafo de conocimiento a representar.

Hasta ahora, se ha expuesto que el objeto a representar es una instancia de un concepto perteneciente a un dominio, y a continuación se ha hablado de las diferentes posibilidades de visualización, e interacción, de dichos objetos a representar, así como de sus propiedades espaciales y temporales. Sin embargo, el proceso de asociación no es trivial y requiere un análisis importante.

Esto es así, porque los grafos de conocimiento del dominio de datos pueden tener una complejidad importante, ya que se suelen diseñar para representar conocimiento con un alto nivel de abstracción. La visualización suele y debe ser mucho más simple, ya que está diseñada para ser comprendida y utilizada por las personas con la mayor sencillez posible. Por ello, generalmente, este proceso de asociación conlleva una simplificación habitualmente.

Por ejemplo, si para visualizar el material de un objeto de patrimonio cultural y acceder a su material, con el fin de mostrarlo en una ventana de información, o asociar una textura a su marcador, a nivel conceptual el usuario ha de entender que el material es como una propiedad del objeto. Es decir, el usuario final visualizará el Elemento X de clase O, que tiene como propiedad el material M, independientemente de que en el dominio de datos existan otras relaciones y conceptos. En la Figura 42 se muestra un diagrama con las instancias y propiedades del anterior ejemplo, tanto en el sistema de visualización (a la izquierda), como en el grafo de conocimiento basado en CIDOC-CRM (derecha). Si en el dominio de datos se usa el modelo CIDOC-CRM, un objeto de patrimonio cultural realizado por el hombre (cuadro, escultura, etc.) está representado por el concepto E22\_Man-Made\_Object. Para conocer el material que se usó para elaborar una instancia de dicho concepto, hay que buscar una relación de esta instancia con otra del concepto E12\_Production, a través de la propiedad P108\_has\_produced\_of. A su vez, la instancia del concepto E12\_Production, que esté relacionada con una instancia del concepto E57\_Material, a través de la propiedad P126\_was\_employed\_in.

Podemos hablar pues de una Instancia, o Elemento Visual (que en el ejemplo anterior sería el elemento x en el sistema de visualización), que pertenece a un Concepto Visual, y que tiene una o varias Propiedades Visuales (material M en el sistema de visualización del ejemplo anterior).



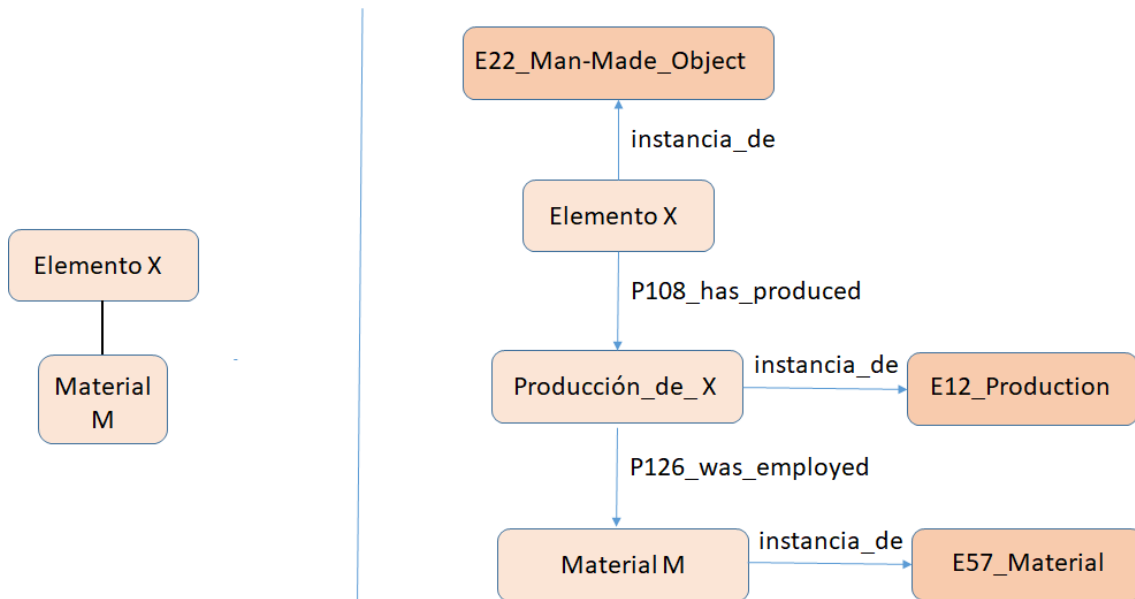


Figura 42 Diagrama con la conceptualización de un elemento X y una propiedad (material), en el sistema de visualización (izquierda) y en el grafo de conocimiento de dominio basado en CIDOC-CRM (derecha). Fuente: Elaboración propia.

La Instancia visual, se corresponde con una instancia de un concepto del dominio. De esta forma se infiere que un Concepto visual también se corresponde con uno, o varios Conceptos del dominio de datos. Obviamente, la Instancia, o Elemento Visual y la Instancia del dominio de datos, tienen el mismo identificador, que es su URI.

Esta asociación es más compleja con las propiedades. Una Propiedad Visual, puede ser una propiedad del Concepto de Dominio, pero también puede ser una propiedad de otro concepto del dominio, que está relacionado con el concepto inicial a través de relaciones con otros conceptos intermedios. Siguiendo con el mismo ejemplo, el “label” del Elemento Visual puede ser una propiedad del E22\_Man-Made\_Object, pero el material del que está hecho, está accesible a través de dos relaciones con el concepto E12\_Production como concepto intermedio.

Cuando sea una propiedad del propio concepto de la ontología de dominio, la asociación es sencilla, únicamente hay que referenciar el nombre de la propiedad en el grafo de conocimiento del dominio de datos.

Cuando el valor de una propiedad visual está en instancias del dominio, de otros conceptos diferentes al concepto de dominio relacionado con el concepto visual, la forma más directa de asociarlo es mediante una consulta. Es decir, el valor de la propiedad visual “material” se corresponde con el resultado, o resultados, de una consulta en la que está envuelta la URI del Elemento Visual. De esta forma, para obtener dicho valor, sólo habrá que ejecutar esta consulta. Resumiendo, el valor, o valores, de una Propiedad Visual se corresponderá con el resultado de una consulta sobre el dominio de datos, en el que la URI del Elemento Visual será un factor determinante.

Reutilizando el mismo ejemplo, si usamos el lenguaje de consulta SPARQL, la consulta para obtener los materiales de una instancia de E22\_Man-Made\_Object cuya URI es <URI\_Elemento> sería:

```
SELECT distinct ?materialLabel
WHERE {
    ?s
    rdf:type
    <http://erlangen-crm.org/current/E12_Production> .
    ?s
    <http://erlangen-crm.org/current/P108_has_produced>
    URI_Elemento .
    ?s
    <http://erlangen-crm.org/current/P126_employed>
    ?q .
    ?q
    <http://www.w3.org/2004/02/skos/core#prefLabel>
    ?materialLabel
}
```

Es decir, obtener la etiqueta de todas las instancias relacionadas, a través de la propiedad **P126\_employed**, con instancias de **E21\_Production**, que estén relacionadas a través de la propiedad **P108\_has\_produced**, con la instancia cuya URI sea la del Elemento Visual en cuestión (URI\_Elemento).

Por lo tanto, la asociación entre una Propiedad Visual y su Propiedad de Dominio (haciendo referencia a su valor en el dominio de datos) se realiza con el nombre de la propiedad, en el caso de que sea una propiedad del Concepto de Dominio, o mediante una consulta en la que interviene el identificador de la instancia del Concepto Visual, que es la misma que la instancia del Concepto de Dominio.

Además, dado que el Elemento Visual va representar un objeto que está referenciado espacio temporalmente, y estas propiedades son fundamentales en el sistema de visualización, habrá que distinguir estas propiedades visuales, del resto de propiedades.

Por lo tanto, habrá un tipo de propiedad visual que usará para representar la información espacial, y que todo concepto visual tendrá que tener una relación de cardinalidad, una, dos, o tres, con propiedades de este tipo (Propiedad Visual Espacial), en función del entorno espacial del sistema de visualización. También existirá un tipo de propiedad visual que se utilizará para representar la información temporal. Existirá una relación de cardinalidad dos, entre un concepto visual y una propiedad de este tipo (Propiedad Visual Temporal), dependiendo del entorno temporal del sistema de visualización.

## 6.5. CONCLUSIONES.

El objetivo de este capítulo era definir un modelo que represente a un sistema de visualización de datos de patrimonio cultural.

Para ello se han analizado las características de este tipo de sistemas, tanto en escenas tridimensionales, como bidimensionales.

Se han expuesto las características de una escena, que está compuesta por objetos visuales y procesos. Los procesos incluyen todas las acciones que pueden realizarse en la escena, ya sea por eventos generados por el sistema, o mediante la interacción del usuario.

Para cada una de estas características se han discutido diferentes técnicas gráficas para representarlas en la escena.

Además, se han tratado problemas propios de los sistemas de visualización espacio temporales, y lógicamente los de visualización de patrimonio cultural. Para cada problema se ha mostrado una, o varias posibilidades de representación gráfica, además de consideraciones que se han de tener en cuenta a la hora de desarrollar el sistema.

*A modo de resumen, se va a mostrar mediante una serie de tablas, las diferentes conclusiones obtenidas durante el capítulo. En primer lugar, en la Tabla 8 Tabla 9 se enumeran los conceptos principales del modelo. Después, en la Tabla 9 se muestran los principales atributos y relaciones del concepto Escena, en la Tabla 10 los del concepto Elemento Visual, en la*

Tabla 11 los del concepto Marcador, y para finalizar, en la Tabla 12 los del concepto Agrupación.

Estas conclusiones habrán de considerarse a la hora de diseñar e implementar una ontología, que represente a un sistema de visualización. Este proceso se analizará en el capítulo siguiente de esta investigación.

*Tabla 8 Principales conceptos del modelo de un Sistema de visualización de datos de patrimonio cultural*

| Concepto         | Descripción  |
|------------------|--|
| Concepto Visual  | <p>Un concepto visual se corresponde con un concepto de la ontología de dominio y sus instancias serán representadas en el sistema de visualización, de acorde al valor de sus atributos espacio temporales.</p> <p>El concepto visual tendrá una relación con un Entorno Espacial. Este entorno referenciará al sistema de coordenadas de los valores espaciales de los conceptos de la ontología de dominio representados por el elemento.</p> <p>El concepto visual tendrá una relación con un Entorno temporal. Este entorno permitirá representar los valores temporales de los conceptos de la ontología de dominio representados por el elemento.</p> <p>El concepto visual tendrá una relación con Propiedad Visual.</p> <p>El concepto visual tendrá una relación con Propiedad Visual Temporal.</p> <p>El concepto visual tendrá una relación con Propiedad Visual Espacial.</p> |
| Propiedad Visual | <p>Una propiedad visual hace referencia a una propiedad de un concepto del dominio de datos.</p> <p>La propiedad visual está relacionada con un Concepto Visual.</p>   |

|                  |  |
|------------------|--|
|                  | <p>Si la propiedad del dominio de datos está relacionada con el Concepto de dominio referenciado por el Concepto Visual, sólo será necesario conocer el nombre, o la referencia a dicha propiedad.</p> <p>Si esa propiedad está relacionada con otro concepto del dominio de datos, se vinculará mediante una consulta en la que intervenga la URI de la instancia del Concepto visual.</p>  |
| Escena           | <p>La Escena hace referencia a un entorno completo dentro del sistema de visualización en el cual se representan elementos visuales.</p> <p>La escena tendrá una relación con un Entorno Espacial. Este entorno referencia al sistema de coordenadas de los valores espaciales que serán visualizados.</p> <p>La escena tendrá una relación con un Entorno temporal. Este entorno permitirá representar los valores temporales que se visualizarán en la escena y con los que se podrá interactuar, si así se establece.</p> <p>La escena, además estará relacionada con un tipo de representación gráfica, que tendrá unas u otras posibilidades de representación e interacción. Por ejemplo, puede ser un mapa interactivo, o un grafo, o simplemente una representación 2D, o 3D de los puntos en un espacio virtual, etc.</p> |
| Entorno Espacial | <p>El Entorno espacial es un concepto utilizado para representar el espacio en el que toman valores las propiedades espaciales.</p> <p>Puede ser tridimensional, o bidimensional. La dimensión estará definida por dos puntos tridimensionales.</p> <p>En el entorno tridimensional, dichos puntos definirán un ortoedro.</p> <p>En el entorno bidimensional, la tercera dimensión tomará el valor cero, y los dos puntos definirán un rectángulo.</p> <p>El entorno usará un sistema de coordenadas.</p>  |
| Entorno Temporal | <p>Tiempo hace referencia a una organización temporal.</p> <p>Esta organización estará definida por un momento inicial, otro final, y una frecuencia, o unidad de tiempo. De esta forma los valores temporales representados por el entorno estarán contenidos en el intervalo [inicial, final] y quedarán definidos por dicha unidad temporal.</p>  |
| Marcador         | <p>Representa el elemento gráfico con el que se representa visualmente un Elemento Visual en la Escena.</p>  |
| Agrupación       | <p>Representa a una técnica de agrupamiento utilizada para agrupar los Elementos Visuales de la Escena.</p>  |

*Tabla 9 Propiedades del concepto Escena.*

| <b>Nombre</b>            | <b>Descripción</b>  |
|--------------------------|---|
| Entorno Espacial         | Entorno Espacial de la Escena.  |
| Entorno Temporal         | Entorno Temporal usado en la Escena.  |
| Elementos                | Una Escena estará relacionado con varios Elementos visuales. De esta forma se podrán indicar los diferentes conceptos de la ontología de dominio que pueden representarse en la escena.   |
| Niveles de zoom.         | Define el número máximo de niveles de detalle que tiene el entorno de la escena.<br><br>El detalle del entorno hace referencia a la visualización gráfica de los elementos de la escena que no son Elementos Visuales.  |
| Niveles de detalle       | Niveles de detalle de la escena. Estos niveles serán utilizados para actualizar los Elementos Visuales visualizados en la Escena. Generalmente existirán menos niveles de detalles que niveles de zoom, por lo que un nivel de detalle de la escena estará asociado a un conjunto de niveles de zoom. De este modo, aunque mejore el aspecto gráfico de la escena, los Elementos Visuales no tienen por qué actualizarse. |
| Línea de tiempo          | Este atributo determina si la Escena va a visualizar un control de Línea de tiempo, o no. Las características de este control vendrán determinadas por el Tiempo asociado a la Escena. Esto determinará, el inicio, el fin y los intervalos de tiempo, así como si tiene que ser jerárquica, o lineal.  |
| Tiempo simultáneo        | Este atributo establece si la Escena va a presentar un control que permita visualizar la distribución de los datos de forma simultánea, o no. En el caso de que se visualice ha de definir si será continuo, o por capas.   |
| Marcador                 | Marcador por defecto de los Elementos Visuales de la Escena. Este marcador será el usado para representar todas las Instancias de Elementos Visual relacionadas con la Escena, en el caso de que estos no tengan ninguna instancia de Marcador relacionada.   |
| Marcador Representante   | Marcador representante por defecto, de la Agrupación de los Elementos Visuales de la Escena.  |
| Agrupación               | Agrupación por defecto de los Elementos Visuales relacionados con la Escena.  |
| Número máximo de objetos | Número máximo de objetos relacionados que se pueden mostrar en la escena. En el caso de que haya más objetos relacionados, el sistema debe de indicar este problema, sugiriendo filtrar más los datos.  |

|                                   |   |
|-----------------------------------|---|
| Técnica de dibujado de Relaciones | Técnica a usar para representar las relaciones, puede ser mediante “Líneas, o Arcos”, o mediante un Marcador que se dibujará superpuesto al del objeto.   |
| Marcador de Relaciones            | Marcador usado para indicar que un objeto está relacionado. Al estar superpuesto al del objeto, será más grande que el del objeto, pero puede ocultarlo, por lo que se recomienda que tenga transparencia en la zona central. |
| Porcentaje en relaciones          | Indica que la Escena va a poder mostrar el control de Anillo para indicar el porcentaje de objetos relacionados con cada instancia de Elemento Visual.  |
| Tratamiento Incerteza             | Define cómo se va a representar la incerteza en la Escena, replicando los objetos inciertos, o no.  |
| Marcador Incerteza                | Marcador por defecto de la incerteza. En el caso de que no haya ningún marcador, ni en este atributo, ni en el Elemento Visual, se representará con la técnica de cambio de color.  |
| Marcador Replicación              | Marcador por defecto de la incerteza por replicación. En el caso de que no haya ningún marcador, ni en este atributo, ni en el Elemento Visual, se representará con la técnica de desplazamiento y cambio de color.           |

Tabla 10 Atributos y Relaciones del concepto Elemento Visual

| Nombre                 | Descripción  |
|------------------------|--|
| Concepto de Dominio    | Concepto del dominio de datos vinculado con el Elemento Visual.  |
| Escena                 | Escena en la que se representarán las instancias del concepto de dominio (Concepto) siguiendo las directrices definidas por las propiedades y relaciones del Elemento Visual.      |
| Entorno Espacial       | Entorno usado por el concepto del dominio de datos para referenciar espacialmente los datos.   |
| Entorno Temporal       | Entorno Temporal usado en el Elemento Visual.  |
| Marcador               | Marcador del Elemento Visual con el que se visualizarán las instancias del concepto del dominio relacionado con este Elemento Visual en las Escenas en las que estén relacionados. |
| Marcador Representante | Marcador representante de la Agrupación de las instancias relacionadas con este Elemento Visual que se representarán en la Escena  |
| Agrupación             | Agrupación de las instancias de este Elemento Visual que se representarán en la Escena del Elemento Visual.  |

|                           |   |
|---------------------------|---|
| Propiedades relacionables | Conjunto de propiedades visuales sobre las que se va a poder buscar objetos relacionados que tengan el mismo valor en dichas propiedades. |
| Propiedades filtrables    | Conjunto de propiedades visuales sobre las cuales se va a poder realizar filtros en el sistema de visualización.                          |
| Marcador incerteza        | Marcador usado para representar instancias del objeto de dominio con incerteza.   |
| Marcador replicación      | Marcador usado para representar instancias del objeto de dominio cuando la incerteza se representa con replicación.                       |

*Tabla 11 Atributos y Relaciones del concepto Marcador*

| Nombre        | Descripción   |
|---------------|---|
| Figura        | En el caso de que el marcador sea una figura simple (cuadrado, círculo, triángulo, etc.) este atributo tendrá el valor correspondiente. Si la escena es tridimensional, el marcador será un modelo 3D equivalente (un cuadrado, será un cubo, un círculo será una esfera, etc.) |
| Color Relleno | En el caso de que el marcador sea una figura simple, mediante este color se rellenará la figura.  |
| Color Línea   | En el caso de que el marcador sea una figura simple, mediante este color se dibuja la línea que define la figura.   |
| Imagen        | URL de una imagen usada en la representación del Marcador. Si es una figura, o su versión en modelo 3D, se rellenará con esta imagen.   |
| Modelo 3D     | URL de un modelo 3D que se usará para representar las instancias relacionadas con los Elementos Visuales que tengan una relación con este Marcador.   |

*Tabla 12 Atributos y Relaciones del concepto Agrupación*

| Nombre               | Descripción   |
|----------------------|---|
| Tipo                 | Tipo de agrupación, (marcadores representantes, o mapa de calor).   |
| Solapamiento Espiral | Número máximo de datos que se representarán en Espiral al desplegarse el grupo, en el caso de que estén en la misma ubicación espacial. En el caso de que se supere ese número se usará una técnica alternativa.  |
| Refuerzo Visual      | Puede tomar los valores “Ninguno”, o “Colores” y/o “Tamaño” y/o “Número”. Indica que el Marcador del agrupamiento puede visualizarse con un color y/o un tamaño diferente, en función del número de datos, además de indicar el número de elementos de la Agrupación. |



|                                  |  |
|----------------------------------|--|
| Rango Color Mapa                 | Dos colores que definen un rango entre el que fluctúan los colores del Mapa de calor, o en el caso de que sea de tipo Marcadores Representantes, y haya refuerzo visual por colores. Los colores intermedios quedan definidos por la escala de calor de los colores. |
| Transiciones Mapa                | Número que define el total de transiciones que pueden darse en las agrupaciones. Esto puede aplicarse en el Mapa de Calor y también en el caso de que haya Refuerzo Visual por colores y tamaños.  |
| Técnica Alternativa Solapamiento | Técnica alternativa de solapamiento en el caso de que el número de datos ubicados en la misma posición sea mayor que el atributo Solapamiento Espiral. Puede ser la técnica de “Lista de Datos”, o “Visualización Dinámica Alternativa”                              |

## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

### 7.1. RESUMEN

En este capítulo se parte de un modelo de un sistema de visualización definido en el capítulo 6. El objetivo de este capítulo es diseñar e implementar una ontología que represente dicho modelo. Además, con el fin de reutilizar los avances expuestos en el campo de Ontologías de Visualización, descrito en el capítulo 4, se discutirán las ventajas e inconvenientes de esos resultados, para mejorarlos y combinarlos con el modelo del capítulo 6. De este modo la ontología no sólo definirá el sistema de visualización, y su conexión con uno, o varios grafos de conocimiento con la información de patrimonio cultural, además, indicará cómo visualizar dicha información mediante una ontología de visualización. Esta ontología recibirá el nombre de STEVO (Spatio TEmporal Visualization Ontology)

Al definir el modelo se han enumerado varios conceptos y propiedades, que serán utilizados para definir la ontología a diseñar. Estos conceptos se han obtenido desde un punto de vista general, a modo de resumen del contenido del modelo. Esta información deberá considerarse en el proceso de diseño de la ontología, pero únicamente como punto de partida, al igual que el resto de información sobre el modelo expuesto en el capítulo anterior.

Para realizar el diseño y construcción de la ontología se estudiarán los resultados de algunas de las metodologías de diseño existentes. Aunque no es el camino habitual, debido a los objetivos de este trabajo y a las características de los dominios de datos, una serie de ontologías externas serán estudiadas, con el fin de analizar su posible integración con la ontología a diseñar.

Primero se analizará la posibilidad de incluir alguna de las ontologías de visualización analizadas. Para ello, se expondrán las ventajas e inconvenientes de las diferentes ontologías de visualización. A continuación, se describirán las mejoras a realizar en dichas ontologías para contemplar las necesidades del modelo, así como las implementaciones más utilizadas actualmente de este tipo de sistemas.

En el modelo es fundamental la representación de datos espacio temporales. La comunidad científica ha desarrollado muchos más trabajos en el diseño de ontologías para definir el espacio y el tiempo. Debido a ello se expondrán dichos resultados en el campo espacial y en el temporal, para indicar a continuación, cuál será seleccionado para reusar y/o extender con el fin de ser incluido en la representación del modelo.

Una vez consideradas las posibilidades de integración de las ontologías estudiadas, se procede al diseño de la ontología de esta investigación, siguiendo las actividades de la metodología METHONTOLOGY (Methodology for Ontology construction) (Corcho, Fernández-López et al. 2005). De esta forma, ya que la metodología de diseño lo permite, se podrán integrar los conceptos de estas ontologías durante el proceso de diseño.

### 7.2. DISEÑO.

Actualmente hay varias metodologías diseñadas con el fin de establecer una serie de pasos para poder construir una ontología sobre un dominio (Bravo, Hoyos Reyes et al. 2019, Corcho,

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Fernández-López et al. 2005, Noy, McGuinness 2001, Fernandez-Lopez, M., Gomez-Perez, A., Juristo, N. March, 1997).

Todas estas metodologías definen una serie de pasos comunes, con mayor o menor detalle y refinamiento, como identificar el propósito y el alcance de la ontología. Proponen el uso de términos y conceptos, que se refinan iterativamente, para una posterior instanciación y validación del resultado obtenido, aunque difieren en el enfoque y algunos pasos (Sure, Studer 2002).

En esta investigación, se parte del conjunto de posibles dominios de datos con información espacio temporal sobre patrimonio cultural, y las diferentes formas de visualizar dicha información es lo que se desea representar mediante una ontología.

Por lo tanto, el dominio de datos de la ontología a diseñar estaría formado por las diferentes formas posibles de visualizar información de datos del patrimonio temporal. Es obvio que hay una enorme variedad de formas de visualizar la información, pero con el contenido analizado en el capítulo anterior se pretende definir cómo son las diferentes formas analizadas en esta investigación, además de contemplar la posibilidad de incluir nuevas formas aún no desarrolladas.

Para construir esta ontología se seguirá la metodología de METHONTOLOGY (Corcho, Fernández-López et al. 2005). METHONTOLOGY está basada en una serie de actividades identificadas por el proceso de desarrollo de software propuesto por (Elkan, Greiner 1993). Estas actividades son:

- Especificación. Determinar el motivo de construir la ontología, su uso y potenciales usuarios de la misma.
- Conceptualización. Organizar una descripción informal del dominio en una especificación semi-formal, en notación tabular y gráfica. Como resultado, se obtiene el modelo conceptual de la ontología.
- Formalización. Transforma el modelo conceptual en un modelo formal.
- Implementación. Construir el modelo formal mediante un lenguaje de ontologías.
- Mantenimiento. Actualización de la ontología.

Durante todo el proceso, se realizan de forma transversal actividades de gestión (planificación, control y calidad), y de soporte (adquisición de conocimiento, integración, evaluación, documentación y gestión de la configuración).

La actividad más compleja es la conceptualización, ya que partiendo de una descripción informal se obtiene el diseño conceptual, desde el cual, obtener la ontología es relativamente sencillo.

Durante esta tarea, mediante la tarea de integración, se puede integrar ontologías externas y evitar retrasar esta tarea para la actividad de implementación.

Esta investigación parte del objetivo de reutilizar una ontología de visualización para representar los objetos gráficos que se van a emplear para la representación de los datos. Además, los datos tienen conceptos espaciales y temporales. Por lo que es probable que existan ontologías que representen estos conceptos y que serán reutilizadas.

Por esta razón, antes de comenzar con las diferentes fases del diseño, según METHONTOLOGY, se procederá, partiendo de la descripción del modelo realizado en el capítulo anterior, a analizar ontologías representativas de estos dominios, y considerar las posibilidades de reutilización de

su contenido. Posteriormente, una vez analizadas, si las consideraciones son oportunas, se incluirán de forma adecuada durante el proceso de diseño.

#### *7.2.1. Ontologías de visualización*

Tal y como se expone en el capítulo 4 de esta tesis, la idea de relacionar el contenido gráfico con una ontología no es nueva, y han surgido varios trabajos en esta línea. A continuación, se analiza la aplicación de cada uno de estos trabajos para aplicar su uso, o parte de sus avances, para conseguir los fines de esta investigación.

##### *Definición de escenas 3D mediante ontologías*

Los primeros estudios y desarrollos (Bolívar, H., González, R., Sanjuán, O. 2009), estaban orientados a representar una escena 3D mediante una ontología. El problema de estos trabajos es que estaban muy ligados a APIs como Java 3D, y este tipo de ontologías eran prácticamente formatos de representación de la escena. Es decir, no estaban diseñadas para representar información de un grafo de conocimiento, lo estaban para que una aplicación procese la ontología con la información de la escena, siguiendo la estructura y filosofía de la tecnología de visualización y la represente.

Para visualizar el contenido de un grafo de conocimiento, habría que procesar los datos de origen, para crear instancias en una ontología de definición de escena. Posteriormente, otra aplicación, o módulo, se encargaba de procesar la ontología de definición y generar la escena. De este modo, la ontología de definición era generalmente como una especificación de la escena que se generaba desde la aplicación ad hoc, a la que además estaba totalmente ligada. Es decir, que una aplicación creaba instancias de la ontología, siguiendo los criterios de sus diseñadores, o desarrolladores, y esta ontología era procesada para generar una escena con pocas posibilidades de variación, ya que el contenido estaba muy ligado a la API. El proceso consistía prácticamente en hacer llamadas al API en función de la clase de las instancias y su contenido.

Hay dos motivos por los que esta idea está bastante alejada de los objetivos de esta investigación:

- No resuelve cómo ha de visualizarse el contenido de un grafo de conocimiento. Para visualizarlo habría que procesar el grafo, y según los criterios particulares, adaptar la información al contenido de una ontología, a modo de archivo, o estructura intermedia. En todo caso podría reutilizar la parte de las primitivas gráficas y objetos 3D, para representar marcadores, incluso niveles de detalle de la escena. Tarea que por otra parte requeriría unos conocimientos importantes, tanto de la API, como de representación gráfica en tres dimensiones.
- La estructura de esta ontología está muy ligada al API Java 3D y a las posibilidades de estándares como X3D. Si no se eligen estas tecnologías para su representación habría que hacer un esfuerzo importante para replicar, o adaptar estas estructuras a las actuales. Además, solamente serviría para representar estructuras gráficas, y unas propiedades concretas, otras más avanzadas, como es el caso de filtrado, relaciones entre los objetos, representar objetos en diferentes intervalos de tiempo, etc., requeriría ampliar el esquema de la ontología, o generar un nuevo esquema, conectado al anterior de alguna forma.

Las dos razones son importantes, pero la primera es la más difícil de superar, ya que choca directamente con el objetivo de esta investigación, que es visualizar el contenido de un grafo de conocimiento.

## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Otros trabajos más avanzados como es el caso de las ontologías de visualización VISO y VUMO, no tienen estos problemas. Los dos diseños permiten relacionar el contenido del grafo de conocimiento con conceptos de la ontología, y cómo visualizarlos. Sin embargo, presentan otras dificultades que se exponen a continuación.

### Ontología VISO

La ontología VISO (Polowinski, Jan, Voigt 2013) realmente es un conjunto de ontologías diseñadas para definir cómo visualizar el contenido de un grafo de conocimiento. Estas ontologías, especialmente la ontología GRAPHIC permite definir estructuras gráficas de bajo y de alto nivel, que permiten definir una escena. Sin embargo, aunque en niveles altos de abstracción, la mayoría de los conceptos son válidos, al estudiar niveles intermedios, o más bajos de abstracción, se observa que estos están definidos para representación de escenas bidimensionales. Esto implicaría la necesidad de extender la ontología para contemplar la visualización de escenas tridimensionales. Esto en sí, no es un problema importante, pero es una tarea que hay que tener en cuenta.

VISO, además de la definición de los conceptos gráficos, datos a manejar, y demás conceptos de factores que influyen en el proceso de visualización, parte de la definición de visualización se hace a través del lenguaje RDFS/OWL Visualization Language (RVL) (Polowinski, Jan Jun 12, 2013). Este lenguaje es el encargado de definir muchos aspectos de la visualización, y consiste esta basado en esquemas OWL con conexiones a SPIN, que se incorpora en conceptos y propiedades de la ontología de dominio. Para ello, hay que extender la ontología de dominio, e incluir en ellos RVL, desde los cuales se hace referencia a los conceptos de VISO.

En el diseño de este trabajo, el requerimiento y el papel del lenguaje RVL es muy importante. Para representar las diferentes formas de representar la incerteza, habría que hacerlo mediante su uso. Por ejemplo, con RVL se pueden consultar valores de variables, y en el caso de que no tengan valor, o tengan varios posibles valores, hecho que sucede con la incerteza, decidir cómo representar su contenido mediante RVL referenciando diferentes conceptos de VISO.

Posteriormente, mediante el uso de la tecnología D3.js, se ha desarrollado una versión inicial de una aplicación visualizador, que analiza la extensión del grafo a representar con los scripts de RVL y accediendo al resultado de consultas sobre el grafo de conocimiento, o accediendo a su contenido por completo, lo representan. En la Figura 18 se pueden observar algunos ejemplos de visualización generados por este visualizador. El visualizador usa esta tecnología, pero realmente se podría usar otra, desarrollando de nuevo la parte gráfica.

A priori, esta filosofía parece encajar en los objetivos de esta investigación, pero hay una serie de problemas importantes:

- El lenguaje RVL es complejo, y requiere un coste importante de aprendizaje. Además, su legibilidad es muy reducida. Esto implica la necesidad de expertos en este lenguaje para definir cualquier sistema de visualización.
- La filosofía de este trabajo está orientada a tomar como base el grafo a visualizar, la ontología VISO y el lenguaje RVL, para que, con toda esa información, se genere automáticamente la escena mediante el visualizador. Este marco de trabajo puede ser aceptable para visualizaciones de complejidad media, o básica. Sin embargo, si el objetivo es implementar un sistema de visualización con interfaces avanzados, o escenas de una alta complejidad, los scripts de RVL serían muy grandes, de costoso desarrollo y de muy difícil

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

mantenimiento. Además, RVL está pensado para definir como visualizar un concepto, permitiendo una definición base y modificarla, o extenderla en función de los valores de las instancias. Sin embargo, no hay ninguna posibilidad para conectar con otras aplicaciones a través de un API, o servicio web, y no presenta opciones para ofrecer interfaces de usuario. Lo que hace que sea prácticamente inabordable utilizar este sistema para escenas de alta complejidad.

Además, al evaluar el estado actual de los desarrollos en el visualizador de RVL se observa que aún requiere mucho trabajo para poder representar muchos elementos de una escena de complejidad básica.

De hecho, tras investigar y analizar la ontología VISO se concluye que es una buena línea de investigación para el desarrollo de visualizaciones de complejidad básica, o media, si se amplía el desarrollo del visualizador. Sin embargo, dejar el peso de la visualización de una escena compleja, con interfaces de usuario avanzados y en el que se van a visualizar muchos datos es realmente complejo. A pesar de esto, se podría mejorar mucho las posibilidades de visualización de datos con VISO, sí tanto en el lenguaje RVL, como los desarrollos del visualizador permiten la especificación y desarrollo de técnicas de visualización, como agrupamiento, relaciones, etc.

#### VUMO

VUMO (Sobral, Thiago, Galvão et al. 2020) es una ontología diseñada para visualizar datos de movilidad urbana. Debido a ello tiene una ontología compuesta de elementos utilizados en estos sistemas, que pueden ser mapeados con una ontología de dominio. Con la ontología de dominio mapeada, ofrece un sistema de recomendación, que popula una ontología de visualización, en la que hay dos tipos de escenas definidas, Geographic Heat Map y Calendar Heat Map, en la Figura 19 se muestran dos esquemas que representan como sería la visualización de estas escenas.

VUMO, incluye una serie de conceptos gráficos, no tan complejos como la ontología VISO, pero que permiten definir la mayoría de las primitivas gráficas, y así personalizar algunos aspectos de la visualización de las escenas a representar.

El trabajo realizado con VUMO está muy bien diseñado, pero tiene varios problemas para utilizarla plenamente para las necesidades de esta tesis:

- Es un sistema diseñado para visualizar elementos de movilidad urbana. A priori no tiene mucho en común con el patrimonio cultural, y de hecho hay características, como la incerteza, o la granularidad que no están contempladas. Pero en el fondo, siguen siendo elementos que están ubicados en el espacio, cuya posición y características puede variar en el tiempo. Esto quiere decir que probablemente podría usarse para representar algunos sistemas de visualización de patrimonio cultural, pero habría que realizar extensiones para contemplar la granularidad, la incerteza y también sobre el tiempo. A nivel temporal, la principal diferencia es que VUMO está muy orientada al tiempo real, los minutos y los segundos son muy importantes en temas de tráfico urbano, pero estos datos jamás se consideran en patrimonio cultural. Además, de que habría una cantidad de conceptos que no se usarían jamás.
- Las capacidades de personalización gráficas están muy limitadas y no contempla el uso de elementos tridimensionales.
- No se ha desarrollado ningún visualizador de su ontología final.

## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

- El sistema de recomendación está diseñado para ontologías de movilidad urbana, por lo que habría que adaptarlo a las necesidades de patrimonio cultural, y los nuevos elementos gráficos y escenas.
- Su ontología de visualización, tal y como se describe en el estado del arte de esta investigación, contempla únicamente dos escenas, las cuales habría que extender y también crear nuevas escenas.

Resumiendo, usar los resultados de VUMO para los fines de esta investigación, implicaría realizar un trabajo importante de extensión de todas sus ontologías básicas, además de definir y modificar las escenas definidas, adaptándolas a las nuevas posibilidades gráficas. Además, habría que extender y modificar notablemente el sistema de recomendación, y desarrollar un visualizador.

### Conclusiones

Tras analizar la posibilidad de reutilizar total, o parcialmente, los trabajos realizados en ontologías de visualización, para representar el modelo propuesto, se obtienen las siguientes conclusiones.

Los resultados de los primeros trabajos sobre definición de escenas 3D con ontologías no son válidos para representar el modelo propuesto en esta investigación. Se limitan a definir una escena tridimensional sin mapearlo con la ontología de dominio, ni tener en cuenta muchas las necesidades del modelo. Además, el API de Java3D y la especificación de X3D hace muchos años que no recibe actualizaciones y cada vez es más difícil ejecutar sus contenidos en sistemas operativos iOS y Mac OS.

Los resultados de VISO no pueden utilizarse en su totalidad, porque en el modelo propuesto son necesarios la posibilidad de definir una serie de interfaces de usuario y acceso a APIs de aplicaciones externas, características no contempladas en el lenguaje RVL. Dado que se trata de un sistema abierto, podrían expandirse las posibilidades de VISO para que cubriera estas necesidades. Sin embargo, de hacerlo, los scripts del lenguaje serían muy grandes, con un mantenimiento muy difícil y con un nivel de legibilidad muy bajo. RVL está diseñado para definir la visualización con una serie de comandos que actúan sobre los conceptos y los valores de los atributos de sus instancias. Sin embargo, las necesidades del modelo propuesto requieren de la complejidad de un lenguaje de programación más desarrollado. Además, también sería necesario realizar desarrollos importantes sobre el visualizador.

Si se decidiera reutilizar VISO y RVL para los fines de esta investigación, haría falta un trabajo de análisis, diseño e implementación de grandes dimensiones. Además, en el caso de hacer este trabajo, se obtendría un sistema que requeriría altos conocimientos de un lenguaje específico, que permitiría desarrollar implementaciones de difícil mantenimiento.

Los resultados de VUMO tampoco son directamente reutilizables para representar el modelo propuesto en esta investigación. Al estar diseñada para representar datos de movilidad urbana, habría que realizar muchas adaptaciones y extensiones. Exactamente igual para la parte gráfica, muy limitada en VUMO. Y lógicamente el visualizador habría que desarrollarlo por completo, además de realizar un trabajo importante en el sistema de recomendación.

Por lo tanto, se concluye que ninguno de los trabajos analizados se puede reutilizar de forma total para los fines de esta investigación, aunque si parcialmente como se explica a continuación.



CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Por una parte, VISO, con el conjunto de sus ontologías, el lenguaje RVL y el visualizador, que aún necesita desarrollos importantes, tiene como objetivo comprender todo el proceso completo de visualización procesando el contenido de una ontología, o un grafo de conocimiento. Desde el punto de vista de esta investigación, se asume que es una iniciativa muy ambiciosa, y se ha desarrollado a varios niveles, tanto en diseño, como en desarrollos tecnológicos. Sin embargo, la realidad tecnológica actual, dista mucho de que este sistema puede utilizarse para representar datos con mucha información y con unos interfaces de usuario adecuados.

La iniciativa de VUMO también es muy buena, pero está muy focalizada en movilidad urbana y necesita muchos desarrollos, tanto en diseño, para representar otro tipo de escenarios y elementos gráficos, como en implementaciones en tecnología para visualizar sus resultados. Sin embargo, la idea de delegar de forma externa el desarrollo tecnológico, a través de tipos de escenas, es más realista que en VISO, aunque desde el punto de vista de esta investigación, debería de poder especificarse más la escena desde la ontología de visualización.

Por ello, desde esta investigación se va a utilizar la idea, presente en VISO y VUMO de mapear los conceptos del grafo de conocimiento a representar, con los de una ontología de visualización.

También se va a usar la idea de VUMO de delegar en un sistema externo, la tarea de visualizar la información de la ontología a través de tipos de escena.

Sin embargo, al contrario que en VUMO, que no define muchos de los conceptos gráficos que se usan en la escena, se pretenderá que el máximo posible de factores usados en la escena pueda definirse mediante la ontología del modelo. Para ello, dado que VISO, en su ontología GRAPHICS ha realizado un esfuerzo para representar diferentes posibilidades gráficas, se usará esta ontología para tal fin.

Para ello, se diseñará e implementará una ontología que representa al modelo propuesto en el capítulo anterior de esta investigación, pero haciendo referencia a los elementos de la ontología GRAPHICS de VISO y asociando la escena a un tipo de escena, como propone VUMO. Además, dado que los conceptos de la ontología GRAPHICS no contemplan muchos de los conceptos de escenas tridimensionales, se extenderá adecuadamente para que estén incluidos.

En la Figura 43 se muestra un esquema que representa los puntos básicos que definen el diseño de esta ontología. Está basada en un conjunto de tipos de escena, sobre los que se podrá definir los elementos y procesos del modelo del sistema de visualización. Pero estos conceptos, referenciarán en la medida de lo posible a conceptos de la ontología GRAPHICS de VISO, a la vez que podrán ser mapeados a conceptos del grafo de conocimiento cuyos datos se desea visualizar.

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

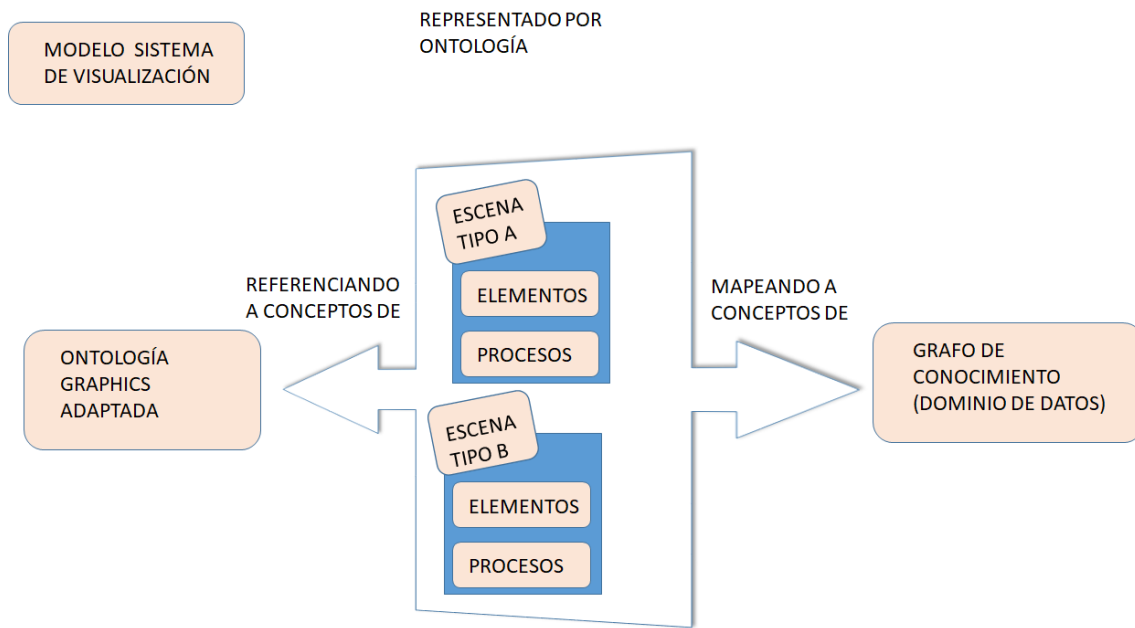


Figura 43 Esquema de elementos de diseño básicos de la ontología de visualización propuesta. Fuente: Elaboración propia.

A nivel gráfico, la ontología VISO es la que mejor define los conceptos gráficos que se usarán para representar el modelo, sin embargo, otros resultados han sido utilizados en el diseño de la ontología que se propone diseñar mediante esta investigación.

A continuación, se expondrán las extensiones realizadas sobre la ontología GRAPHICS de VISO, y también se definirán unos tipos de escena base, que posteriormente podrán ampliarse.

#### Extensiones sobre ontología GRAPHICS

El módulo GRAPHICS de la ontología VISO consiste en un esquema que formaliza los términos del dominio gráfico basándose en dos clases principales: **Graphic Object** y **GraphicRelation**.

Con **Graphic Object** se representan todas las entidades gráficas, desde objetos simples, que estarían representados por la clase **Elementary Graphic Object**, hasta representaciones más complejas, que se corresponden con una escena, representados por la clase **Graphic\_Representation**.

Con **GraphicRelation** se hace referencia a todas las relaciones que puede tener un objeto **Graphic Object**. Esta clase está extendida por dos clases. Una primera clase para relacionar los atributos del objeto, representada por la clase **GraphicAttribute**. Estos atributos pueden ser continuos (**GraphicAttributeContinuos**), como por ejemplo su posición, su color, etc., pero también discretos (**GraphicAttributeDiscrete**) como la forma, una etiqueta, etc. Una segunda clase, que permite relacionar el un objeto gráfico (**GraphicObject**) con otro, se trata de la clase **GraphicObjetToObjectRelation**. Con esta última clase se puede representar la pertenencia de objetos a una escena, o como se representa gráficamente la relación objetos, por ejemplo, uniéndose con líneas, agrupándolos, etc.

En la Figura 44 se muestra un esquema de estas clases, donde se puede apreciar su jerarquía, además de indicar algunos ejemplos de posibles instancias, gráfica y textualmente, a modo explicativo.

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

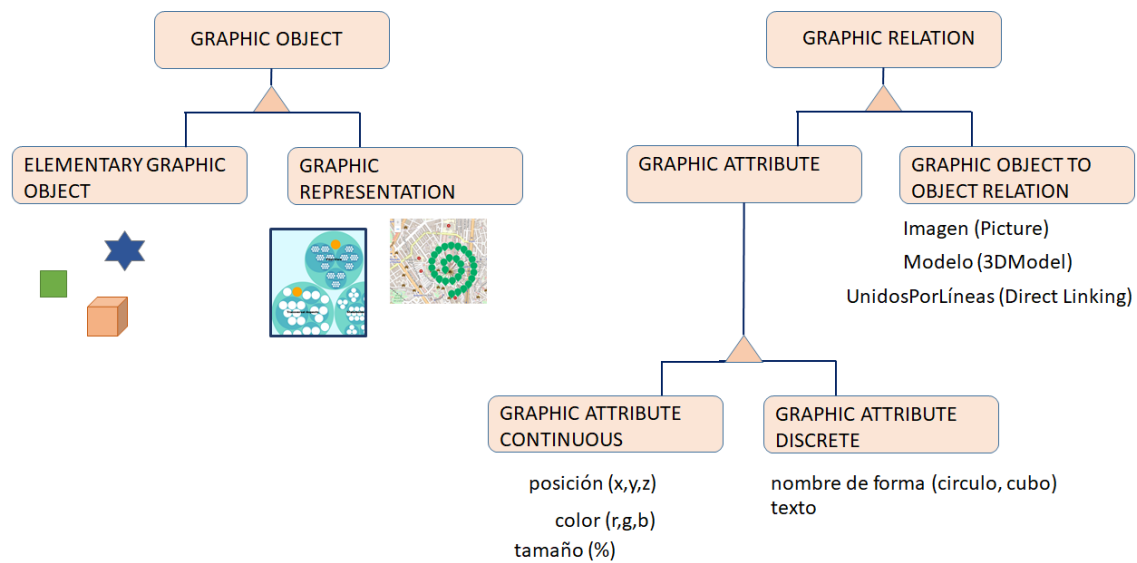


Figura 44 Esquema de las principales clases del módulo GRAPHICS de VISO. Fuente: Elaboración propia.

Para representar el modelo descrito en el capítulo anterior es necesario que exista una Escena, que como se ha comentado se corresponde con la clase **Graphic Representation**, del módulo GRAPHICS de VISO. En el apartado siguiente se analizarán los diferentes tipos de escena y las adaptaciones de VISO necesarias. Independientemente de los tipos de Escena, según el modelo descrito van a ser necesarios una serie de objetos gráficos, obtenidos tras analizar el contenido de la Tabla 8, Tabla 9, Tabla 10,

Tabla 11 y la Tabla 12.

En la Tabla 13, se enumeran los objetos gráficos requeridos por el objeto Marcador del modelo y como se referencia a ellos usando el módulo GRAPHICS.

En la Tabla 14, se describen los objetos gráficos solicitados por el objeto Escena del modelo y como referenciarlos mediante el módulo GRAPHICS

En la Tabla 15, se detallan los objetos gráficos requeridos por el objeto Agrupación del modelo y la forma de relacionarlos con el módulo GRAPHICS

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tabla 13 Descripción del objeto gráfico Marcador del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados.

| Objeto del modelo | Conceptos de ontología GRAPHICS de VISO utilizados y/o extendidos   |
|-------------------|---|
| Marcador          | <p>Esta clase está directamente relacionada con la clase <b>Primary_Graphic_Representation</b>, que extiende la clase <b>Graphic_Object</b> de VISO. Hay que tener en cuenta las siguientes indicaciones para instanciar objetos gráficos de forma adecuada:</p> <ul style="list-style-type: none"> <li>- Al extender <b>Graphic_Object</b> se tiene acceso a las propiedades de objeto y relaciones: <ul style="list-style-type: none"> <li>a) <b>color_RGB</b> (Color del marcador)</li> <li>b) <b>shape_named</b> (nombre de la forma: círculo, etc.)</li> <li>c) <b>width</b> (tamaño del objeto, 10%, 20%, etc.)</li> <li>d) <b>contains</b> (Relación con otro objeto gráfico, por ejemplo: <b>Picture</b>, para indicar la imagen con la que está asociada)</li> <li>e) <b>Text_Value</b> para asociar un texto al objeto (Etiqueta del marcador)</li> </ul> </li> <li>- Las siguientes clases se crean como extensión de <b>Primary_Graphic_Representation</b> : <ul style="list-style-type: none"> <li>a) <b>3D_Model</b>, para representar modelos 3D.</li> <li>b) <b>Light</b>, para representar luces en la escena, así como las clases correspondientes a los diferentes tipos de luces (<b>DirectionalLight</b>, <b>Point_Light</b>, <b>Spot_Light</b>)</li> </ul> </li> <li>- Se ha creado la clase <b>"Material_Attribute"</b> extendiendo a <b>"owl:Thing"</b>, y su extensión <b>"Material"</b>, para representar el material de una modelo 3D, así como los componentes de la luz 3d (<b>Light</b>)</li> <li>- Se han creado en <b>"Graphic_Object"</b> las propiedades: <ul style="list-style-type: none"> <li>a) <b>URL</b>, para indicar una URL asociada al objeto gráfico, puede ser una imagen, o un modelo 3D.</li> <li>b) <b>color_RGBA</b> para representar colores con la componente Alpha de transparencia. En VISO solamente había colores RGB. De los que se reutilizan las subpropiedades <b>"color_RGB Blue/Green/Red"</b>.</li> </ul> </li> <li>- Se han creado tanto en <b>"Material"</b>, como en <b>"3D_Model"</b> y <b>"Light"</b>, las propiedades <b>"emission"</b>, <b>"diffuse"</b>, <b>"ambient"</b> y <b>"specular"</b>, para representar las componentes de los materiales y de la luz.</li> <li>- Se ha creado en <b>"Material"</b> y en <b>"3D_Model"</b> la propiedad <b>"shininess"</b>, para representar dicha componente en el material de un objeto 3D.</li> </ul> |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|  |  |
|--|--|
|  | <p>- Se ha creado la clase “<b>RelationDisplayer</b>” como extensión de “<b>Primary Graphic Representation</b>”, para representar objetos que muestran información sobre las relaciones de un objeto con otros.</p> <p>- Se ha creado la clase “<b>RingRelationDisplayer</b>” para representar el control de Anillo usado para representar el porcentaje de objetos con el mismo valor para cada propiedad. Un esquema de este objeto puede verse en la Figura 39.</p> |
|--|--|

Tabla 14 Descripción del objeto gráfico Escena del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados

| Objeto del modelo | Conceptos de ontología GRAPHICS de VISO utilizados y/o extendidos   |
|-------------------|---|
| Escena            | <p>La escena contiene también información del tipo de representación gráfica. Esto implica una complejidad importante, que se va a ver enriquecida con las posibilidades de filtrado, o relación que se establecen con las diferentes propiedades del Elemento Visual. Sin embargo, un tipo base tiene que estar asociado a la escena. Para indicar ese tipo se utilizarán conceptos de la ontología VISO, en concreto aquellos que extienden el concepto “Graphic Representation” que son:</p> <ul style="list-style-type: none"> <li>- <b>Dynamic Graphic Representation</b></li> <li>- <b>Graphic Space</b></li> <li>- <b>Hybrid GRaphic REpresentation</b></li> <li>- <b>Link Diagram</b></li> </ul> <p>Así como las extensiones de dichos conceptos <b>Chronological Link Diagram</b> y <b>Map</b>. Sobre estos conceptos se ha extendido el concepto <b>Chronological Link Diagram</b>, para permitir la visualización de datos temporales simultáneamente y/o linealmente. Los conceptos introducidos son:</p> <ul style="list-style-type: none"> <li>- <b>Serial_Time_Visualization.</b> <ul style="list-style-type: none"> <li>- <b>TimeLine.</b></li> </ul> </li> <li>- <b>Simultaneous_Time_Visualization.</b> <ul style="list-style-type: none"> <li>- <b>Hypercube.</b></li> <li>- <b>TimeLayer.</b></li> </ul> </li> </ul> <p>Con el concepto <b>TimeLine</b> se representan escenas que permiten visualizar objetos en un momento concreto del tiempo, por ello se representa como extensión se <b>Serial_Time_Visualization</b>.</p> <p>Con el concepto <b>TimeLayer</b> se representan escenas que permita la representación de elementos en diferentes etapas del tiempo, visualizando al mismo tiempo su distribución espacial por capas, en determinados momentos de tiempo.</p> <p>Todos estos conceptos, combinándolos adecuadamente, cubren un amplio espectro de sistemas de tipos de representación. De cualquier forma, si fueran necesarios conceptos adicionales, únicamente habría que añadir más, extendiendo la ontología VISO. Esta combinación es posible porque los conceptos no son disjuntos.</p> |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tabla 15 Descripción del objeto gráfico Agrupación del modelo y su correspondencia con los conceptos de la ontología GRAPHICS de VISO que serán utilizados

| Objeto del modelo | Conceptos de ontología GRAPHICS de VISO utilizados y/o extendidos   |
|-------------------|---|
| Agrupación        | <p>El modelo propuesto permite definir si se van a agrupar los datos a visualizar, y cómo lo van a hacer.</p> <p>Para indicar este comportamiento del sistema de visualización también se reusará de nuevo la ontología VISO.</p> <p>En concreto, para representar este comportamiento se hará referencia al concepto Grouping Diagram, que extiende a <b>Primary Graphic Representation</b>.</p> <p>A priori este hecho podría incluirse en el tipo de representación de la escena, sin embargo, según el modelo este comportamiento no sólo puede afectar a la escena, sino que puede ser diferente en función del Elemento Visual. Es decir, que un Elemento Visual podría agruparse de una forma y otro mediante otra técnica diferente. Por ello se ha extendido <b>Grouping Diagram</b>, creando los conceptos:</p> <ul style="list-style-type: none"><li>- <b>Clustering_Diagram</b>, que agrupará los datos mediante un marcador representante.</li><li>- <b>HeatMap_Diagram</b>. En este caso, los datos se agruparán mediante un mapa de calor.</li></ul> <p>De esta forma en la ontología finalmente diseñada, habrá que poder relacionar, tanto la escena, como los elementos visuales, con estas técnicas, o comportamientos, que definirán la forma de agrupar la información a visualizar.</p> |

#### 7.2.2. Ontologías espaciales y temporales

Además de los conceptos gráficos, hay otra serie de conceptos a incluir en el modelo. Unos relacionados con la referenciación espacial de una instancia visual y otros con la especificación de los datos temporales. De igual forma que sucedía con la parte gráfica, la comunidad científica ya ha desarrollado varias ontologías que representan esta información. Por lo que procede realizar una investigación y análisis de estos trabajos, con el fin de reutilizarlos, y/o extenderlos. Seguidamente se exponen las necesidades de nuestro modelo en los ámbitos espacial y temporal, así como las decisiones que se han adoptado.

##### Referenciación espacial.

La representación de un punto en el espacio es un tema que no está tratado con la suficiente profundidad en la ontología VISO. En esta ontología, para posición un objeto se crean instancias de la entidad GraphicAttributeContinuous, que deben ser manejadas por el lenguaje RVL de forma adecuada para representar el objeto en la posición adecuada.

Mediante esta forma de trabajar, una coordenada tiene el mismo valor semántico que un color, o un área. Es cierto que todos son atributos continuos, y que podrían ser también instancias de la entidad "Spatial Attribute", que indica el tipo de atributo que es.

Por lo tanto, incluso instanciando debidamente un atributo de coordenada con VISO, es complejo conocer el tipo de coordenada que es, y por supuesto más datos, como el sistema de

coordenadas que utiliza. En VISO estos conceptos se gestionan mediante RVL de forma adhoc. Es decir, el código desarrollado en RVL sabe que los valores de dichos atributos son posiciones espaciales, especificadas mediante un determinado sistema de coordenadas y el desarrollador debe de conocer esta información. Y esto sería posible en el caso de que los datos se han instanciado correctamente, definiéndose como un tipo doble, de no hacerlo, no habría forma de diferenciarlo de un color, u otro atributo continuo.

En el modelo que se propone representar con esta investigación no se puede asumir esta forma de tratamiento de las posiciones espaciales. En el modelo, los datos del dominio pueden estar en diferentes sistemas de coordenadas. Por ejemplo, si el dominio de datos usa datos georeferenciados, generalmente se usarán sistemas de coordenadas geográficas como WGS 84 (World Geodetic System) (Slater J.A. 1998), o el ETRS 89 (European Terrestrial Reference System) (IGN 2007), o también sistemas de coordenadas proyectadas, como es el caso de UTM (Universal Transverse Mercator). De igual manera, aunque generalmente el sistema de visualización usará sistemas de coordenadas cartesianas para el dibujado, puede usar otro tipo de sistemas de coordenadas. Por ejemplo, en el sistema de visualización de la aplicación Museum of the World (Ver Figura 20, en Capítulo 4 – Desarrollos Web para la visualización de datos del patrimonio cultural) el sistema de coordenadas de visualización tiene dos coordenadas, el continente y el tiempo de producción del objeto.

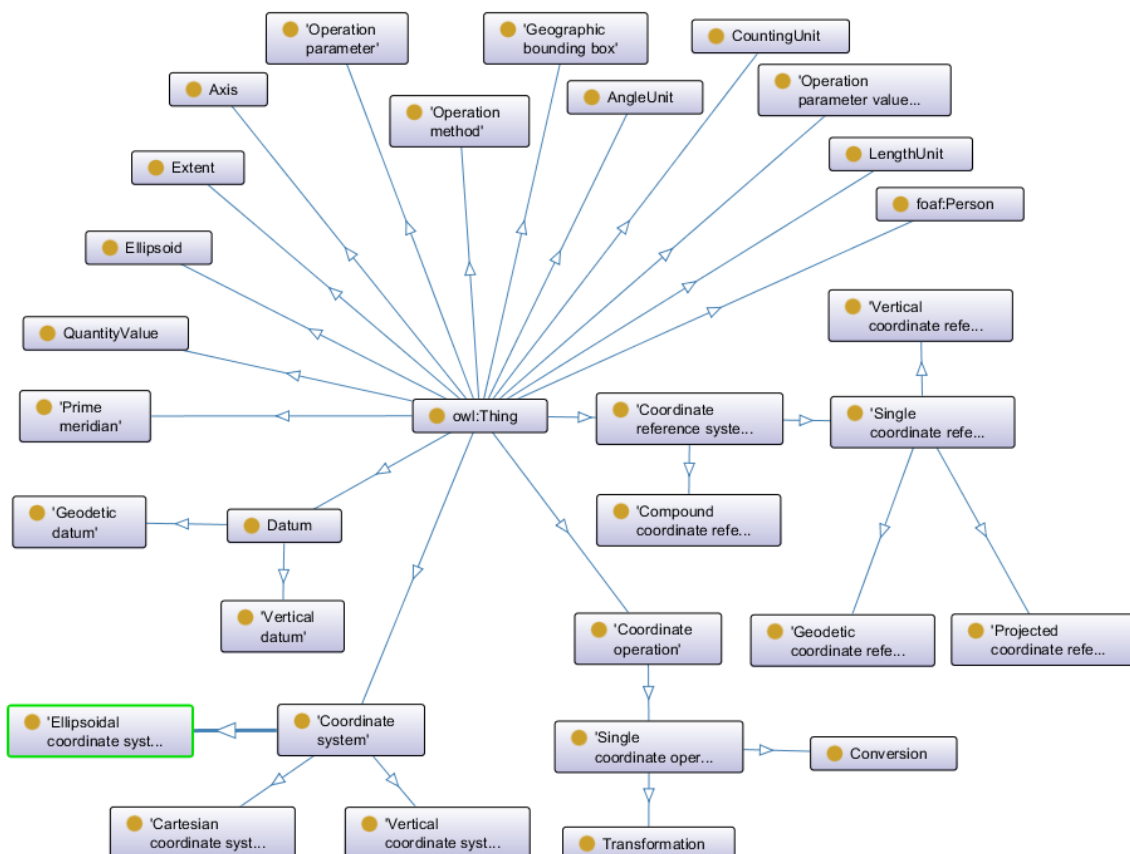


Figura 45 Diagrama de la ontología de Sistemas de Coordenadas del IGF (Abadie, Atemezing 2015).

Por ello, en este tipo de sistemas, donde tanto los datos del dominio, como la visualización, pueden usar diferentes sistemas de coordenadas, es muy importante definir de forma adecuada la localización espacial. Esta localización debe de ir vinculada a un sistema de coordenadas para



## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

que el sistema de visualización tenga la información necesaria para realizar las transformaciones correspondientes.

La comunidad científica ha desarrollado varias ontologías y vocabularios para representar la referenciación geoespacial, pero es en la actualidad y en los próximos años cuando realmente se van a aplicar estos resultados de forma importante (Claramunt 2020) . Una de las aplicaciones más relevantes y con más futuro de estos trabajos se puede encontrar en el campo del patrimonio cultural (Nishanbaev, Champion et al. 2019). Aunque hay varios desarrollos, la gran mayoría reutilizan los conceptos de la ontología GeoSPARQL (Open Geospatial Consortium 2011).

En el modelo de este trabajo, esta información es importante, pero únicamente, porque algunos de los dominios de datos que se van a visualizar pueden usar estos trabajos para referenciar los datos espacialmente. En este modelo, lo fundamental es poder indicar los sistemas de coordenadas usado por el sistema de visualización y el dominio de datos.

Por ello, se ha elegido la ontología de sistemas de coordenada desarrollada por Institut National de l'Information Géographique et Forestière (IGF) (Abadie, Atemezing 2015) . . En la Figura 45 se muestra un diagrama con los conceptos de esta ontología y sus relaciones. En la Figura 45 se muestra un diagrama con los conceptos de esta ontología y sus relaciones.

Conceptos referentes a los sistemas de coordenadas aparecen en varias ontologías, como es el caso de la propuesta por CIDOC CRM, la ontología CRMgeo (Hiebel, Martin et al. 2015) o Geo-Core (Espinoza-Arias, García-Delgado et al. 2019), desarrollada para la información del Instituto Geográfico Nacional . Sin embargo, la ontología elegida, no está vinculada a otro dominio y está diseñada únicamente para representar los sistemas de coordenadas. Esto hace que tenga conceptos que permiten definir operaciones de transformación entre sistemas de coordenadas, y otro tipo de operaciones propias de los sistemas GIS.

Además, también hace uso de la norma ISO 19111:2007 de Información Espacial Geográfica referenciada por coordenadas (ISO 2019). Esta ontología está soportada por la entidad pública francesa que la creó y también está incluida en DBPedia

En la Tabla 16 se exponen los elementos del modelo y su relación con los conceptos de la ontología de sistemas de coordenadas del IGF que serán reutilizados y/o extendidos.

*Tabla 16 Objetos del modelo a representar y su correspondencia con los conceptos de la ontología de Sistemas de Coordenadas del IGF que serán utilizados.*

| Objeto del modelo | Conceptos de ontología de Sistemas de Coordenadas utilizados y/o extendidos   |
|-------------------|---|
| Entorno Espacial  | <p>En el Entorno Espacial, que representa las características espaciales de los datos, tanto del dominio de datos, como del sistema de visualización, es fundamental el uso de un Sistema de Coordenadas.</p> <p>Por este motivo, el entorno debe de tener una relación con los siguientes conceptos y sus extensiones</p> <ul style="list-style-type: none"> <li>- <b>Coordinate System</b></li> <li>- <b>Cartesian coordinate system.</b></li> <li>- <b>Ellipsoidal coordinate system.</b></li> <li>- <b>Vertical coordinate system.</b></li> <li>- <b>Coordinate reference system (CRS)</b></li> <li>- <b>Geodetic coordinate reference system.</b></li> <li>- <b>Projected coordinate reference system.</b></li> <li>- <b>Vertical coordinate reference system.</b></li> </ul> <p>Mediante estos conceptos se puede representar perfectamente el tipo de Sistema de coordenadas, incluso se pueden asignar propiedades a los ejes (<b>Axis</b>), como la dirección de los mismos.</p> |
| Punto             | <p>Aunque no hay ningún objeto en el modelo descrito como punto espacial, es obvio que en algún momento tendrá que existir la posibilidad de definir puntos espaciales, independientemente del sistema de coordenadas.</p> <p>Aunque la ontología VISO contempla la posibilidad de asignar la posición de los objetos mediante la propiedad de datos “position”, no está vinculada a ningún sistema de coordenadas. Por ese motivo, se optará por usar el concepto <b>Point</b> de la ontología de sistemas de coordenadas de IGF, ya que está perfectamente integrada con estos conceptos.</p>   |

Además de la ubicación espacial, es muy importante tener en cuenta los problemas derivados por la granularidad y la incerteza en las diferentes referencias espaciales.

El sistema de visualización puede ser consciente de la incerteza de un dato espacial en el momento en que hay varias posibles ubicaciones de un mismo objeto. Sin embargo, hay ocasiones en que esta ubicación es única, pero también es incierta, en este caso el dominio de datos es el que debe proporcionar la información al sistema de visualización.

## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Lo mismo sucede con la granularidad. La ubicación espacial de un objeto puede estar indicada por unas coordenadas geográficas, en el caso de un mapa. Pero esto no quiere decir que el objeto esté en esa posición, sino que esa posición representa la región asociada, bien por estar en el centro geográfico, o por otro motivo. Por este motivo es el dominio de datos el que debe de proporcionar la información sobre si posición del objeto tiene incerteza debido a la granularidad.

Lógicamente, la ontología que se va a desarrollar debe de contemplar la posibilidad de definir esta incerteza con los conceptos y/o propiedades adecuadas.

### Referenciación temporal

La temporalidad es la otra característica fundamental de los datos a visualizar. Además, tal y como se ha comentado a lo largo del trabajo, debido a que hay una frecuencia importante de imprecisión en los datos temporales, el modelo ha de contemplar los datos temporales de los objetos mediante un rango.

Dadas las características del modelo, el Entorno Temporal, que establece un marco de referencia temporal para los datos, tanto para la visualización, como para los del dominio de datos, necesita también un intervalo de tiempo, además de una unidad de medida temporal.

Al igual que sucede con las referencias espaciales, las referencias temporales también tienen diferentes sistemas de referencias. Desde cambios simples, que afecten a una unidad temporal, como la hora, como pueda ser el utilizar 12 horas, o 24 horas, hasta utilizar diferentes calendarios, como el islámico, o el chino. Calendarios cuya organización cambia en el tiempo, debido a los continuos ajustes que han sufrido a lo largo de la historia por estar basados en ciclos lunares y solares, que era necesario corregir para fijar las estaciones.

Estos calendarios, y así como otros términos y los nombres de diferentes unidades de tiempo están referenciadas en la norma ISO 19108:2002 (ISO 2021)

Los diferentes calendarios, o al menos los definidos y usados por la humanidad están debidamente referenciados y hay un criterio con un alto nivel de uniformidad. Esto no sucede con la representación de las eras geológicas, utilizadas para referenciar temporalmente fósiles y minerales, en las que en la actualidad hay varias ontologías con diferentes puntos de vista (Cox, Richard 2015, Xiaogang 2021)

Afortunadamente, en el ámbito de la referencia temporal, el consorcio W3C ha diseñado e implementado la ontología TIME (W3C 2020) que es muy utilizada por la mayoría de desarrollos. Esta ontología de conceptos temporales, que hace uso de la norma ISO 19108:2022, está diseñada para describir las propiedades temporales de los recursos descritos en las páginas web. Proporciona un vocabulario para expresar hechos sobre relaciones topológicas sobre instantes e intervalos, junto a la información sobre su duración y ubicación temporal. En la Figura 46 se muestra un diagrama de la ontología TIME con los conceptos y las relaciones entre ellos.

Por lo tanto, para expresar los diferentes entornos temporales, tanto del dominio de datos, como del sistema de visualización, la ontología diseñada en este trabajo usará los conceptos de la ontología TIME de W3C.

Esta asunción en ocasiones requerirá un trabajo extra en el dominio de datos. Simplemente porque los datos temporales del dominio de datos no están expresados numéricamente, o usan términos de vocabularios muy utilizados en este ámbito (Getty 1997a, Golden, Shaw 2016), o

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

bien porque usan su propio terminología, en ocasiones sin seguir un esquema homogéneo. En la Tabla 17 se muestran una serie de términos, a modo de ejemplo, usados en patrimonio cultural. Algunos de estos términos podrían referenciarse directamente con ontología TIME de W3C, otros si, pero con un pequeño procesamiento, y otros requerirían un procesamiento importante.

*Tabla 17 Términos temporales usados en patrimonio cultural y conceptos relacionados con ontología TIME*

| Termino Domino Datos | Concepto ontología TIME | Significado en Años   | Vocabulario |
|----------------------|-------------------------|-----------------------|-------------|
| Siglo V a.C.         | Century                 | 500 antes de Cristo   | TIME        |
| Año 500              | Year                    | 500 después de Cristo | TIME        |
| Alta Edad Media      | ---                     | 400 d.C -700 d.C      | PERIO.DO    |
| 1500 BP              | ---                     | 500 d.C               | PERIO.DO    |
| 1500 c.a.            | ---                     | Aprox. 1500 d.C       | ----        |

Por ejemplo, la etiqueta “c.a.”, que una abreviatura de “circa” en latín, al igual que el uso del carácter “?”, se usan en patrimonio cultural para transmitir que la información es aproximada, pero no hay certeza de su exactitud. Al ser una lengua muerta, o símbolos, es muy sencillo de procesar, ya que no hay variantes. Sin embargo, cuando se trata de otras etiquetas como “BP”, que proviene del inglés “Before Present”, usada para indicar dataciones de carbono, es más complejo, ya que puede cambiar en función de la lengua, en español sería “AP”. El procesamiento de estos datos sería ligeramente más costoso, pero no sería muy grande. Sin embargo, etiquetas como “Alta Edad Media”, no sólo dependen de la lengua, sino también de que estén, o no referenciadas con un vocabulario, como PERIO.DO (Shaw, Rabinowitz et al. 2016) Conceptos referentes a los sistemas de coordenadas aparecen en varias ontologías, como es el caso de la propuesta por CIDOC CRM, la ontología CRMgeo (Hiebel, Martin et al. 2015) o Geo-core y BTN100 (Espinoza-Arias, García-Delgado et al. 2019), desarrollada para la información del Instituto Geográfico Nacional. Sin embargo, la ontología elegida, no está vinculada a otro dominio y está diseñada únicamente para representar los sistemas de coordenadas. Esto hace que tenga conceptos que permiten definir operaciones de transformación entre sistemas de coordenadas, y otro tipo de operaciones propias de los sistemas GIS (Golden, Shaw 2016).

De cualquier forma, este tipo de datos ha de ser procesado para ser válido para su visualización. En esta etapa de la investigación se exigirá que los datos de la ontología de dominio mantengan una unidad temporal que se pueda definir mediante la ontología TIME de W3C.

En la Tabla 18 se exponen los elementos del modelo y su relación con los conceptos de la ontología TIME de W3C que se usarán y/o extenderán.

*Tabla 18 Objetos del modelo y conceptos de la ontología TIME utilizados y/o extendidos.*

| Objeto del modelo | Conceptos de ontología TIME utilizados y/o extendidos |
|-------------------|---|
|-------------------|---|

|                  |  |
|------------------|--|
| Entorno Temporal | <p>En el Entorno Temporal, que representa las características temporales de los datos, tanto del dominio de datos, como del sistema de visualización, es fundamental el uso de un Sistema de referenciación temporal.</p> <p>Por este motivo, el entorno debe de tener una relación con el concepto <b>Instant</b>. Este concepto representa un instante de tiempo concreto. Es una fecha asociada a un tipo de calendario concreto.</p> <p>Como el entorno temporal está definido por un intervalo de tiempo, siempre deberán de existir dos instancias del concepto <b>Instant</b> para definir dicho intervalo.</p> <p>La fecha marca un instante concreto de tiempo, pero es muy importante conocer la unidad temporal asociada a ese instante. Por ejemplo, la fecha 01/01/1000 00:00:00 puede entenderse como que hace referencia a una hora, un día, un año, un siglo, o un milenio.</p> <p>Por ello es fundamental que el entorno temporal permita asociar la unidad temporal asociada al momento de tiempo.</p> <p>En la ontología TIME, dicho concepto se representa mediante <b>TemporalUnit</b>, que define una duración, o unidad temporal.</p> <p>Mediante estos conceptos se puede representar perfectamente el tipo de Sistema de coordenadas, incluso se pueden asignar propiedades a los ejes (Axis), como la dirección de los mismos.</p> |
|------------------|--|

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

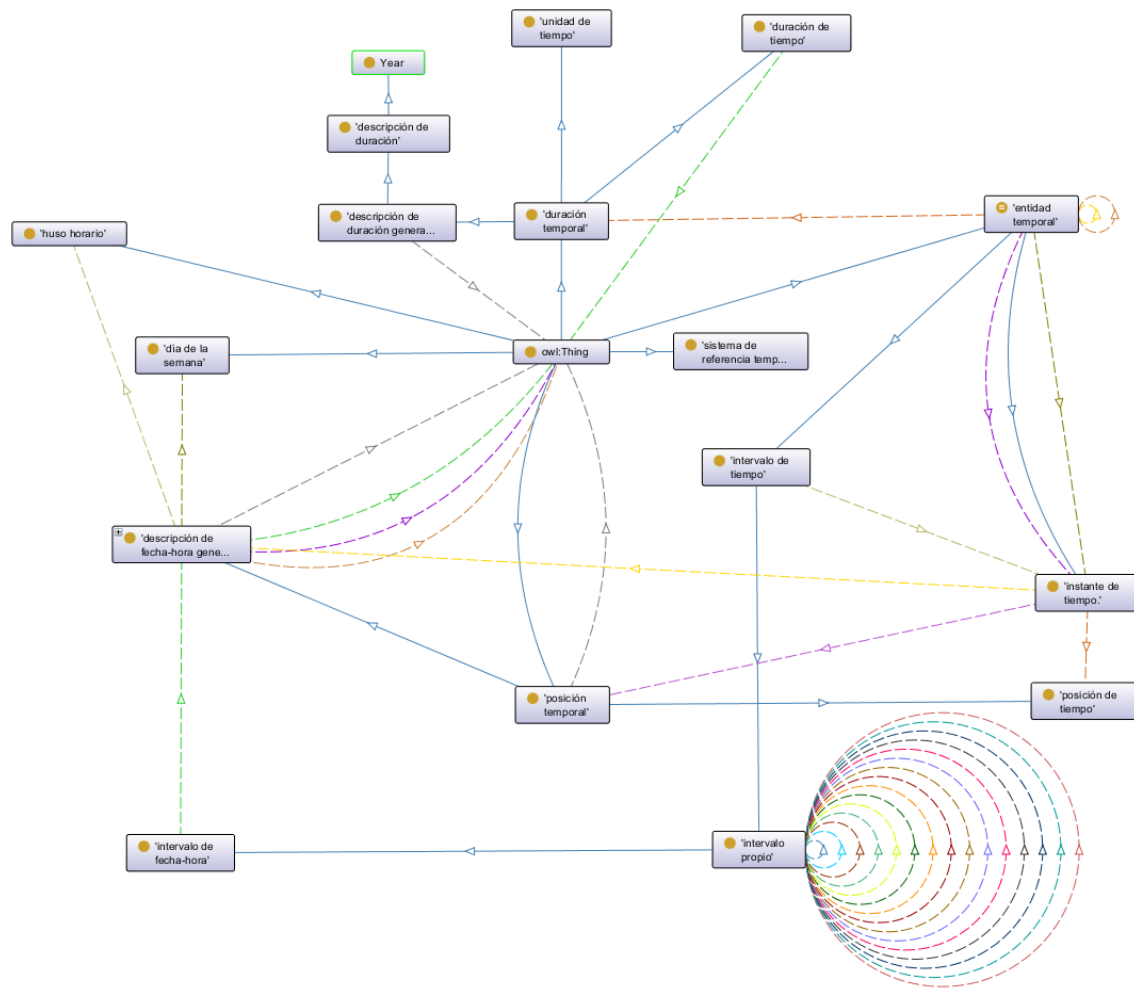


Figura 46 Diagrama de la ontología TIME (W3C 2020)

Al igual que sucede con las referencias espaciales, también existe incerteza y problemas con la granularidad en las referencias espaciales.

En esta investigación se va exigir que mediante un procesamiento previo sobre el dominio de datos se defina una propiedad que indique este tipo de incerteza. Obviamente, cuando un objeto tenga varias referencias temporales, ese objeto tendrá incerteza. Pero cuando sólo se tiene una, como se ha comentado anteriormente, se usan abreviaturas como “circa”, o símbolos como “?” para indicar que la datación no es exacta.

Con la granularidad no es necesario dicho procesamiento. Esto es debido a que es posible conocer esta información, mediante las unidades temporales del dominio de datos y la del sistema de visualización. Únicamente no será incierto cuando el intervalo de datación sea de una unidad temporal y la unidad temporal del sistema de visualización sea igual, o mayor. Por ejemplo, si la unidad temporal es el año en ambos sistemas, y un objeto está datado entre 1500 y 1600 (Siglo XVI), el objeto aparecerá datado en cualquier filtro por año que se haga en ese siglo, sin embargo, ese dato es incierto y deberá de identificarse como tal. Sin embargo, si la unidad temporal del sistema de visualización fuera el siglo, dicho dato no sería incierto, ya que efectivamente la datación corresponde a un único siglo.

## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

### *7.2.3. Aplicación de METHONTOLOGY*

Seguidamente, se describen las tareas realizadas, dentro de las tareas de METHONTOLOGY, para construir la ontología resultado de esta investigación.

#### *Especificación*

En esta actividad hay que especificar la razón por la que se construye la ontología, como se utilizará, así como sus usuarios.

La razón por la que se construye esta ontología es la de representar el modelo descrito en el capítulo anterior. Con este modelo se podría representar como se van a visualizar, e interactuar con los datos de cualquier grafo de conocimiento con datos espacio temporales del ámbito de patrimonio cultural.

El primer resultado de esta ontología sería de utilidad para desarrollos de sistemas de visualización, que tienen como objetivo representar visualmente el contenido de estos grafos de conocimiento. Al instanciar debidamente la ontología, la aplicación podrá acceder a esta especificación y conocer:

- Qué conceptos de la ontología de dominio se van a representar y cómo. La ontología debe definir cómo visualizar la escena y transformar los datos desde el dominio de datos hasta el sistema de visualización. También se ha de alcanzar un nivel de detalle gráfico importante, así como si se van a realizar agrupaciones de los datos, de qué forma, y como se va a tratar visualmente la incerteza.
- La forma de acceder a los datos del dominio de datos. Ya que en la ontología no sólo se ha de especificar cómo se visualizan, sino que además se tiene que establecer cómo acceder a esta información. Es muy habitual que un objeto de un sistema de visualización esté compuesto por varias propiedades de diferentes conceptos del dominio de datos. El acceso a esta información, generalmente se realiza mediante consultas, que se especificarán en la ontología.
- Cómo interactuar con estos datos. El usuario de un sistema de visualización debe de poder realizar filtros por diferentes propiedades, así como poder examinar las relaciones entre los objetos que se visualizan. Esta información se ha de especificar en la ontología.

Con esta información, se puede desarrollar una herramienta genérica, o ad hoc, capaz de visualizar de diferentes formas, el contenido de los grafos de conocimiento, únicamente procesando la ontología y desarrollando las funciones de visualización e interacción adecuadas.

Un segundo resultado de esta ontología, una vez se haya introducido información de diferentes grafos de conocimiento, sería el de servir como base de un sistema de recomendación, capaz de recomendar cómo visualizar la información de un grafo de conocimiento con datos espacio temporales.

Este segundo resultado sobrepasa los objetivos de esta investigación. Pero tanto en el primero, como en el segundo resultado, los usuarios de esta ontología serían aplicaciones informáticas.

En el primer caso, serían sistemas de visualización, en el segundo sistemas de recomendación.



### Clasificación

La actividad de clasificación es la actividad más compleja de METHONTOLOGY y es donde se hace el principal trabajo de diseño, ya que se va a generar un diseño conceptual, partiendo de una descripción informal del modelo.

En este caso, en la descripción del modelo ya se han generado tablas con diferentes conceptos y su significado, propiedades, etc. La información de estas tablas, será considerada en las diferentes tareas de esta actividad, pero también el resto de la descripción informal.

También será tenido en cuenta para el desarrollo de esta actividad, el análisis de ontologías externas, explicado en este capítulo, tanto de visualización como de referencias espaciales y temporales.

En la Figura 47 se expone un diagrama donde se muestran las tareas de la actividad conceptualización de esta metodología.

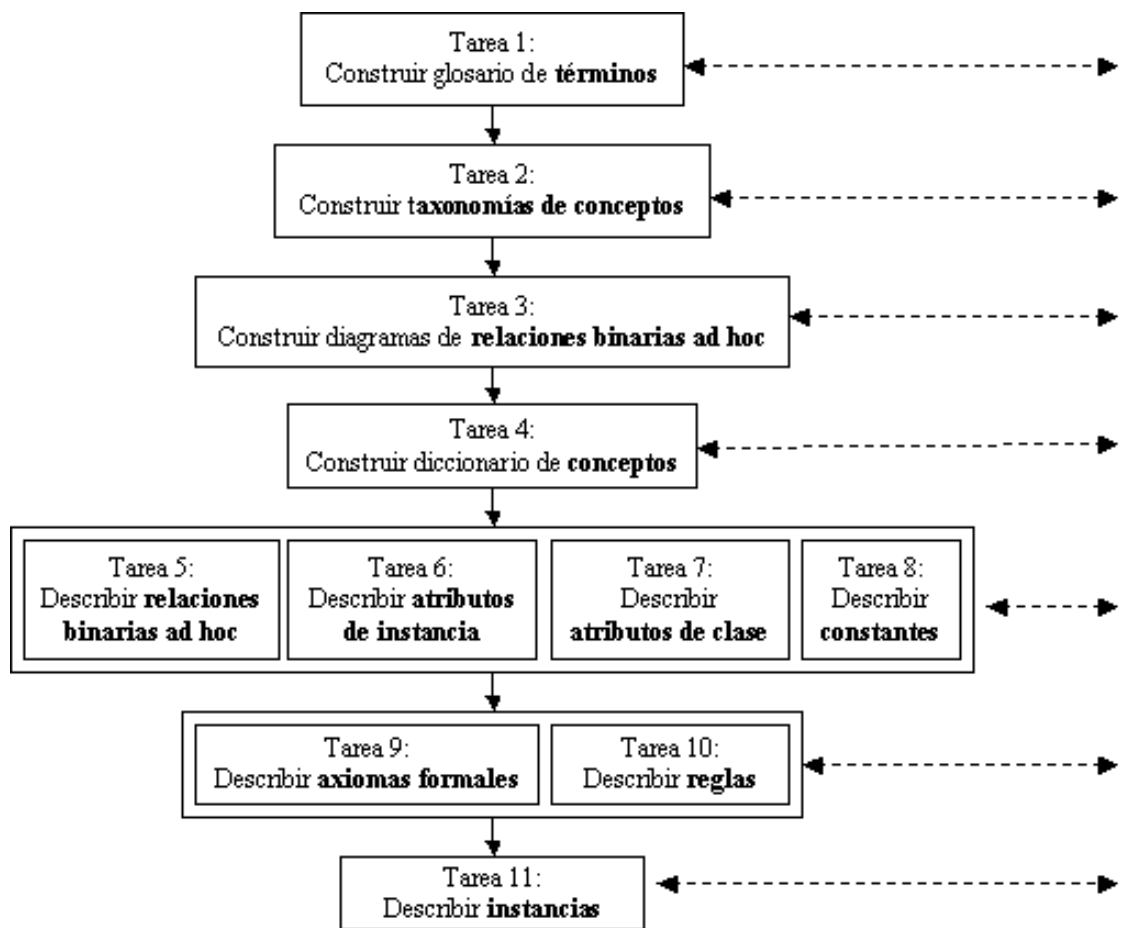


Figura 47 Tareas de la conceptualización en METHONTOLOGY. Fuente:(Corcho, Fernández-López et al. 2005)

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tarea 1: Glosario de términos.

Consiste en construir un glosario con los términos más importantes del dominio (conceptos, relaciones, atributos, etc.), su descripción en lenguaje natural, sinónimos, acrónimos y el tipo de datos que es.

El glosario tendrá un formato tabular con los siguientes títulos:

- Nombre: Nombre del término.
- Sinónimos: Sinónimos del término.
- Acrónimos: Acrónimos del término.
- Descripción: Descripción del término.
- Tipo del término: Que puede tomar los siguientes valores: Constante, Concepto, Atributo, Atributo de Instancia, Relación.

En la Tabla 19 se enumera una selección de los términos más importantes.

*Tabla 19 Selección del Glosario de términos de actividad de conceptualización de METHONTOLOGY para construir ontología de visualización.*

| Nombre   | Sinónimos       | Acrónimos | Descripción  | Tipo      |
|--|-----------------|-----------|--|-----------|
| Concepto de dominio                              |                 |           | Concepto de la ontología de dominio que se va a visualizar   | Concepto  |
| Propiedad de dominio                             |                 |           | Propiedad de un concepto de la ontología de dominio  | Concepto  |
| Concepto Visual                                  |                 |           | Objeto a visualizar en el sistema de visualización.  | Concepto  |
| Propiedad Visual                                 |                 |           | Propiedad de un objeto visual  | Concepto  |
| Instancia de dominio                             |                 |           | Instancia del dominio de datos.  | Instancia |
| Instancia Visual                                 | Elemento Visual |           | Instancia que se visualiza en el sistema de visualización. Es un objeto gráfico que se va a visualizar en la escena que está relacionado con una instancia de un concepto de dominio de datos. | Instancia |
| Representado( Concepto Visual, Concepto Dominio) |                 |           | Concepto de dominio representado por un Concepto Visual  | Relación  |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|   |   |  |   |           |
|---|---|--|---|-----------|
| Propiedad_de(<br>Concepto Visual,<br>Propiedad Visual)              |   |  | Propiedad Visual asociada con un<br>Concepto Visual   | Relación  |
| Consulta  |   |  | Consulta expresada en un<br>lenguaje de consultas formal (ej.<br>SPARQL)  | Concepto  |
| Obtiene Valor<br>de( Propiedad<br>Visual, Consulta)                 |   |  | Consulta usada para obtener el<br>valor de una Propiedad Visual.  | Relación  |
| Propiedad<br>temporal(<br>Concepto Visual,<br>Propiedad Visual)     |   |  | Propiedad Visual usada para<br>obtener la referencia temporal<br>de las Instancias Visuales de un<br>Concepto Visual  | Relación  |
| Propiedad<br>Espacial(<br>Concepto Visual,<br>Propiedad Visual)     |   |  | Propiedad Visual que permite<br>referenciar espacialmente las<br>Instancias Visuales relacionadas<br>con un Concepto Visual.                                | Relación  |
| Propiedad<br>filtrable<br>(Concepto Visual,<br>Propiedad Visual)    |   |  | Propiedad Visual por la que el<br>usuario podrá realizar filtros en<br>el sistema de visualización.   | Relación  |
| Propiedad<br>relacionable(<br>Concepto Visual,<br>Propiedad Visual) |   |  | Propiedad Visual por la que el<br>usuario podrá visualizar<br>instancias visuales que se estén<br>visualizando y tengan el mismo<br>valor en esa propiedad. | Relación. |
| Calendario  | Sistema<br>de<br>Referencia<br>Temporal |  | Sistema para definir el tiempo,<br>generalmente basado en ciclos<br>solares, y/o lunares.   | Concepto  |
| Unidad de<br>tiempo   | Cantidad<br>de<br>tiempo                |  | Magnitud usada para medir el<br>tiempo.   | Concepto. |
| Instante de<br>tiempo   | Momento                                 |  | Momento, o instante concreto<br>de tiempo, definido por un<br>calendario y una unidad de<br>tiempo.   | Concepto  |
| Intervalo de<br>tiempo  |   |  | Intervalo temporal definido por<br>dos instantes de tiempo, usando<br>un calendario determinado   | Concepto  |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                                      |                |  |   |           |
|--------------------------------------|----------------|--|---|-----------|
| Entorno temporal                     | Marco temporal |  | Marco usado para definir el tiempo en un contexto, o conjunto de datos determinado. Está definido por un intervalo de tiempo, compuesto por datos instantes de tiempo, expresados en un calendario concreto y en una unidad temporal determinada.   | Concepto  |
| Sistema de Coordenadas               |                |  | Sistema de coordenadas espacial.  | Concepto  |
| Punto                                |                |  | Punto del Espacio   | Concepto  |
| Sistema Cartesiano                   |                |  | Sistema de coordenadas cartesiano   | Concepto  |
| Sistema de coordenadas de referencia |                |  | Sistema de coordenadas de referencia  | Concepto  |
| Sistema geodético                    |                |  | Sistema geodético de coordenadas de referencia  | Concepto  |
| Datum                                |                |  | Datum   | Concepto  |
| WGS84                                |                |  | Datum WGS84   | Instancia |
| Entorno Espacial                     |                |  | Marco utilizado para definir las características espaciales de un contexto, o conjunto de datos. Está definido por dos puntos asociados a un sistema de coordenadas, o un sistema de coordenadas.   | Concepto  |
| Escena                               |                |  | Entorno de visualización. Define un sistema de visualización completo. Requiere la asociación con un entorno espacial, así como un entorno temporal, para el contexto espacio temporal de la visualización.<br><br>En la escena se representarán las instancias de los conceptos visuales que se asocien. | Concepto  |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|  |  |  |   |                       |
|--|--|--|---|-----------------------|
| Representado Visualmente( Concepto Visual, Escena)   |  |  | Concepto Visual que se representa visualmente en una escena.  | Relación              |
| Representación Gráfica                               |  |  | Representa el tipo de representación gráfica que se va a visualizar, puede ser básica (una imagen) , a un sistema interactivo que muestra mapas, relaciones, etc. | Concepto              |
| Tipo Representación( Escena, Representación gráfica) |  |  | Representación gráfica que se va a visualizar en una escena.  | Relación              |
| Representación Gráfica Interactiva                   |  |  | Representación gráfica en la que el usuario va a poder interactuar con los elementos que se representan.  | Concepto              |
| Representación Gráfica de Mapa                       |  |  | Representación gráfica  | Concepto              |
| Marcador   |  |  | Objeto gráfico usado para representar una Instancia Visual o un agrupamiento de estas.  | Concepto              |
| Figura geométrica                                    |  |  | Objeto gráfico 2D compuesto por una figura geométrica   | Concepto              |
| Círculo  |  |  | Objeto gráfico compuesto por un círculo   | Instancia             |
| Modelo 3D  |  |  | Objeto gráfico compuesto por un modelo 3D   | Concepto              |
| Color  |  |  | Color de una figura o un modelo   | Atributo de instancia |
| Imagen   |  |  | Enlace a imagen con la que se rellena una figura, o se pega sobre un modelo 3D  | Atributo de instancia |
| Textiles.png   |  |  | Imagen con iconos de textiles.  | Instancia             |
| Esfera.pfb   |  |  | Modelo 3D de una cubo con una imagen  | Instancia             |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|  |  |  |  |           |
|--|--|--|--|-----------|
| Modificador de Color   |  |  | Representa el proceso de asociar un color a una representación gráfica, en función del valor de la propiedad de una instancia visual.      | Concepto  |
| Modificador de Imagen  |  |  | Representa el proceso de asociar una imagen a de una representación gráfica, en función del valor de la propiedad de una instancia visual. | Concepto  |
| Modificador de Marcador( Marcador, Modificador)                    |  |  | Modificador de Imagen, o de Color asociado a un Marcador.  | Relación  |
| Propiedad_ asociada_a Modificador ( Propiedad Visual, Modificador) |  |  | Propiedad Visual asociada a una Modificador  | Relación  |
| Marcador de concepto( Concepto Visual, Marcador)                   |  |  | Marcador gráfico usado para representar en la escena las instancias del Concepto Visual  | Relación  |
| Marcador representante( Concepto Visual, Marcador)                 |  |  | Marcador gráfico para representar las agrupaciones de un concepto visual.  | Relación. |
| Agrupación por clusters  |  |  | Comportamiento que define el agrupamiento de instancias de un mismo concepto en grupos.  | Concepto  |
| Agrupamiento por mapa de calor                                     |  |  | Comportamiento que define el agrupamiento de instancias de un mismo concepto en mapa de calor  | Concepto  |
| Visualizar relaciones con enlaces                                  |  |  | Comportamiento que indica que las relaciones entre instancias se visualizan con enlaces  | Concepto  |
| Visualizar relaciones con marcador de Anillo                       |  |  | Comportamiento que indica que las relaciones entre instancias se pueden ver con un marcador de anillo                                      | Concepto  |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|  |  |  |   |          |
|--|--|--|---|----------|
| Control de tiempo lineal   |  |  | Comportamiento que indica que se pueden visualizar las instancias de un intervalo de tiempo concreto  | Concepto |
| Control de tiempo simultáneo   |  |  | Comportamiento que indica que se pueden visualizar las instancias y el intervalo de tiempo en el que están relacionadas de forma simultánea | Concepto |
| Comportamiento de la escena( comportamiento, escena)                       |  |  | Uno de los comportamientos asociados a una escena.  | Relación |
| Agrupamiento asociado a un concepto visual( agrupamiento, concepto visual) |  |  | Agrupamiento asociado a un concepto visual.   | Relación |
| Visualización de incerteza de forma individual                             |  |  | Visualización de la incerteza sin replicación de los objetos con datos inciertos.   |          |

Tarea 2. Construir taxonomía de conceptos.

Esta tarea consiste en construir una serie de taxonomías de conceptos que definen su jerarquía. Esta metodología de construcción de ontologías propone utilizar las relaciones taxonómicas de:

- Subclase-de. A es subclase de B, si todas las instancias de A, son también instancias de B.
- Descomposición Exhaustiva. Una descomposición de este tipo de un concepto A, se trata de un conjunto de subconceptos, donde toda instancia de A, tiene que ser también instancia de uno, o varios de estos subconceptos. Por lo tanto, estos subconceptos pueden tener instancias y subclases comunes.
- Descomposición Disjunta. Una descomposición disjunta de un concepto A consta de una serie de subconceptos de A, que no pueden tener instancias comunes. Además, pueden existir instancias de A, que no sean instancias de ninguno de los conceptos de esta descomposición.
- Partición. La partición de un concepto A está compuesta de un conjunto de conceptos, donde cada instancia de A, tiene que ser instancia de uno y sólo uno de estos conceptos. Estos conceptos no pueden tener instancias comunes.



CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Estas relaciones están definidas en la OKBC Ontology (Chaudhri, Farquhar et al. 1998) y la Frame Ontology (FARQUHAR, FIKES et al. 1997) . En la Figura 48, Figura 49, Figura 50, Figura 51, Figura 52 y Figura 53, se muestran diagramas con las relaciones de las taxonomías extraídas de los conceptos del glosario de términos.

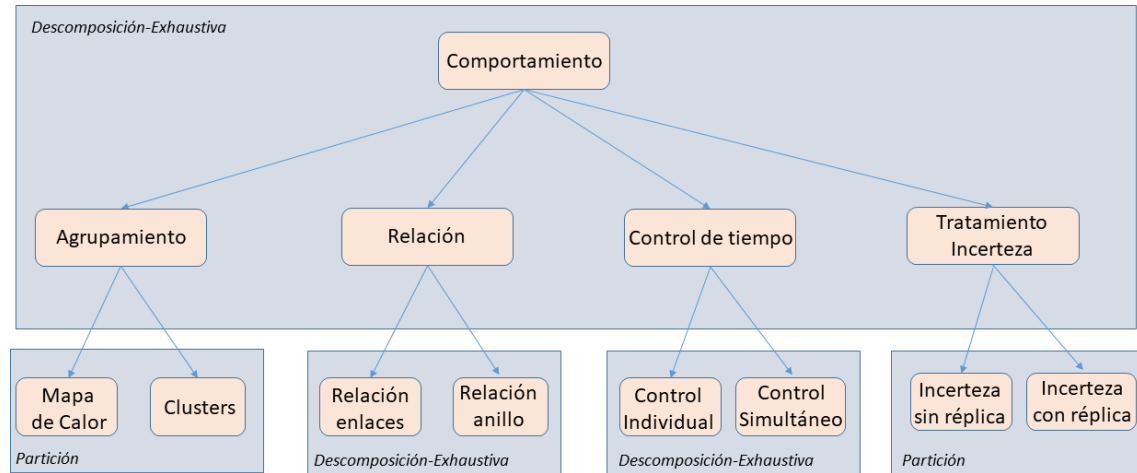


Figura 48 Relación taxonómica de Descomposición-Exhaustiva y Partición, entre el concepto Comportamiento y sus subconceptos.

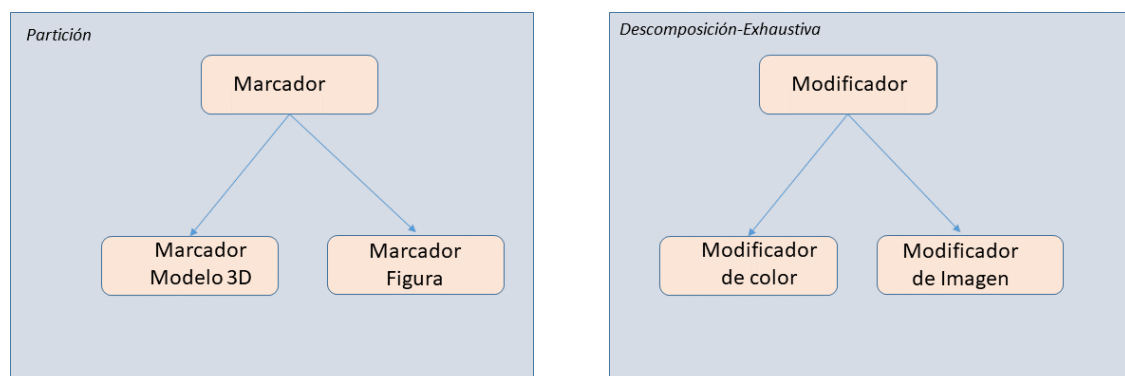


Figura 49 A la izquierda, relación taxonómica de Partición entre el concepto Marcador y sus subconceptos, a la derecha relación taxonómica de Descomposición Exhaustiva del Concepto Modificador y sus subconceptos.

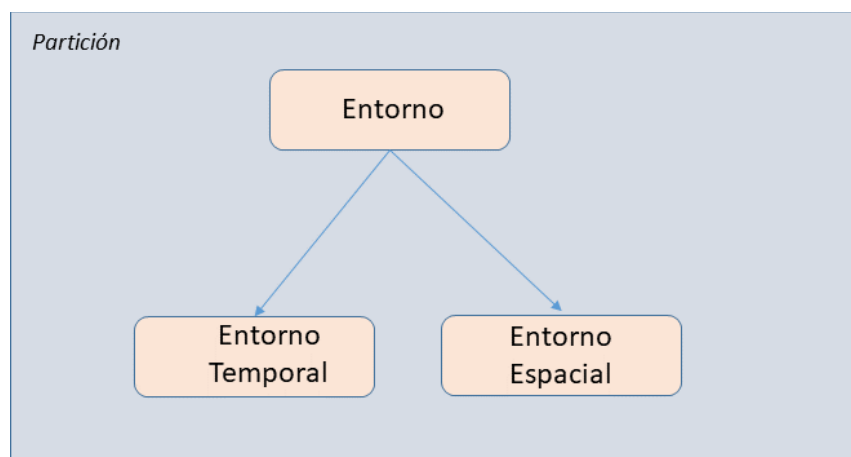


Figura 50 Relación taxonómica de Partición entre el concepto Entorno y sus subconceptos.

**CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL**

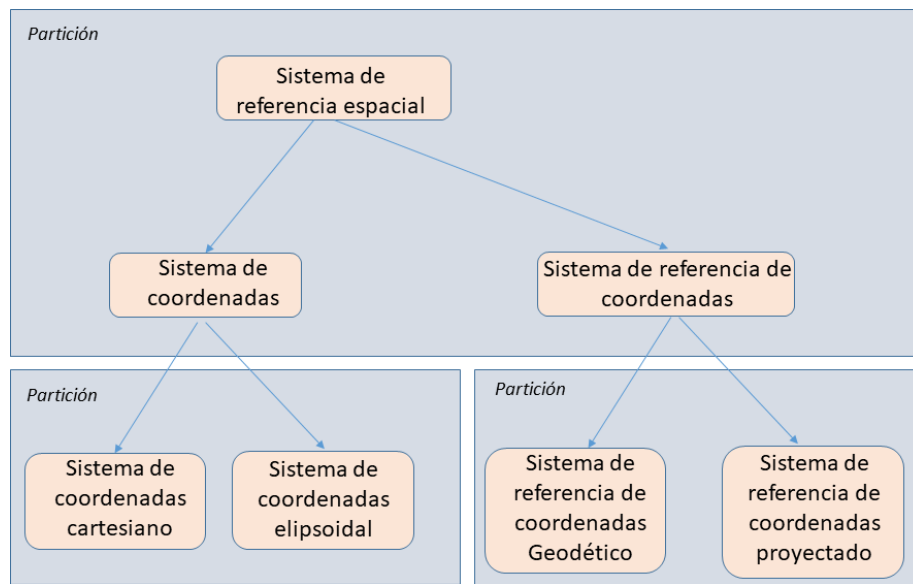


Figura 51 Relación taxonómica de Partición entre el concepto Sistema de Referencia Espacial y sus subconceptos.

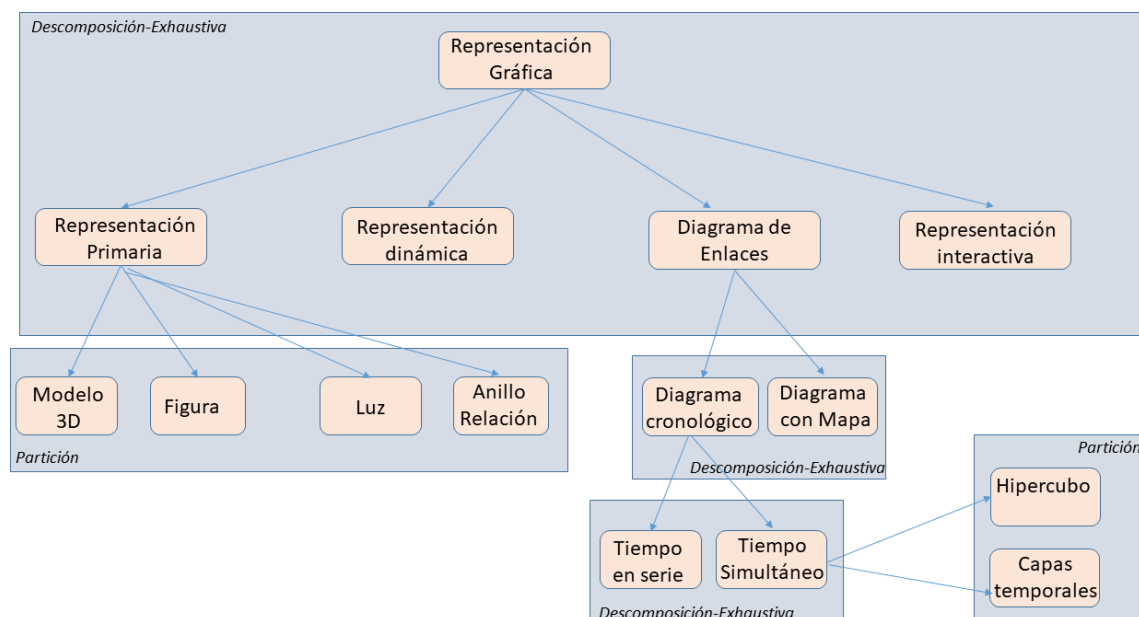


Figura 52 Relación taxonómica del concepto Representación gráfica y sus subconceptos.

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

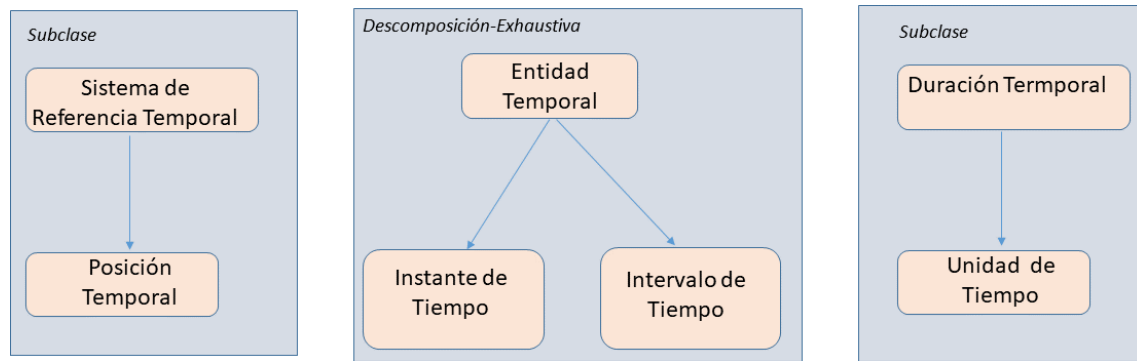


Figura 53 Relación taxonómica de los conceptos Sistema de Referencia Temporal, Duración Temporal, Entidad Temporal y sus subconceptos.

Tarea 3. Construir diagramas de relaciones binarias ad hoc.

En esta tarea, una vez definida la taxonomía de los conceptos, el objetivo es establecer las relaciones entre conceptos de una taxonomía, o de distintas taxonomías de conceptos.

En la Figura 54 se muestran las relaciones binarias ad hoc entre los conceptos Concepto Visual y Propiedad Visual. Como se ha discutido anteriormente, un Concepto Visual puede tener varias propiedades visuales, pero hay un conjunto de propiedades espaciales y temporales que también debe de tener. Además, en el diagrama se muestran las relaciones que muestran que un Concepto Visual representa a un concepto del dominio de datos, y una Propiedad Visual puede obtener su valor, mediante una consulta expresada en un lenguaje formal.

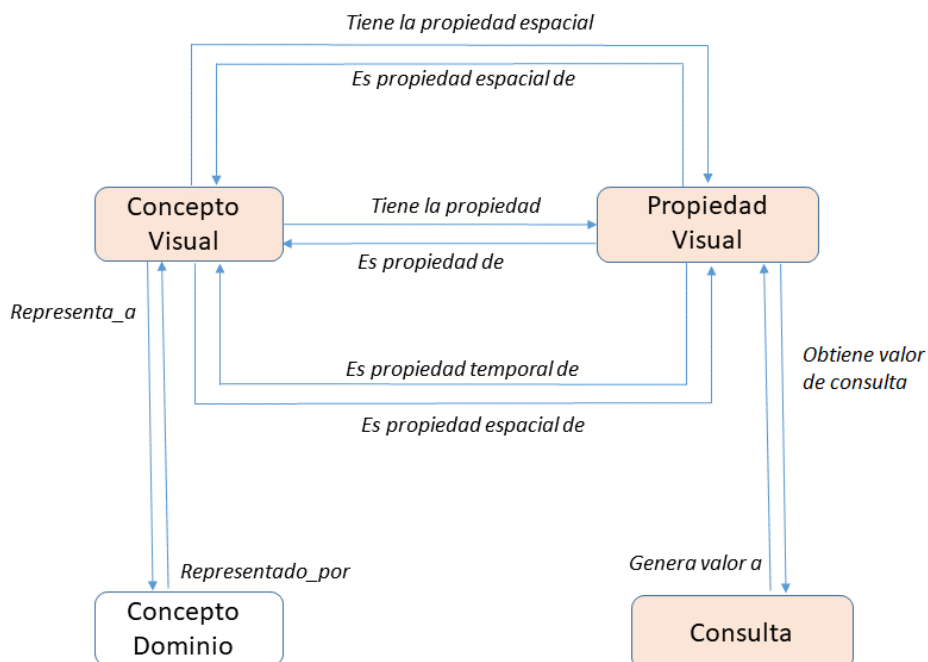


Figura 54 Relaciones binarias ad hoc entre los conceptos Concepto Visual y Propiedad Visual, así como con otros conceptos utilizados para asociar estos conceptos a la ontología de dominio.

En la Figura 55 se muestran las relaciones binarias de Entorno Temporal y Entorno Espacial. Entorno Temporal tiene relaciones con Intervalo Temporal, definido por dos Instantes de

Tiempo, y con Sistema de Referencia Temporal, para determinar el Calendario de referencia con el que se registrarán las instancias de estos conceptos.

Entorno Espacial tiene relaciones con Intervalo Espacial, definido por dos Puntos Espaciales, y con Sistema de Referencia Espacial, que determina el Sistema de Coordenadas, o Sistema de Referencia de Coordenadas, necesario para referenciar de forma adecuada las coordenadas espaciales.

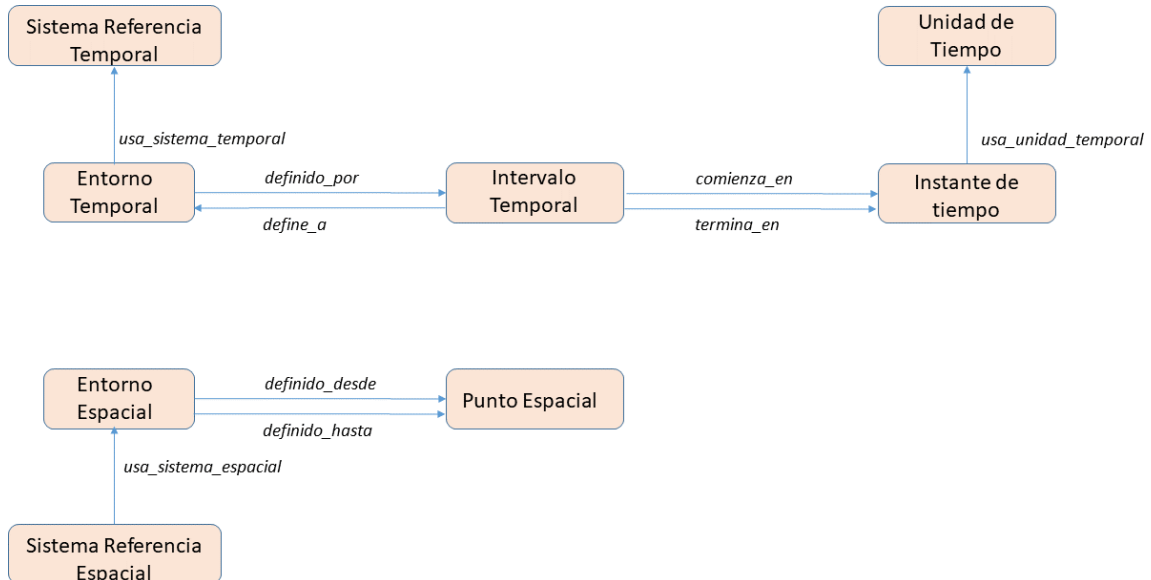


Figura 55 Relaciones binarias ad hoc de los conceptos Entorno Temporal y Entorno Espacial con los conceptos que los definen y sus respectivos sistemas de referencia.

En la Figura 56, se muestra un diagrama con las relaciones binarias del concepto Escena. Este concepto tiene una relación con Concepto Visual, que indica que las instancias del dominio asociadas a este Concepto Visual serán representadas en la escena. También tiene una relación con los subconceptos de Entorno: Entorno Temporal y Entorno Espacial, con el fin de definir el entorno espacio temporal de la escena. Por último, Escena tiene varias relaciones con el concepto Comportamiento, usadas para definir el comportamiento de la escena frente a la agrupación, visualización de relaciones, tratamiento de la incerteza y la interacción para controlar el tiempo. El Concepto Visual también tiene que tener una relación con los conceptos Entorno Temporal y Entorno Espacial. En el dominio de datos, los datos tendrán asociados un marco espacial y temporal, que se asocian a este concepto. De este modo, el sistema podrá conocer los marcos en el dominio de datos, referenciados por el Concepto Visual, y en el sistema de visualización, referenciados por la Escena.

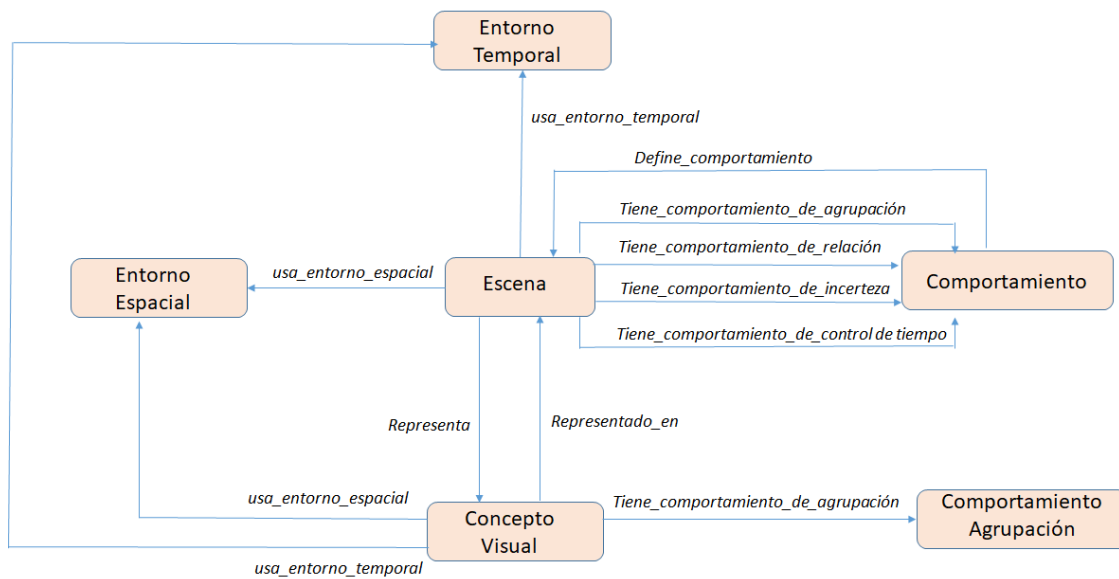


Figura 56 Relaciones binarias ad hoc del concepto de Escena con Comportamiento, Concepto Visual y los subconceptos de Entorno

La Figura 57 es un esquema con las relaciones entre el concepto “Concepto Visual” y el concepto “Marcador”. Estas relaciones indican como se van a representar gráficamente:

- Las instancias asociadas a ese concepto visual, a través de la relación “usa\_marcador de instancias”.
- Las agrupaciones de las instancias que estén en una posición especial cercana, a través de la relación “usa\_marcador\_representante”.
- Las instancias con incerteza, mediante la relación “usa\_marcador\_de\_incerteza”.
- Las instancias con incerteza, en el caso de que se use replicación en el tratamiento de la incerteza. Para ello se usa la relación “usa\_marcador\_de\_replicación”.

También aparecen en el diagrama las relaciones entre el concepto “Marcador” y el concepto “Representación Gráfica” que es el concepto encargado de representar los diferentes objetos gráficos.

Por último, se muestran las relaciones del concepto “Marcador” con el concepto “Modificador”, que también está relacionado con el concepto “Propiedad Visual”. Modificador representa el proceso de modificación de la instancia del concepto “Representación Gráfica” en función del valor de la Propiedad Visual en cada Instancia Visual que se va a visualizar.

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

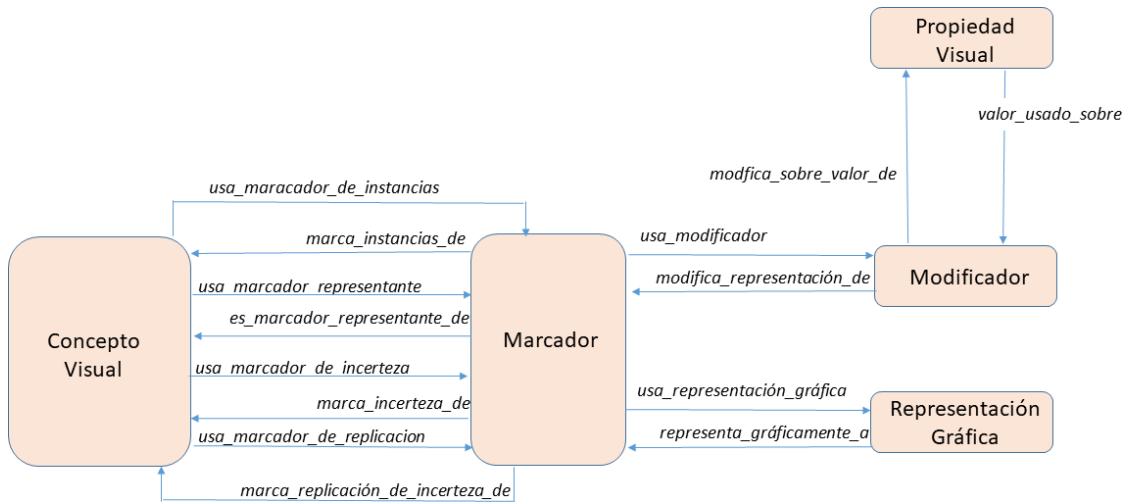


Figura 57 Relaciones binarias ad hoc del concepto de Concepto Visual con Marcador, que a su vez está relacionado con Representación gráfica.

Tarea 4: Construir el diccionario de conceptos.

El objetivo de esta tarea es construir un diccionario, donde para cada concepto, hay que incluir sus relaciones, instancias y propiedades de datos.

En la Tabla 20, Tabla 21, Tabla 22 y Tabla 23, se detalla el diccionario de conceptos.

Tabla 20 Diccionario de conceptos para los conceptos Concepto\_Visual, Propiedad\_Visual y Consulta.

| Concepto        | Propiedades Datos     | Relaciones   |
|-----------------|-----------------------|--|
| Concepto_Visual | representa_a<br>label | tiene_propiedad<br>tiene_propiedad_espacial<br>tiene_propiedad_temporal<br>usa_entorno_espacial<br>usa_entorno_temporal<br>representado_en<br>usa_marcador_instancias<br>usa_marcador_representante<br>usa_marcador_incerteza<br>usa_marcador_replicación<br>tiene_comportamiento_agrupación |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                  |  |   |
|------------------|--|---|
| Propiedad_Visual | Label<br>es_filtrable<br>es_relacionable<br>es_visible<br>variable_en_consulta | Es propiedad de<br>Es_propiedad_espacial_de<br>Es propiedad_temporal_de<br>Obtiene_valor_de<br>Valor_usado_sobre<br>tiene_color |
| Consulta         | expresión  | Genera_valor_a  |
|                  | lenguaje   |   |
|                  | usa_variable_de_instancia  |   |

En el esquema de la Figura 54 el concepto “Concepto de Dominio” está resaltado. Esto es así, porque dicho concepto pertenece al dominio de datos. Un dominio que en principio no se ha de modificar para poder visualizarse, además de que va a variar con cada visualización. Por lo tanto, en la práctica, estas relaciones no estarán, como tales, no existirá ninguna relación entre Concepto Visual y Concepto Dominio. Para mantener la información, se creará una propiedad de datos “representa\_a”, donde el dominio será “Concepto Visual” y el rango será una URI. Cuando se creen instancias de este concepto, la propiedad tendrá el valor de la URI del concepto de dominio que se desee representar. Tal y como se especificó en la definición del modelo, se usará una consulta para acceder al valor de una Propiedad Visual de cada Instancia Visual. Para vincular la consulta al concepto del dominio de datos, se usará una etiqueta, que el sistema de visualización tendrá que intercambiar por la URI de la instancia del concepto de dominio. El mismo proceso habrá que realizar para asociar el valor del dato devuelto por la consulta con el de la propiedad. Para ello, el concepto “Consulta” tendrá la propiedad de datos “usa\_variable\_de\_instancia”, que indicará la variable usada en la consulta para referenciar a la instancia del concepto de dominio de datos representado por la instancia del Concepto Visual.

Del mismo modo, en la expresión de la consulta, también aparece una variable que hace referencia a la propiedad, cuyo valor se va a obtener. El nombre de esta variable debe de ser una propiedad de datos del concepto “Propiedad Visual”.

En la Figura 58 se representa un esquema en el que una instancia de Concepto Visual (en verde), que representa al concepto E22\_Man-Made\_Object del modelo de CIDOC-CRM, tiene dos instancias del concepto “Propiedad Visual”. Una (azul) representa la propiedad material, y la otra (naranja) la propiedad técnica, del textil. La primera tiene la variable “materialLabel” y la segunda “tecnicLabel” que aparecen en la expresión de la consulta (violeta). Para referirse a la URI de cada instancia del concepto E22\_Man-Made\_Object, la instancia de consulta usa la variable “this”. De este modo, para obtener los valores de la URI, por ejemplo, de este objeto que el grafo de conocimiento del proyecto SILKNOW corresponde a una casulla (prenda eclesiástica):

<<http://data.silknow.org/object/b80ad914-4cbe-3b27-ac64-4ad7e0a375b9>>



## CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Solo habría que ejecutar la consulta, sustituyendo “this” por la URI, quedando la consulta de la siguiente forma:

```
SELECT distinct ?techniqueLabel ?materialLabel
WHERE {
  ?s
  rdf:type
  <http://erlangen-crm.org/current/E12_Production> .
  ?s
  <http://erlangen-crm.org/current/P108_has_produced>
  <http://data.silknow.org/object/b80ad914-4cbe-3b27-ac64-4ad7e0a375b9> .
  ?s
  <http://erlangen-crm.org/current/P32_used_general_technique>
  ?p .
  ?p
  <http://www.w3.org/2004/02/skos/core#prefLabel>
  ?techniqueLabel .
  ?s
  <http://erlangen-crm.org/current/P126_employed>
  ?q .
  ?q
  <http://www.w3.org/2004/02/skos/core#prefLabel>
  ?materialLabel
}
```

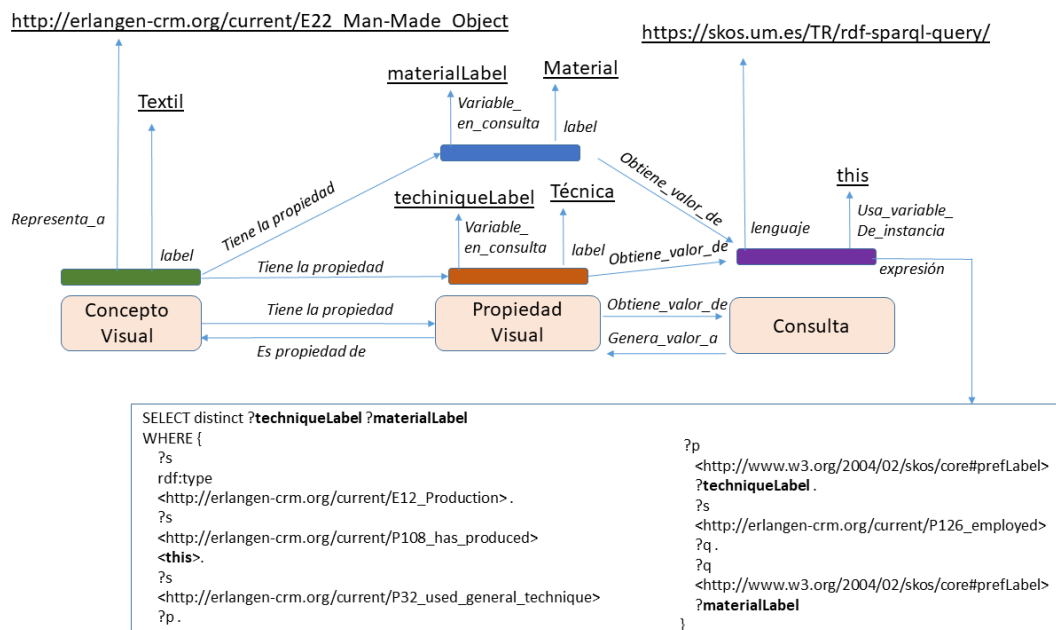


Figura 58 Esquema de generación de valores para dos propiedades visuales de un concepto visual, sobre la misma consulta.

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tabla 21 Diccionario de conceptos para los conceptos Marcador, Modificador y Representación\_Gráfica.

| Concepto               | Propiedades Datos  | Relaciones  | Instancias |
|------------------------|--|---|------------|
| Marcador               |  | Marca_instancias_de<br>Es_marcador_representante_de<br>Marca_incerteza_de<br>Marca_replicación_de_incerteza_de<br>Usa_modificador<br>Usa_representación_gráfica |            |
| Modificador            | Valor_propiedad_es_número<br>Valor_propiedad_es_string<br>Valor_propiedad_es_uri<br>Valor_propiedad_es_bool<br>Valor_propiedad_es_fecha              | Modifica_representación_de<br>Modifica_sobre_valor_de<br>Modifica_al_color<br>Modifica_a_imagen   |            |
| Representación Gráfica | <p><b><i>Este concepto es equivalente a al concepto</i></b></p> <p><b><i>Primary_Graphic_Representation de la ontología GRAPHICS de VISO</i></b></p> |   |            |

En la Tabla 21 se expone el diccionario de conceptos para los conceptos Marcador, Modificador y Representación\_Gráfica. Las propiedades de datos del concepto Modificador están divididas en dos grupos:

- Las propiedades usadas para comparar el valor de la propiedad y compararla con un valor concreto, de un tipo determinado, y en caso de ser el mismo valor, aplicar las modificaciones adecuadas. Estas propiedades siguen el esquema: Valor\_propiedad\_<tipo>.
- Las propiedades de aplicación de modificación, que siguen el esquema: Modifica\_a\_<color/imagen>

Dada la analogía entre el concepto Representación\_Gráfica del modelo de esta investigación y el concepto Primary\_Graphic\_Representation de la ontología GRAPHICS, con las extensiones discutidas en este capítulo, se reusará este concepto de la ontología GRAPHICS de VISO como equivalente.

En la Tabla 22 se muestra el diccionario de conceptos para Escena, Entorno Espacial, Sistema de Coordenadas, Sistema de Referencia de Coordenadas, Punto Espacial, Entorno Temporal, Unidad Temporal, Sistema de Referencia Temporal, e Instante de Tiempo.

**CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL**

Tras la discusión realizada en el apartado Referenciación Espacial, de este capítulo, se observa una gran similitud semántica entre los conceptos Sistema de Coordenadas, Sistema de Referencia de Coordenadas y Punto Espacial del modelo propuesto en el capítulo anterior y los conceptos Coordinate System, Coordinate Reference System y Point, de la ontología Institut National de l'Information Géographique et Forestière (IGF) (Abadie, Ateamezing 2015) . Debido a ello, se reutilizarán estos conceptos de dicha ontología, como equivalentes.

Igualmente, en función de lo expuesto en el apartado de Referenciación Temporal, de este capítulo. Se aprecia la equivalencia entre los conceptos Unidad Temporal, Sistema de Referencia Temporal e Instante de Tiempo, con los conceptos Temporal Unit, Time Reference System e Instant de la ontología TIME de W3C.

*Tabla 22 Diccionario de conceptos para los conceptos Escena, Entorno Espacial, Sistema Referencia Espacial, Sistema de Coordenadas, Sistema de Referencia de Coordenadas, Punto Espacial, Entorno Temporal, Unidad Temporal, Sistema de Referencia Temporal, e Instante de Tiempo.*

| Concepto                             | Propiedades Datos  | Relaciones  | Instancias |
|--------------------------------------|--|---|------------|
| Escena                               | Nivel_de_detalle<br>Máximo_nivel_zoom<br>Mínimo_nivel_zoom   | Usa_representación_gráfica<br>Representa<br>Usa_entorno_temporal<br>Usa_entorno_espacial<br>Tiene_comportamiento_agrupación<br>Tiene_comportamiento_relación<br>Tiene_comportamiento_incerteza<br>Tiene_comportamiento_control_tiempo |            |
| Entorno Espacial                     |  | Definido_desde<br>Definido_hasta  |            |
| Sistema Referencia Espacial          |  | Usa_sistema_espacial  |            |
| Sistema de Coordenadas               | <p><b><i>Este concepto es equivalente a al concepto</i></b><br/> <b><i>Coordinate system de la ontología del IGF</i></b></p>                 |   |            |
| Sistema de Referencia de Coordenadas | <p><b><i>Este concepto es equivalente a al concepto</i></b><br/> <b><i>Coordinate reference system (CRS) de la ontología del IGF</i></b></p> |   |            |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                                |   |  |  |
|--------------------------------|---|--|--|
| Punto Espacial                 | <p><b><i>Este concepto es equivalente a al concepto</i></b></p> <p><b><i>Point de la ontología del IGF</i></b></p>                  |  |  |
| Entorno temporal               |   | Comienza_en<br>Termina_en<br>Usa_Sistema_temporal<br>Usa_Unidad_Temporal |  |
| Unidad_Temporal                | <p><b><i>Este concepto es equivalente a al concepto</i></b></p> <p><b><i>TemporalUnit de la ontología TIME</i></b></p>              |  |  |
| Sistema de Referencia Temporal | <p><b><i>Este concepto es equivalente a al concepto</i></b></p> <p><b><i>TimeReferenceSystem (TRS) de la ontología TIME</i></b></p> |  |  |
| Instante de Tiempo             | <p><b><i>Este concepto es equivalente a al concepto</i></b></p> <p><b><i>Instant de la ontología TIME</i></b></p>                   |  |  |

En la Tabla 23 se describe el diccionario de conceptos del Concepto Comportamiento y sus extensiones. El concepto comportamiento tiene una relación inversa, con el concepto Escena, sobre el que define sus comportamientos a varios niveles (agrupamiento, relación, etc.), y también para definir el comportamiento de agrupamiento del Concepto Visual, en el caso de que tenga un comportamiento de agrupamiento diferente al genérico de la escena.

Se incluyen propiedades en el comportamiento de Agrupamiento por Clusters, para indicar si se tiene que dibujar sobre la escena el área de influencia de cada grupo, y si se produce animación, o no, durante las transiciones. En el caso del Agrupamiento Mapa de Calor, se define el número de niveles de color, y si hay un color base, o se sigue el mapa de color básico de temperatura de color.

En el concepto Relación de Enlaces se incluyen propiedades para definir el número máximo de enlaces visibles en escena, si se aplica la técnica de dibujar líneas, o arcos, o se aplica la técnica de Color.

En el concepto Relación\_Anillo, se incluye una propiedad para definir el número máximo de propiedades que se van a contemplar en este tipo de marcador de relaciones.

En el concepto Control de Tiempo, presenta una propiedad para indicar si se utilizan pasos para referenciar el paso del tiempo, además de indicar la frecuencia de estos pasos, con una relación con una instancia del concepto Unidad Temporal (usa\_frecuencia)

**CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL**

En el concepto Control\_Simultáneo, se incluyen propiedades para definir el número máximo de pasos, y si en cada paso tiene que visualizarse la zona espacial sobre la que se aplica el filtro temporal.

El concepto Tratamiento de Incerteza contempla la propiedad Aplica\_decoloración, que indica si se usa decoloración para marcar la incerteza, además de permitir definir un marcador gráfico especial para tal fin (Usa\_marcador\_incerteza).

Por último, sobre Incerteza\_con\_réplica, se presenta la propiedad Aplica\_desplazamiento, que da la opción de aplicar un desplazamiento sobre el marcador gráfico, de las réplicas, para indicar que efectivamente son réplicas.

*Tabla 23 Diccionario de conceptos para los conceptos de Comportamiento y sus extensiones.*

| Concepto                | Propiedades Datos  | Relaciones               | Instancias |
|-------------------------|--|--------------------------|------------|
| Comportamiento          |  | define_comportamiento_de |            |
| Agrupamiento Clusters   | dibujar_influencia<br>animación_activa                                       | marcador_representante   |            |
| Agrupamiento Mapa Calor | numero_niveles   | Tiene_color_base         |            |
| Relación_Enlaces        | número_maximo_enlaces<br>Aplica_Técnica_de_líneas<br>Aplica_Técnica_de_Color |                          |            |
| Relación_Anillo         | Numero_maximo_propiedades  |                          |            |
| Control de Tiempo       | Usa_pasos  | Usa_frecuencia           |            |
| Control Simultáneo      | Número_máximo_de_pasos<br>Visualiza_zona_espacial                            |                          |            |
| Control Individual      |  |                          |            |
| Tratamiento Incerteza   | Aplica_decoloración  | Usa_marcador_incerteza   |            |
| Incerteza con Réplica   | Aplica_desplazamiento  |                          |            |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tarea 5: Describir las relaciones binarias ad hoc.

La finalidad de esta tarea es describir en detalle las relaciones binarias ad hoc identificadas hasta el momento.

En la Tabla 24 se muestra esta información, indicando para cada relación, su nombre, el concepto origen, la máxima cardinalidad, el concepto destino y la relación inversa (si existe).

Tabla 24 Tabla de relaciones binarias ad hoc.

| Nombre                          | Concepto Origen  | Card.<br>Min,<br>Máxima | Concepto Destino          | Relación Inversa               |
|---------------------------------|------------------|-------------------------|---------------------------|--------------------------------|
| Tiene propiedad                 | Concepto Visual  | 2, N                    | Propiedad Visual          | Es propiedad de                |
| Tiene propiedad espacial        | Concepto Visual  | 1, 3                    | Propiedad Visual          | Es propiedad espacial de       |
| Tiene propiedad temporal        | Concepto Visual  | 1, 2                    | Propiedad Visual          | Es propiedad temporal de       |
| Usa entorno espacial            | Concepto Visual  | 1,1                     | Entorno Espacial          | Es entorno espacial de         |
| Usa entorno temporal            | Concepto Visual  | 1,1                     | Entorno Temporal          | Es entorno temporal de         |
| Representado en                 | Concepto Visual  | 1,N                     | Escena                    | Representa                     |
| Usa marcador instancias         | Concepto Visual  | 1,N                     | Marcador                  | Marca instancias de            |
| Usa marcador representante      | Concepto Visual  | 0,N                     | Marcador                  | Es representante de            |
| Usa marcador incerteza          | Concepto Visual  | 0,N                     | Marcador                  | Marca incerteza de             |
| Usa marcador replicación        | Concepto Visual  | 0,N                     | Marcador                  | Marca replicación incerteza de |
| Tiene Comportamiento Agrupación | Concepto Visual  | 0,1                     | Comportamiento Agrupación |                                |
| Obtiene valor de                | Propiedad Visual | 0,1                     | Consulta                  | Genera valor a                 |
| Valor usado sobre               | Propiedad Visual | 0, 1                    | Modificador               | Modifica sobre valor de        |
| Tiene color                     | Propiedad Visual | 0, 1                    | Color                     |                                |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                                     |                  |      |                               |   |
|-------------------------------------|------------------|------|-------------------------------|---|
| Usa modificador                     | Marcador         | 0, 1 | Modificador                   | Modifica representación de                |
| Usa representación gráfica          | Marcador         | 1,1  | Representación gráfica        |   |
| Usado en Escena                     | Marcador         | 1,1  | Escena                        | Se aplica marcador                        |
| Usa representación gráfica          | Escena           | 0,N  | Representación gráfica        |   |
| Usa Entorno Temporal                | Escena           | 1,1  | Entorno Temporal              | Es entorno temporal de                    |
| Usa Entorno Espacial                | Escena           | 1,1  | Entorno Espacial              | Es entorno espacial de                    |
| Tiene Comportamiento Agrupación     | Escena           | 0,1  | Comportamiento Agrupación     | (Comportamiento)<br>Define comportamiento |
| Tiene Comportamiento Relación       | Escena           | 0,2  | Comportamiento Relación       | (Comportamiento)<br>Define comportamiento |
| Tiene Comportamiento Incerteza      | Escena           | 0,1  | Comportamiento Incerteza      | (Comportamiento)<br>Define comportamiento |
| Tiene Comportamiento Control Tiempo | Escena           | 0,2  | Comportamiento Control Tiempo | (Comportamiento)<br>Define comportamiento |
| Definido desde                      | Entorno Espacial | 0,1  | Punto Espacial                |   |
| Definido Hasta                      | Entorno Espacial | 0,1  | Punto Espacial                |   |
| Usa Sistema Espacial                | Entorno Espacial | 1,1  | Sistema Referencia Espacial   |   |
| Comienza en                         | Entorno Temporal | 1,1  | Instante de Tiempo            |   |
| Termina en                          | Entorno Temporal | 1,1  | Instante de Tiempo            |   |



CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                        |                               |     |                                |  |
|------------------------|-------------------------------|-----|--------------------------------|--|
| Usa Sistema Temporal   | Entorno Temporal              | 1,1 | Sistema de Referencia Temporal |  |
| Usa Unidad Temporal    | Entorno Temporal              | 1,1 | Unidad Temporal                |  |
| Marcador representante | Agrupamiento Clusters         | 1,1 | Marcador                       |  |
| Usa frecuencia         | Comportamiento Control Tiempo | 1,1 | Unidad Temporal                |  |
| Usa marcador incerteza | Tratamiento Incerteza         | 0,1 | Marcador                       |  |
| Modifica_al_color      | Modificador                   | 0,1 | Color                          |  |
| Modifica_a_imagen      | Modificador                   | 0,1 | Picture                        |  |
| Tiene_color_base       | Agrupamiento Mapa de Calor    | 0,1 | Color                          |  |

Tareas 6 y 7: Describir propiedades de datos.

El objetivo de esta tarea es describir con detalle las propiedades de los conceptos incluidos en el diccionario de conceptos.

En la Tabla 25, se enumera para cada concepto, el nombre de la propiedad, el tipo de valor definido por XML Schema Definition Language (XSD) (W3C 2012) , el rango de valores y la cardinalidad.

Tabla 25 Propiedades de datos.

| Concepto         | Propiedad              | Tipo             | Rango         | Cardinalidad |
|------------------|------------------------|------------------|---------------|--------------|
| Concepto Visual  | Representa a           | xsd:anyURI       |               | (1,1)        |
| Propiedad Visual | Es filtrable           | xsd:boolean      | {true, false} | (1,1)        |
|                  | Es relacionable        | xsd:boolean      | {true, false} | (1,1)        |
|                  | Es visible             | xsd:boolean      | {true, false} | (1,1)        |
|                  | Variable en consulta   | xsd:string       |               | (1,1)        |
| Consulta         | Expresión              | xsd:string       |               | (1,1)        |
|                  | Lenguaje               | xsd:string (URL) |               | (1,1)        |
|                  | Usa variable instancia | xsd:string       |               | (1,1)        |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                         |                            |             |               |       |
|-------------------------|----------------------------|-------------|---------------|-------|
| Modificador             | Valor propiedad es número  | xsd:float   |               | (1,1) |
|                         | Valor propiedad es string  | xsd:string  |               | (1,1) |
|                         | Valor propiedad es uri     | xsd:anyURI  |               | (1,1) |
|                         | Valor propiedad es boolean | xsd:boolean | {true, false} | (1,1) |
|                         | Valor propiedad es fecha   | xsd:date    |               | (1,1) |
| Escena                  | Nivel detalle              | xsd:integer |               | (1,1) |
|                         | Máximo nivel zoom          | xsd:integer |               | (1,1) |
|                         | Mínimo nivel zoom          | xsd:integer |               | (1,1) |
| Agrupamiento Clusters   | Dibujar influencia         | xsd:boolean | {true, false} | (1,1) |
|                         | Animación activa           | xsd:boolean | {true, false} | (1,1) |
| Agrupamiento Mapa Calor | Número niveles             | xsd:integer |               | (1,1) |
| Relación Enlaces        | Número máximo enlaces      | xsd:integer |               | (1,1) |
|                         | Aplica Técnica Líneas      | xsd:boolean | {true, false} | (1,1) |
|                         | Aplica Técnica Color       | xsd:boolean | {true, false} | (1,1) |
| Relación Anillo         | Número máximo propiedades  | xsd:integer |               | (1,1) |
| Control de Tiempo       | Usa pasos                  | xsd:boolean | {true, false} | (1,1) |
| Control Simultáneo      | Número máximo pasos        | xsd:integer |               | (1,1) |
|                         | Visualiza zona espacial    | xsd:boolean | {true, false} | (1,1) |
| Tratamiento incerteza   | Aplica_decoloración        | xsd:boolean | {true, false} | (1,1) |
| Incerteza con réplica   | Aplica_desplazamiento      | xsd:boolean | {true, false} | (1,1) |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

Tarea 8: Describir constantes.

En la ontología que se está diseñando en este trabajo de investigación no se aprecian constantes de relevante importancia para la definición del modelo.

En el contexto de este modelo, más que constantes, lo que se requerirían son instancias, que serán debidamente expuestas en la tarea 11 de METHONTOLOGY.

Tarea 9: Definir axiomas formales.

Al tratarse de un modelo de representación gráfica, hay muchas posibilidades de representación de la información, pero hay muy pocos axiomas que se puedan concluir de la descripción del modelo.

La mayoría de limitaciones del modelo vienen determinadas por la cardinalidad máxima y mínima de las relaciones. Sin embargo, hay un axioma que es básico, y es que al tratarse de representación de datos espacio temporales, tanto la Escena, como los Conceptos Visuales que se representan han de tener un Entorno Temporal y un Entorno Espacial relacionado. De otra forma, no se podría interpretar debidamente la información fundamental del modelo, que es el espacio y el tiempo. En la Tabla 26 se detalla la reducida lista de axiomas que tiene la ontología.

Tabla 26 Axiomas de la ontología STEVO.

| Nombre axioma   | Descripción   | Expresión   | Conceptos  | Relaciones                                       | Variables |
|---|---|---|--|--|-----------|
| La visualización se da en Entorno Temporal y Espacial | Toda Escena ha de tener una relación con un Entorno Espacial y Temporal | no (existe(?E) ,<br>Escena(?E) y<br>no (existe(?ET, ?EE),<br>Entorno Temporal(?ET)<br>y<br>Entorno Espacial(?EE)<br>y<br>Usa Entorno Temporal(E, ET) y<br>Usa Entorno Espacial(E, EE))) | Escena<br><br>Entorno Temporal<br><br>Entorno Espacial | Usa Sistema Espacial<br><br>Usa Sistema Temporal | E, ET, EE |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|  |   |  |  |  |  |
|--|---|--|--|--|--|
| Un concepto visual ha de estar relacionado con un entorno temporal y espacial. | Todo concepto visual ha de tener una relación con un entorno Espacial y con un Entorno Temporal | no (existe(?C) ,<br>Concepto Visual(?C) y<br>no (existe(?ET, ?EE),<br>Entorno Temporal(?ET)<br>y<br>Entorno Espacial(?EE)<br>y<br>Usa Entorno Temporal(C, ET) y<br>Usa Entorno Espacial(C, EE))) |  |  |  |
|--|---|--|--|--|--|

Tarea 10: Definir reglas.

Aunque el modelo de representación permite que haya muchas propiedades y relaciones que sean opcionales, hay una serie de reglas que han de ser cumplidas para verificar la integridad del modelo. En la Tabla 27 se exponen estas reglas.

Tabla 27 Reglas de la ontología STEVO.

| Nombre  | Descripción   | Expresión   |
|---|---|---|
| Representación requiere datos espaciales y temporales   | Si un concepto visual se representa en un escena, este tiene al menos una propiedad espacial y una propiedad temporal | Si (Escena(?E) y Concepto Visual(?C) y Representa(E,C))<br>entonces<br>Existe(?PT,?PE) y P. Visual(?PT) y P. Visual(?PE) y<br>Tiene_propiedad_espacial(C,PE) y<br>Tiene_propiedad_temporal(C, PT) |
| Representación requiere un concepto de dominio de datos | Si un concepto visual se representa en escena, este representa un concepto del dominio de datos                       | Si (Escena(?E) y Concepto Visual(?C) y Representa(E,C))<br>entonces<br>Existe(?c) y xsd:anyURI( c ) y Representa_a(C, c)  |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|   |  |  |
|---|--|--|
| Todo concepto visual ha de tener un marcador gráfico.                     | Si un concepto visual se representa en una escena, ha de tener asociado un marcador que sea usado en dicha escena. | Si (Escena(?E) y Concepto Visual(?C) y Representa(E,C)) entonces<br>(Existe(?M), Marcador(?M) , y<br>Usa Marcador Instancias(C,M) y<br>Usado en Escena(E, M))  |
| Una propiedad relacionable tiene que tener un comportamiento de relación. | Si la escena no tiene asociado un comportamiento de relación, no puede tener propiedades relacionables.            | Si ( Existe(?E, ?C, ?PV),<br>Escena(?E) y Concepto Visual(?C) y Representa(E,C), y<br>Propiedad Visual(?PV) y Tiene Propiedad(C, PV ) y<br>Es filtrable(PV)<br>entonces<br>Existe(?CR),<br>Comportamiento Relación (?CR) y<br>Tiene Comportamiento Relación(E, CR) |

Tarea 11: Definir instancias.

A continuación, en la Tabla 28, se muestra una lista de instancias de la ontología STEVO que serán usadas por un gran número de visualizaciones, y por ello serán inicialmente incluidas en la ontología.

Tabla 28 Instancias de la ontología STEVO

| Instancia                   | Concepto                             | Atributos | Valor     | Relación  |
|-----------------------------|--------------------------------------|-----------|-----------|---|
| Circulo                     | Shape                                | label     | circle    |   |
| Cuadrado                    | Shape                                | label     | square    |   |
| Rectángulo                  | Shape                                | label     | rectangle |   |
| Entorno Espacial WGS84      | Entorno Espacial                     | Label     | WGS84     | Usa sistema de referencia (Sistema de referencia WGS84) |
| Sistema de referencia WGS84 | Geodetic coordinate reference system | label     | WGS84     | uses datum (Datum WGS84)                                |

CAPÍTULO 7. DISEÑO E IMPLEMENTACIÓN DE UNA ONTOLOGÍA QUE REPRESENTA UN MODELO DE VISUALIZACIÓN DEL CONTENIDO DE GRAFOS DE CONOCIMIENTO DE PATRIMONIO CULTURAL

|                                   |                  |              |                          |  |
|-----------------------------------|------------------|--------------|--------------------------|--|
| Datum WGS84                       | Geodetic Datum   | label        | WGS84                    |  |
| Entorno Espacial<br>Cartesiano    | Entorno Espacial | label        | Cartesiano               | Usa sistema de referencia (Sistema de coordenadas cartesiano)            |
| Sistema de coordenadas cartesiano | CartesianCS      | label        | Cartesiano               |  |
| Siglo                             | Temporal Unit    |              |                          | Same as (time# unityCentury)   |
| Año                               | Temporal Unit    |              |                          | Same as (time# unitYear)   |
| 10000 ac                          | Instant          | En_fecha_xsd | -10000-01-01T00:00:00    |  |
| 2100                              | Instant          | En_fecha_xsd | 2100-01-01T00:00:00      |  |
| Historia humanidad                | Entorno Temporal | label        | Historia de la humanidad | Comienza en (10000 ac)<br>Termina en (2100)<br>Usa unidad temporal (Año) |
| Azul                              | Color            | Has_color    | "#0096FF"                |  |
| Rojo                              | Color            | Has_color    | "#FF0000"                |  |
| Verde                             | Color            | Has_color    | "#008000"                |  |
| Amarillo                          | Color            | Has_color    | "#FF00FF"                |  |
| Violeta                           | Color            | Has_color    | "#FFFF00"                |  |
| Negro                             | Color            | Has_color    | "#000000"                |  |
| Blanco                            | Color            | Has_color    | "#FFFFFF"                |  |

### 7.3. IMPLEMENTACIÓN

Para implementar la ontología STEVO se ha usado la herramienta Protégé (Stanford University 2022), y se ha publicado una primera versión en el repositorio Github (Sevilla, J. 2022).

El formato elegido ha sido OWL. La primera versión de esta ontología se ha desarrollado en inglés. Hacer que esta ontología sea multilingüe es posible mediante la aplicación de varias técnicas. La técnica más directa es la de escribir la ontología en otra lengua, con la misma estructura y semántica que la ontología original y hacer uso del operador de equivalencia entre términos y relaciones.

Otra alternativa, es usar atributos XML:lang en los comentarios de la ontología. De esta forma se puede describir cada concepto de forma natural en cualquier lengua. Sin embargo, esta alternativa no es accesible para algunos razonadores, como RACER(Haarslev, Möller 2003) , debido a su limitación de lenguaje de requerimientos para obtener el valor de campos como rdf:comment en la descripción del concepto.

En la Figura 57, se muestra una captura de pantalla de la herramienta Protégé, con el resultado de la inferencia tras aplicar el razonador HermiT (Glimm, Horrocks et al. 2014).

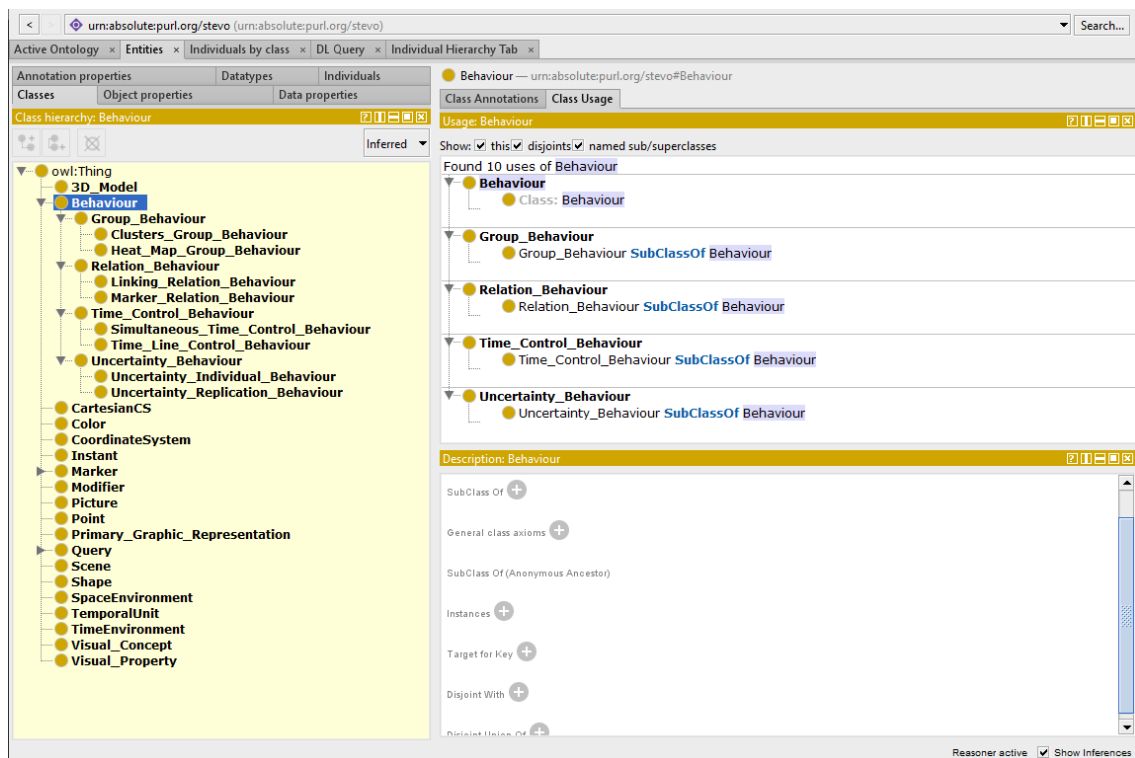


Figura 59 Captura de pantalla de la aplicación Protege, con la inferencia de la ontología STEVO, tras aplicar el razonamiento HermiT.

#### 7.4. CONCLUSIONES

En este capítulo se ha analizado el modelo descrito en el capítulo 6 y se ha procedido a diseñar una ontología, que se ha denominado STEVO, que permita la representación del modelo descrito.

El objetivo del modelo era describir como visualizar de forma genérica, los datos espacio temporales de patrimonio cultural, soportados por un grafo de conocimiento.

Antes de proceder al diseño de la ontología se han estudiado varias ontologías cuyo contenido podría ser reutilizado y ser integrado con el diseño de STEVO.

Las ontologías analizadas han sido:

- VISO (Polowinski, Jan, Voigt 2013) y VUMO (Sobral, Thiago, Galvão et al. 2020). Referencias en la representación de visualización del contenido de grafos de conocimiento mediante ontologías.
- TIME (W3C 2020). Ontología del consorcio W3C para representar unidades y duraciones temporales.
- IGF. Ontología de sistemas de coordenadas desarrollada por Institut National de l'Information Géographique et Forestière (IGF) (Abadie, Atemezing 2015)

Con toda esta información, y aplicando la metodología METHONTOLOGY se ha diseñado e implementado una ontología que cumple los objetivos previstos en este capítulo. Varios conceptos y propiedades de las ontologías analizadas han sido reusados en STEVO. Todo el proceso de diseño ha sido debidamente expuesto en este capítulo.

El resultado ha sido una ontología, implementada en formato OWL, que puede analizarse de forma pública en el repositorio Github.

El idioma elegido para la ontología es el inglés, pero tal y como ha sido descrito en el apartado de implementación, realizar una versión en otro idioma es muy sencillo.



## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

### 8.1. RESUMEN

En este capítulo se describen varios procesos dirigidos al desarrollo de una solución software, que permita visualizar mediante una aplicación web, el contenido de un grafo de conocimiento con datos de patrimonio cultural. La especificación de dicha visualización ha de realizarse mediante la ontología STEVO, cuyo diseño e implementación ha sido descrito en el capítulo anterior.

Comienza con el diseño y la implementación de un marco de trabajo que soporte un componente software que pueda mostrarse en una aplicación web. Este componente, recibirá como parámetros, la especificación de una visualización, y las instancias a visualizar, en caso de que sea un conjunto determinado.

Por una parte, se expone el diseño de ese marco de trabajo, en el que ese componente software de visualización web, toma un papel fundamental, y también, el resto de elementos que conforman la aplicación web. Por otra parte, también se detalla el diseño y la implementación de dicho componente de visualización.

Al tratarse de una implementación, es necesario considerar las diferentes tecnologías que hacen posible alcanzar los resultados buscados. En el capítulo 5 de esta investigación se realiza dicho análisis, que se evaluará en este capítulo, con el fin de elegir la tecnología más adecuada para los fines de esta investigación.

### 8.2. MARCO DE TRABAJO

En este apartado se describe el diseño y desarrollo de un marco de trabajo, que permitirá visualizar el contenido de un grafo de conocimiento con datos de patrimonio cultural, mediante una aplicación web.

La especificación sobre cómo se han de visualizar dichos datos gráficamente vendrá determinada por el contenido de la ontología STEVO. En dicha ontología tienen que existir instancias que permitan definir:

- Los conceptos visuales, relacionado con el dominio de datos, que se van a visualizar, así como las diferentes propiedades.
- La escena, o escenas, en la que dichos objetos van a visualizarse. El tipo de escena define las posibilidades de interacción para navegar por ella.
- Las diferentes posibilidades de interacción con los datos a visualizar. Que propiedades van a visualizarse, si se podrán realizar filtrados sobre ellas, si se van a poder ver objetos relacionados, etc.
- También tienen que estar definidos los espacios temporales y espaciales, tanto de los datos del dominio de datos, como de la escena de visualización.
- Los diferentes comportamientos asociados a los datos a visualizar.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

Por lo tanto, por una parte, existirá un grafo de conocimiento y por otra parte una ontología con información sobre cómo visualizar el contenido de dicho grafo.

Además, tiene que existir una forma de indicar si se quiere visualizar parte del contenido del grafo, o todo, de forma dinámica, de manera que se pueda solicitar la visualización de diferentes contenidos.

El esquema básico del marco de trabajo está compuesto por una aplicación web, en la que varios usuarios, pueden decidir qué datos visualizar. Estos datos, se corresponderán con un conjunto de instancias de uno, o varios, conceptos del dominio de datos. El sistema accederá a ellos a través de una consulta en SPARQL, sobre el grafo de conocimiento. El sistema puede consultar a la ontología STEVO, con la información de visualización, para conocer posibles escenas de visualización. Y en función de la escena seleccionada, conocer cómo visualizar estos datos. La aplicación web, activará una herramienta de visualización, que tendrá acceso a los datos a visualizar, y cómo visualizarlos. Esta herramienta representará una escena interactiva, con los datos elegidos, permitiendo al usuario interactuar con ellos. De este modo, se representarán los datos con el agrupamiento adecuado, y el usuario podrá realizar filtros con esos datos, ver sus relaciones, así como aplicar los controles de tiempo que estén especificados en STEVO para la escena seleccionada.

Si el usuario desea seleccionar nuevos datos con los que visualizar e interactuar, tendrá que especificarlos y comenzar el proceso de nuevo. En la Figura 60 se representa, mediante un diagrama, el funcionamiento básico de este esquema de trabajo.



Figura 60 Esquema de funcionamiento de sistema de visualización web usando ontología STEVO.

La aplicación web usada en este marco de trabajo tiene la función de incluir un componente de visualización gráfico web, capaz de representar contenido interactivo en dos y tres dimensiones. Además, la arquitectura web debe de permitir actualizar el contenido de este componente de forma dinámica y la interacción del usuario. Con esta filosofía, el componente de visualización podrá activarse, indicando la escena que se va a visualizar, el acceso a la ontología STEVO y el conjunto de instancias a visualizar de los conceptos relacionados con los conceptos visuales que se representan en la escena. Si no se indica que instancias se han de visualizar, el componente visualizará todas las instancias de estos conceptos del dominio de datos.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

A continuación, se describirán los diferentes procesos que se llevan a cabo en el marco de trabajo, así como los diferentes casos de uso del mismo. Para finalizar este apartado, se detalla una solución tecnológica que permite implementar los procesos y soporta los diferentes casos de uso del sistema.

### *8.2.1. Procesos*

En el marco de trabajo, se han de llevar a cabo una serie de procesos para poder visualizar el contenido de un grafo de conocimiento, siguiendo las especificaciones de la ontología STEVO. Los procesos serían los siguientes:

- Un proceso de definición de los datos que se van a visualizar. Tras una petición web, iniciada por un usuario, este proceso, tiene que ser capaz de identificar el conjunto de URIs de las instancias del dominio de datos que se van a visualizar. Para ello usará una consulta que se puede lanzar más tarde sobre el grafo de conocimiento, o bien, tener acceso a esta información mediante otro proceso. Además, deber de conocer la forma de acceder a la ontología STEVO y el nombre de la escena, instanciado en dicha ontología, que se desea visualizar.
- Creación y/o actualización de las estructuras de datos que se usarán para visualizar la información. Con los datos obtenidos del anterior proceso de definición, se crearán estructuras de datos con información sobre los datos a visualizar, y sus propiedades, los datos necesarios para activar los comportamientos definidos en la escena y la información para poder generar la interfaz de usuario. Estos datos de interfaz serán consultados para mostrar botones y controles de relaciones, filtrado y control de tiempo. Durante este proceso se accede al contenido de la ontología STEVO y a través de esta, a los datos del grafo de conocimiento del dominio de datos.
- Creación de estructuras de visualización y de interfaz de usuario. En este proceso se crean estructuras de datos necesarias para representar gráficamente los datos y la interfaz de usuario. Estas estructuras están directamente relacionadas con la tecnología que va a representar visualmente la escena y la interfaz de usuario. Debido a esto, su definición dependerá totalmente de la tecnología usada para implementar la visualización y los elementos que permiten al usuario interactuar con los datos.
- Visualización e interacción de la información. Este proceso se encarga de usar las estructuras del proceso anterior, para representar la escena de visualización y mostrar la interfaz de usuario. En este proceso, puede requerirse información adicional del dominio de datos, por petición del usuario.

En la Figura 61 se representa un diagrama con estos procesos dentro del marco de trabajo y su relación con el grafo de conocimiento, la ontología STEVO y con las diferentes interacciones del usuario.

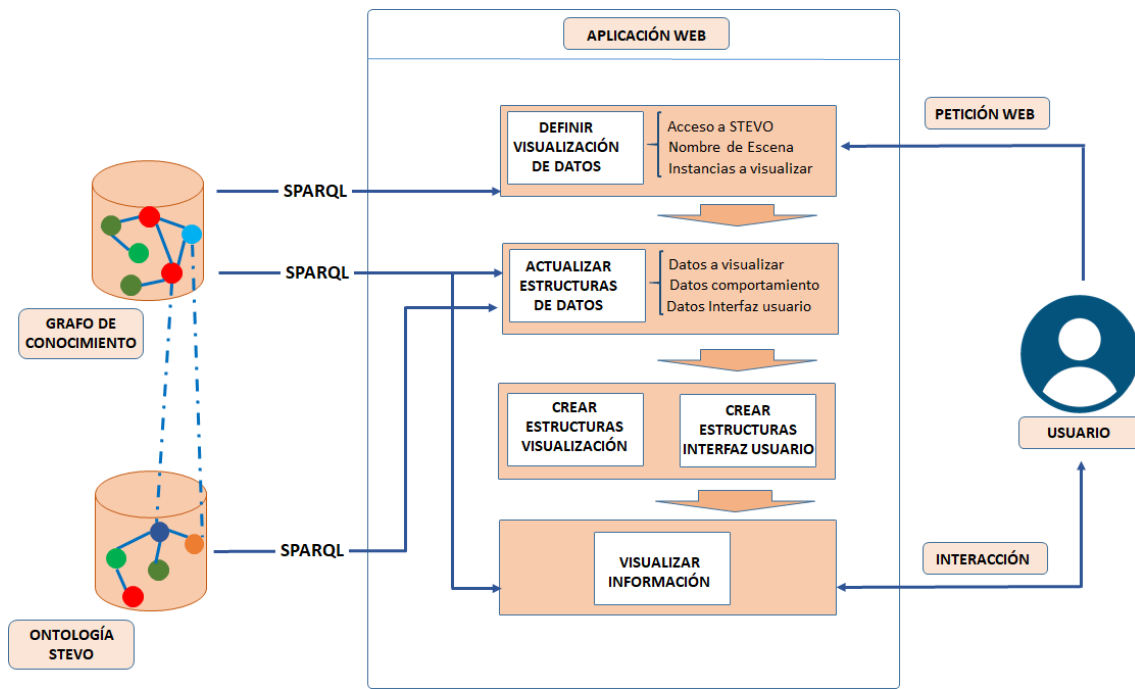


Figura 61 Procesos del marco de trabajo para visualizar el contenido de un grafo de conocimiento mediante la ontología STEVO.

### 8.2.2. Casos de uso

El marco propuesto en este capítulo es una aplicación web. Tal y como ya se ha descrito, esta aplicación web, recibe qué datos ha de visualizar, cómo acceder a la ontología STEVO, y el nombre de la escena, definida en la ontología STEVO. Esto definirá tanto la información, como la forma de verla. Las posibilidades de interacción con los datos vendrán definidas por la escena definida en STEVO y sus diferentes propiedades y relaciones. Por lo tanto, la aplicación tendrá tres grandes estados principalmente: iniciando la visualización y cargando datos, visualizando y permitiendo al usuario interactuar con ella, y cerrando la visualización.

Los casos de uso principales del usuario con el sistema están muy ligados a estos estados, siendo el de visualización e interacción el que tiene más complejidad.

El usuario, cuando desea visualizar información, inicia la aplicación. Dentro de ese proceso, la aplicación realiza las conexiones con STEVO y el grafo de conocimiento, para obtener la información necesaria, además, crea las estructuras necesarias para visualizar los datos gráficamente y mostrar la interfaz de usuario. En ese momento, el usuario puede visualizar los datos, además de interactuar con el sistema, realizando operaciones de navegación, filtrado de datos, usar un control de tiempo, de los disponibles, ver información de algún objeto y por último ver relaciones existentes entre los objetos que se están visualizando. Por último, el usuario puede cerrar la aplicación.

En la Figura 62 se muestra el diagrama de casos de uso del usuario con la aplicación web de visualización.

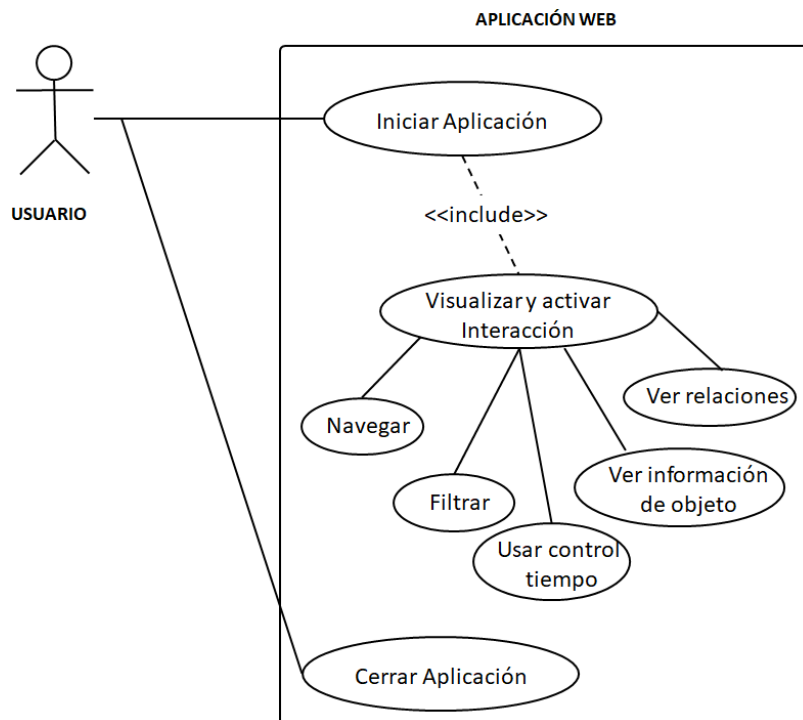


Figura 62 Diagrama de casos de uso del usuario con la aplicación web de visualización.

### 8.2.3. Arquitectura del sistema

La arquitectura del sistema tiene que contemplar los elementos software y hardware, así como su interconexión, para que el sistema sea operativo.

En el marco de trabajo propuesto varios elementos a considerar:

- Un soporte hardware y software para soportar y permitir el acceso al contenido de la ontología STEVO.
- Definir y desarrollar los módulos para acceder al contenido del grafo de conocimiento.
- El uso de un soporte web, para poder desplegar la aplicación web.
- Desarrollo de un componente de visualización que permita visualización de datos en dos y tres dimensiones sobre plataformas web.

En el capítulo 5 de esta memoria se describen tecnologías existentes en la actualidad para la gestión de grafos de conocimiento y también las que permiten visualizar datos en una aplicación web en dos y tres dimensiones.

De acuerdo al contenido de este capítulo, cualquier herramienta de gestión de grafos de conocimiento de las descritas podría ser válida para soportar la información de un grafo de conocimiento. Sin embargo, para ser válida en el marco de trabajo diseñado, debería permitir el acceso al contenido del grafo mediante un punto de acceso SPARQL y usando servicios web. Aunque el grafo de conocimiento es un elemento externo al marco de trabajo, debe de ofrecer dicha funcionalidad para que la aplicación web pueda acceder a su contenido.

Estas herramientas también serían válidas para mantener y dar acceso al contenido de la ontología STEVO. Sin embargo, dado el reducido tamaño de la misma, también puede ser tratada

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

como un archivo y manejada por cualquier librería que permita acceder a sus contenidos. De este modo la ontología STEVO puede ser tratada como un archivo accesible a la aplicación web, o bien, y gestionada desde alguna librería de la aplicación, o bien, usar una herramienta de gestión de grafos de conocimiento y acceder a su contenido mediante servicios web.

En función de las tecnologías analizadas, la aplicación web debe de estar basada en tecnología Javascript. Esto permitirá incluirla en la totalidad de frameworks web, además de que, por otra parte, se asegura una integración adecuada con las tecnologías de visualización web.

La aplicación web estaría integrada en el framework, junto a otros módulos y aplicaciones del sistema. Esta aplicación contará con los siguientes componentes:

- Una interfaz de la aplicación para recibir y procesar las peticiones web. Recibiría información de otros módulos, o aplicaciones web, para solicitar diferentes visualizaciones.
- Un gestor de datos. Que se encargará de acceder al contenido de la ontología STEVO y también al grafo de conocimiento del dominio de datos. Además, será el responsable de adaptarlos y suministrarlos al componente de visualización.
- Un componente de visualización, que se encargará de recibir los datos a visualizar, y cómo visualizarlos, para representarlos gráficamente y permitir la interacción con el usuario.

En la Figura 63 se muestra un diagrama con los elementos de la arquitectura del marco de trabajo y su interconexión.

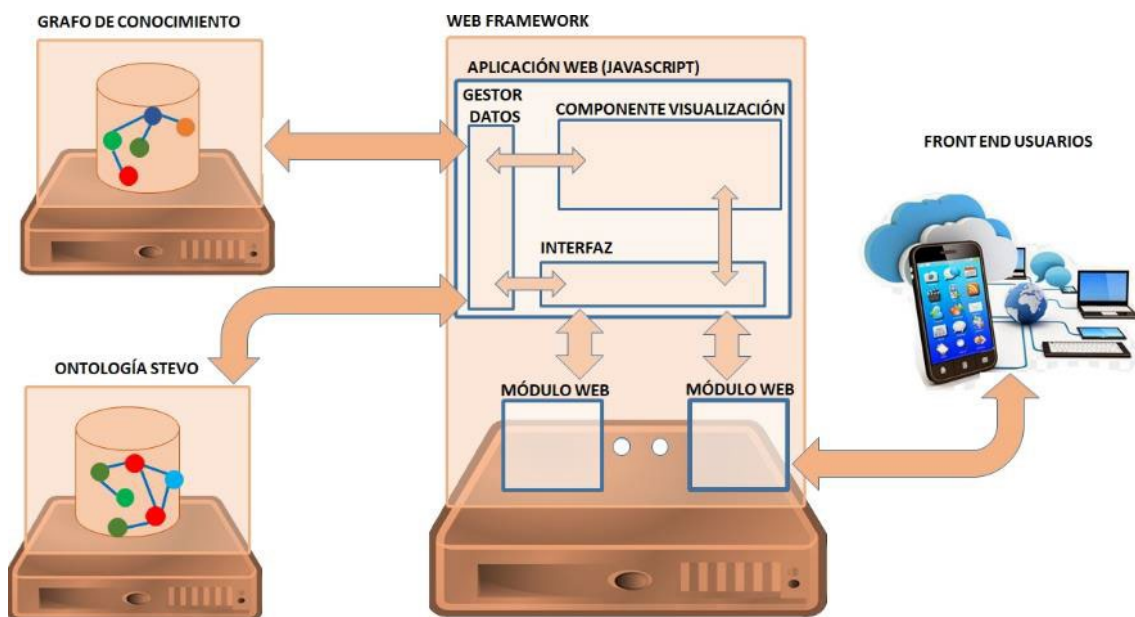


Figura 63 Arquitectura del marco de trabajo.

Puede parecer innecesario separar la gestión de los datos del componente de visualización. Sin embargo, además de hacer el sistema más modular, que siempre es positivo, es muy importante separar estos dos componentes. A priori, es lógico pensar que el procesamiento de los datos y su posterior visualización es un proceso secuencial. Es decir, los datos se obtienen, se procesan, y una vez hecho este trabajo, se visualizan. Sin embargo, el comportamiento no sigue este patrón en su totalidad. Es cierto que los datos de visualización van a seguir ese patrón, ya que hasta que no se sabe que visualizar, es imposible visualizar nada. Sin embargo, en la visualización hay muchos datos que no se tienen porque visualizarse inicialmente:

- Por una parte, están los datos de las relaciones entre los objetos. Obtener la información de que objetos están relacionados con otros, puede ser un proceso costoso en función del número de datos y propiedades. No es conveniente poder visualizar datos sobre las relaciones de forma inicial, ya que introduce un ruido considerable en la escena. Además, el mostrar relaciones puede requerir una interacción por parte del usuario. Debido a esto, esta información puede calcularse en otro componente, liberando procesamiento al componente de visualización. Únicamente cuando estuviese lista, estaría disponible para poder ser visualizada.
- Por otra parte, procesar toda la información es un objetivo muy ambicioso y poco realista. Como se ha indicado en esta investigación, la cantidad de datos digitalizados es muy grande y tiende a ser cada vez mayor. Por ejemplo, en el proyecto SEMAP; se manejan datos de 400.000 objetos y sólo son los museos de España. Los recursos del servidor están limitados y es inviable pretender que el sistema sea capaz de procesar y visualizar cantidades ingentes de información. Por este motivo, aunque definirlo correctamente no es un objetivo de esta investigación, es conveniente diseñar el sistema para que sea capaz de interactuar con los datos mediante consultas de forma incremental. Es decir, que en función de la zona en la que esté el usuario, se pudieran solicitar únicamente los datos que va a necesitar ver. De esta forma, se consultará al grafo del conocimiento por la información a visualizar en un momento determinado, no de toda la información. Esto implicaría el diseño de un sistema de predicción de posibles nuevos conjuntos de datos, y manejar una información de datos temporales, para acelerar la carga y hacer más fluida la navegación. El diseño y desarrollo de este módulo sobrepasa los objetivos de esta investigación, pero debido a esta necesidad, es conveniente separar el componente gestor de datos del de visualización, y contemplarlo así en el diseño.

#### 8.2.4. Implementación

Como ya se ha discutido en el capítulo 5 de esta investigación, hay varias tecnologías que permiten desarrollar el marco de trabajo. Debido a esto, también hay varias posibilidades para implementarlo. Seguidamente se expone una de estas posibilidades, indicando las tecnologías que se podrían usar para implementar la aplicación, los componentes externos, el componente de Gestor de Datos y el de Visualización.

##### Aplicación

A nivel de aplicación web, al estar basado en Javascript, hay varias tecnologías que permiten su desarrollo, desde el Javascript nativo implementado en cada navegador, hasta las diferentes librerías que se han publicado en la última década: JQuery(OpenJS 2022), REACT (Meta Platforms 2022), o VUE (You 2022). Es muy conveniente elegir cualquiera de estas librerías, en lugar del Javascript del navegador. Esto es debido a que hay notables diferencias entre los navegadores, desde temas de interfaz, hasta manipulación de elementos. Mediante el uso de estas librerías, aunque requieren una formación adicional, se eliminan todos estos problemas, además de obtener otras ventajas y componentes, desarrollados específicamente para ellas. En este caso, para la implementación propuesta, se ha optado por la librería VUE.JS, ya que es una de las más utilizadas en la actualidad y donde la comunidad de desarrolladores está muy activa. Además, en la actualidad existe un gran conjunto de recursos para formarse en su utilización.

Dentro de la aplicación web, hay que desarrollar una interfaz, que estará desarrollada en Javascript. La interfaz es muy simple y se limita a atender las peticiones HTTP, procesando y validando los parámetros, además de almacenarlos en las variables adecuadas. Además, en caso



de que todo esté correcto, debe de arrancar el proceso de visualización, activando los componentes Gestor de Datos y de Visualización. En caso de que haya algún problema deberá mostrar un mensaje indicándolo.

#### Componentes externos

El framework web elegido realmente es algo externo a la aplicación web y en principio debería ser secundario. Por simplicidad, y dado que también está basado en Javascript, se ha optado por el framework Node.JS (Open 2022), pero realmente podrías ser cualquier otro, como Spring, por ejemplo.

Sucede lo mismo con la plataforma para soportar el grafo de conocimiento del dominio de datos. Es un elemento externo a la aplicación, y mientras tenga acceso a su contenido mediante SPARQL y servicios web, la elección de una u otra plataforma debería de ser algo dependiente de otros factores. En esta propuesta de implementación se ha optado por la plataforma Virtuoso, por cumplir los requisitos de la arquitectura propuesta, estar disponible de forma gratuita, y ser una de las más utilizadas y de mejor rendimiento a nivel general. De todas formas, tal y como se ha comentado, cualquier herramienta similar, que permita el acceso al contenido mediante servicios web y ofrezca un punto de acceso SPARQL, sería igualmente válida. La diferencia con la plataforma elegida estaría principalmente en el rendimiento.

#### Acceso a STEVO.

El contenido de la ontología de STEVO puede introducirse en una plataforma que soporte grafos de conocimiento, como podría ser Virtuoso. También se puede proporcionar acceso mediante un servidor de aplicaciones que permita ejecutar módulos implementados en lenguaje Java, como es el caso de Apache Tomcat, o similares, y así usar librerías Java, como JENA, o directamente enlazar con OWLAPI.JAR y proporcionar los métodos necesario para obtener los contenidos de los conceptos recorriendo las instancias de determinados conceptos, y sus propiedades y relaciones, incluso usar consultas en SPARQL. También hay soluciones basada en JS y NODE.JS. En la Tabla 29, se muestran varias librerías de código abierto, con acceso a contenido RDF, incluso algunas permiten realizar consultas en SPARQL.

Instalar una plataforma como Virtuoso, diseñada para soportar grandes grafos de conocimiento, para acceder a un archivo con una ontología como STEVO, de reducido tamaño, sería un uso de recursos innecesario. Por otra parte, usar un framework web basado en NODE.JS, o que permita el uso de código Java, si es el mismo que el usado para incluir la aplicación web no supone ningún problema. Pero, en caso contrario, implica un consumo de recursos importante, y un costoso mantenimiento.

La opción que menos recursos consume y es compatible con cualquier sistema es usar una librería de acceso al contenido de una ontología a través de un archivo, mediante una librería escrita en Javascript sin dependencias de ningún servidor.

Tal y como se muestra en la Tabla 29, hay varias librerías que incluyen esta funcionalidad. De ellas, se selecciona rdfstore-js (Garrote 2016) porque está en activo desde 2012 y permite acceder a contenido RDF a nivel estructural, y también realizar consultas SPARQL.

De esta forma el archivo con la ontología STEVO podría estar, idealmente, en el mismo servidor web que soporta la aplicación web, pero tampoco supondría ningún problema ubicarlo en un servidor externo, siempre que se pudiera acceder por HTTP, o mediante un servicio web, al contenido del archivo.



**CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO**

*Tabla 29 Librerías para acceder a RDF con JS y con NODE.JS . Fuente:(W3C 2019)*

| Librería        | Licencia                  | Soporta                              | Entorno     | Interfaz                |
|-----------------|---------------------------|--------------------------------------|-------------|-------------------------|
| node-rdf        | Public Domain             | Turtle                               | node.js     | RDF Interfaces          |
| sparql-spin-js3 | Public Domain (Unlicense) | SPARQL (Parsed into SPIN vocabulary) | node.js     | Custom                  |
| rdfstore-js     | MIT                       | SPARQL, Turtle                       | JS, node.js | RDF Interfaces          |
| rdfquery        | MIT                       | RDFa                                 | JS          | Custom                  |
| N3.js           | MIT                       | Turtle, TriG, N-Triples, N-Quads     | JS, node.js | Custom, Node.js Streams |
| SPARQL.js       | MIT                       | SPARQL 1.1                           | JS, node.js | Custom                  |
| jsonld.js       | BSD                       | JSON-LD                              | JS, node.js | JSON-LD                 |
| green-turtle    | BSD                       | RDFa, Turtle, Microdata (partial)    | JS          | RDFa API                |
| RDF-Ext         | MIT                       | JSON-LD, RDF/XML, Turtle             | JS, node.js | RDF Interfaces          |
| rdflib.js       | MIT                       | RDF/XML, Turtle, N3, RDFa            | JS, node.js | Custom                  |
| graphy.js       | ISC                       | Turtle, TriG, N-Triples, N-Quads     | JS, node.js | RDFJS Interfaces        |

#### Gestor de datos

El gestor de datos a desarrollar en esta investigación no será muy complejo, ya que no realiza procesos en segundo plano, ni predicciones de carga ni paginación de datos, que optimicen la carga del sistema, ya que esto sobrepasa los objetivos de esta investigación. Simplemente se encargará de cargar los datos y suministrarlos al componente de visualización. Debido a esto, a efectos prácticos, podría estar incluido en el componente de visualización. Pero para seguir con la línea marcada por la arquitectura del marco de trabajo, habrá que separarlo como un componente diferente al de visualización.

Aunque en esta propuesta de implementación no se realizan estos procesos de optimización, idealmente hay que seleccionar una tecnología que permita la ejecución de procesos en segundo plano. De esta manera, se podrá ampliar en un futuro para poder realizar estas mejoras.

Lógicamente, también podría extraerse este componente de la aplicación web, para ello podría instalarse en un servidor externo que ofrezca servicios web. Esto, por una parte, implicaría una separación de la arquitectura propuesta, además de que podría causar con relativa facilidad serios problemas de sobrecargas del sistema. Dado el conjunto de datos que se pueden requerir visualizar, en el que no es nada extraño que se manejen varias decenas de miles de datos, si un

número elevado de usuarios solicitan varias peticiones de forma simultánea, se requeriría la implantación de sistemas de altas prestaciones para evitar estos problemas.

Por este motivo es muy recomendable mantenerlo como parte de la aplicación web y que sea el navegador cliente el que soporte este procesamiento de forma distribuida.

Considerando los argumentos expuestos, una de las tecnologías que mejor los satisface sería Javascript. Tiene la ventaja de que funciona en cualquier framework web, no es necesario aprender ninguna tecnología adicional y el procesamiento se realiza en el navegador cliente.

La programación en segundo plano no era posible en Javascript hasta hace algunos años. En la actualidad la mayoría de los navegadores tienen implementada la tecnología Service Worker (Google. 2021), que permite ejecutar procesos en segundo plano con este lenguaje sin una repercusión en la energía de los dispositivos (Malavolta, Procaccianti et al. 2017) De esta forma, se satisface el requisito de que en un futuro se podría mejorar el componente con optimizaciones de la carga de datos.

#### Componente de visualización.

El componente de visualización es el componente más crítico, en lo referente a tecnología. Esto es debido al hecho de tener que integrarse en una aplicación web, y ser capaz de representar elementos gráficos en dos y tres dimensiones en la mayoría de navegadores. En la actualidad hay varias tecnologías que permiten realizar esta funcionalidad, pero muchas de ellas han entrado en desuso y no tienen mantenimiento, como es el caso de Java3D, o X3D, otras no son soportadas en ningún navegador moderno, como es el caso de las tecnologías basadas en Flash.

Tal y como se expuso en el capítulo 5 de esta investigación, las tecnologías basadas en WebGL son las que cumplen la funcionalidad, funcionan en la mayoría de navegadores y forman parte de una comunidad activa.

Por una parte, el componente de visualización tendrá un módulo gráfico desarrollado en WebGL, que tendrá un API en Javascript, el cual vendrá fijado por la tecnología implementada para generar el código WebGL. Por otra parte, existirá otro módulo desarrollado en JavaScript que se encargará de soportar la comunicación entre el módulo WebGL y el resto de componentes de la aplicación.

De estas tecnologías estudiadas que permiten implementar gráficos en WebGL, las más utilizadas son Three.js y Unity.

La primera es gratuita (MIT License) y está basada en el lenguaje Javascript, mientras que la segunda no es gratuita, y está basada en el lenguaje C#. Las dos tecnologías están respaldadas por una comunidad de desarrolladores e investigación muy activa, en la que continuamente se desarrollan mejoras y nuevos componentes.

Unity es una tecnología propietaria, mientras que Three.js es una tecnología de código abierto. Además, Unity requiere la precarga de un plugin en el navegador que implica un pequeño retraso la primera vez que se usa en el navegador.

Aunque Unity es una tecnología propietaria, desde sus inicios permite generar aplicaciones, sin necesidad de adquirir licencia, exceptuando casos que generen unos beneficios importantes, con la única penalización de mostrar su logo durante el arranque de la aplicación.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

Con toda esta información, a priori, es más recomendable el uso de Three.JS para el desarrollo del componente, ya que es un sistema de código abierto y tiene un rendimiento similar. Sin embargo, considerando factores como la popularidad de Unity y la gran cantidad de Assets disponibles, hace que sea un sistema mejor para asegurar su mantenimiento y facilitar su integración con otros lenguajes, dispositivos, nuevas funcionalidades y sistemas operativos. Esto, junto con el hecho de que el equipo de desarrollo tiene una experiencia mucho mayor de desarrollo con Unity que con Three.JS, se decide usar Unity para la primera versión del marco de trabajo el componente, que se usará en esta investigación.

Unity, además, permite generar ejecutables en varias plataformas. Si en un futuro se decidiera usar este componente para integrarlo en una aplicación standalone, la adaptación no sería muy costosa.

El componente generado en Unity está diseñado para ser integrado en una aplicación web, por lo que ofrece mecanismos para invocar métodos de las clases de Unity, desde la página web, usando Javascript, y también, poder invocar métodos Javascript desde scripts C# de Unity. En la documentación de Unity (Unity Technologies. 2022) se describe perfectamente cómo realizar este proceso.

En la Figura 64 se muestra un diagrama con la tecnología empleada para el desarrollo del marco de trabajo, así como su integración en el framework web, el acceso a la ontología STEVO y a los datos del grafo de conocimiento.

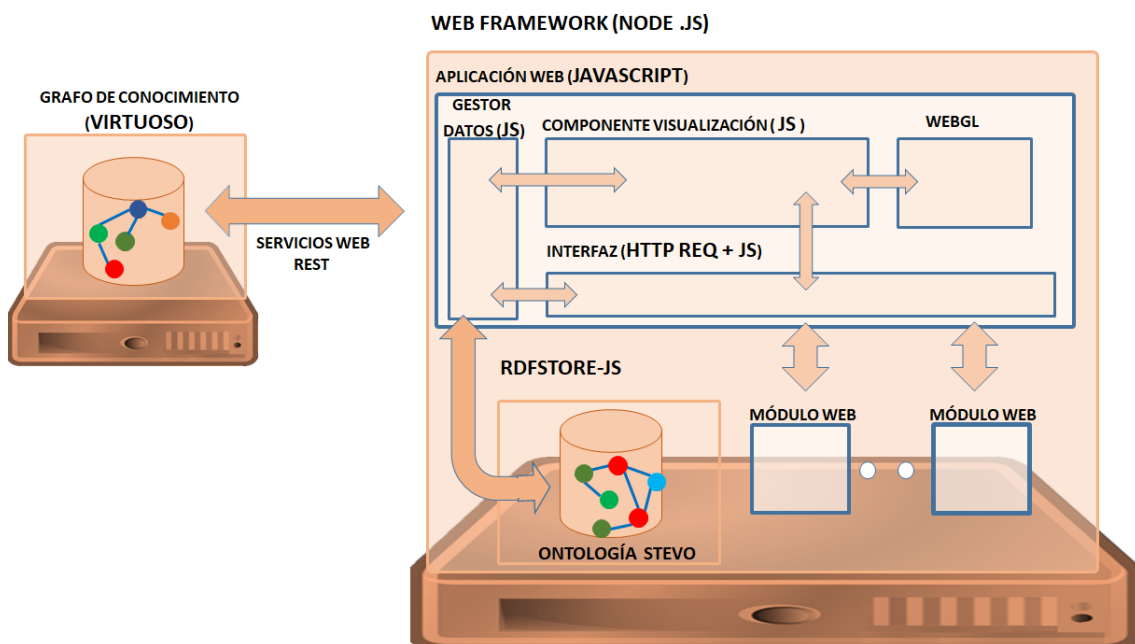


Figura 64 Diagrama con la tecnología propuesta para el desarrollo de los componentes del marco de trabajo y su comunicación con el framework web y el resto de elementos necesarios.

Según la propuesta de desarrollo de esta investigación, toda la aplicación web estaría desarrollada en JavaScript, a excepción de la parte de visualización que estaría implementada en WebGL, en este caso con código generado a partir de Unity.

A continuación, se detallan los diferentes módulos de la aplicación, indicando las clases que los componen, su funcionalidad, API y diagrama de estados.

Primero desde el punto de vista de la comunicación, por una parte entre la aplicación web y los diferentes módulos web del framework en el que estará embebida la aplicación, y también la comunicación entre los diferentes componentes de la aplicación.

Después de exponer los módulos desde el punto de vista de la comunicación, se expone también el componente de visualización desarrollado a partir de Unity. Además de usar una tecnología diferente, la funcionalidad de este módulo tiene una complejidad y unos objetivos muy diferentes del resto de módulos del sistema. Mientras la primera parte sería para conocer que visualizar, y obtenerlo, este componente es el que realmente se encarga de mostrar la información recibida según las especificaciones de STEVO.

#### 8.2.5. Comunicación

En esta sección se describe con más detalle la comunicación entre los diferentes componentes del marco de trabajo: El Gestor de Datos, el Componente de Visualización y la Interfaz. Tal y como se ha descrito, todos los módulos descritos en esta sección, exceptuando el Componente WebGL, están desarrollados en JavaScript y están intercomunicados. El diagrama de clases que conforman estos componentes se muestra en la Figura 65.

El código fuente de estas clases puede descargarse y consultarse en un repositorio público (Sevilla, J. 2022) Github bajo licencia MIT.

#### Interfaz de la aplicación.

El proceso de comunicación comienza, bien porque un usuario, o bien porque un evento del sistema, solicita la visualización de una página web que tiene incluida la aplicación web. Esta página tiene que indicarle a través de una petición web a la aplicación:

- El nombre de la escena a visualizar.
- Cómo acceder a la ontología STEVO. Si el contenido no es muy grande, que será lo habitual, puede estar en un archivo. En este caso, el acceso se realizará mediante una URL que permita al sistema acceder a este archivo. En el caso de contenidos con tamaño considerable, o por otro motivo, puede estar ubicado en un servidor que permita acceder al contenido mediante servicios web. Si se da esta situación habrá que indicar la URL y los parámetros del servicio para poder acceder al contenido.
- Cómo acceder al contenido del grafo de conocimiento. Aunque el grafo de conocimiento generalmente será accesible mediante servicios web, la aplicación web, al igual que con la ontología STEVO permite especificar los datos del servicio, además de una URL para poder acceder al contenido del grafo de conocimiento.
- Los datos a visualizar, mediante diferentes posibilidades, se especificarán los conceptos del dominio de datos y las URIS de instancias de estos conceptos que se desee visualizar.

Este diseño, permite a una web, poder visualizar los datos de una forma tradicional, mediante formularios, listas de datos y/o imágenes, y si el usuario de la web lo decide, visualizar el contenido buscado, mediante la aplicación de visualización. Lógicamente, la aplicación puede usarse de forma exclusiva también. El comportamiento de la aplicación será visualizar el conjunto de datos especificado, y en caso de que no se defina un conjunto, se visualizarán todos los datos de la escena especificada mediante la ontología STEVO.

Para ello, la aplicación web define la clase *VisualizationApplication*, desarrollada en JavaScript. Esta clase, al igual que las clases *DataManager* y *VisualizationComponent*, que representan

respectivamente al Gestor de Datos y al Componente de Visualización, extienden de la clase `ApplicationComponent`. Una vez instanciada la clase *VisualizationApplication*, se ha de invocar al método *init*. Este método recibe cuatro parámetros de tipo `String`:

- `scene`: Nombre de la escena especificada en STEVO.
- `stevoAddress` : Texto en formato JSON con los datos para acceder a la ontología STEVO.
- `domainAddress`: Texto en formato JSON con los datos para acceder al grafo de conocimiento del dominio de datos.
- `domainData`: Texto en formato JSON con los datos, o información para acceder a ellos.

El formato JSON de los dos últimos parámetros sigue el siguiente patrón, que indica el tipo de datos (`type`) y a continuación los datos, mediante una lista JSON, denominada `data`.

{`type`: <tipodedatos>, `data`:<datos>}

En la Tabla 30 se especifica el formato JSON de los dos últimos parámetros.

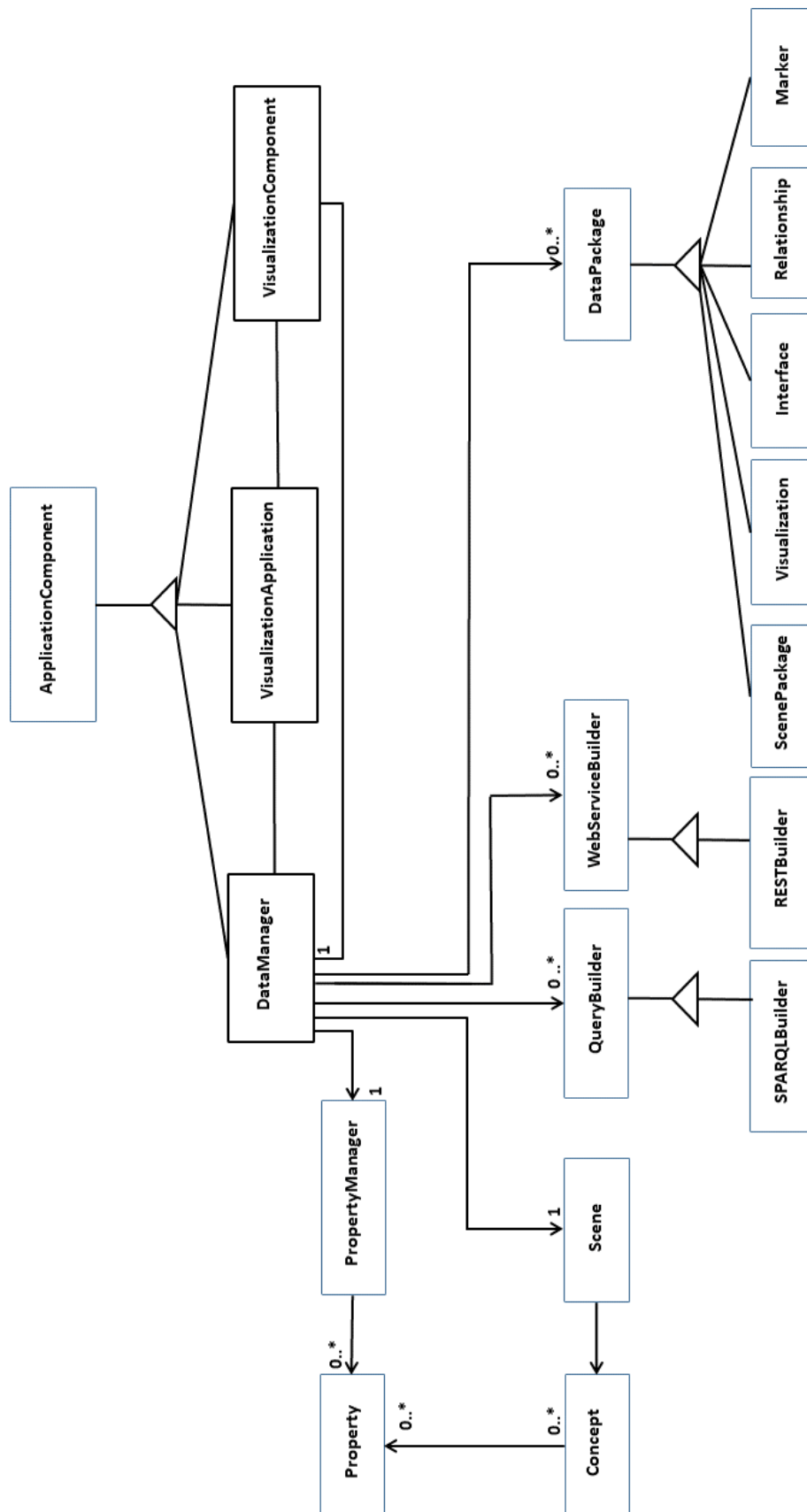


Figura 65 Diagrama de clases de los módulos de la aplicación web de visualización.

**CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO**

*Tabla 30 Formato JSON usado en los parámetros stevoAddress, domainAddress y domainData*

| Parámetro     | Tipo    | Datos   |
|---------------|---------|---|
| stevoAddress  | FILE    | URL del archivo.  |
|               | SERVICE | {<br>url: < url del servicio web >,<br>params : {<br>[ name : < nombre >, value : < valor > ]},<br>query_param: < nombre del parámetro para consulta ><br>} |
| domainAddress | FILE    | URL del archivo.  |
|               | SERVICE | {<br>url: < url del servicio web >,<br>params : {<br>[ name : < nombre >, value : < valor > ]},<br>query_param: < nombre del parámetro para consulta ><br>} |
| domainData    | QUERY   | {<br>[ concept: < uri del concepto >, query: < expresión > ]<br>}   |
|               | DATA    | {<br>[ concept: < uri del concepto>, { [ uri : < instancia uri > ]}]<br>}   |

El tipo FILE en stevoAddress se usa cuando la ontología STEVO está en un archivo accesible mediante una URL.

El tipo SERVICE en stevoAddress se usa cuando la ontología está accesible mediante un servicio web, en este caso se indicará la URL del servicio (url), los parámetros y sus respectivos valores (name, value) y se indicará el parámetro donde se formula la consulta (query\_param).

Los tipos en domainAddress siguen el mismo formato que los de stevoAddress. Lógicamente el dominio de datos será accesible generalmente mediante un servicio web, en lugar de un archivo, pero se mantiene por simplicidad y para poder visualizar grafos de conocimiento de reducido tamaño.

El tipo QUERY en domainData se usa para asociar a cada concepto del grafo de conocimiento (concept) una consulta (expresión) que permita obtener las URIs a visualizar de dicho concepto.

El tipo DATA en domainData se utiliza para indicar a cada concepto del grafo (concept) un conjunto de URIS de instancia (uri) que se han de visualizar en este concepto.

Una vez recibida esa información, se inicia el Gestor de Datos y el Componente de Visualización. En caso de existir problemas, la aplicación, genera un mensaje de error que puede ser mostrado por el framework web.

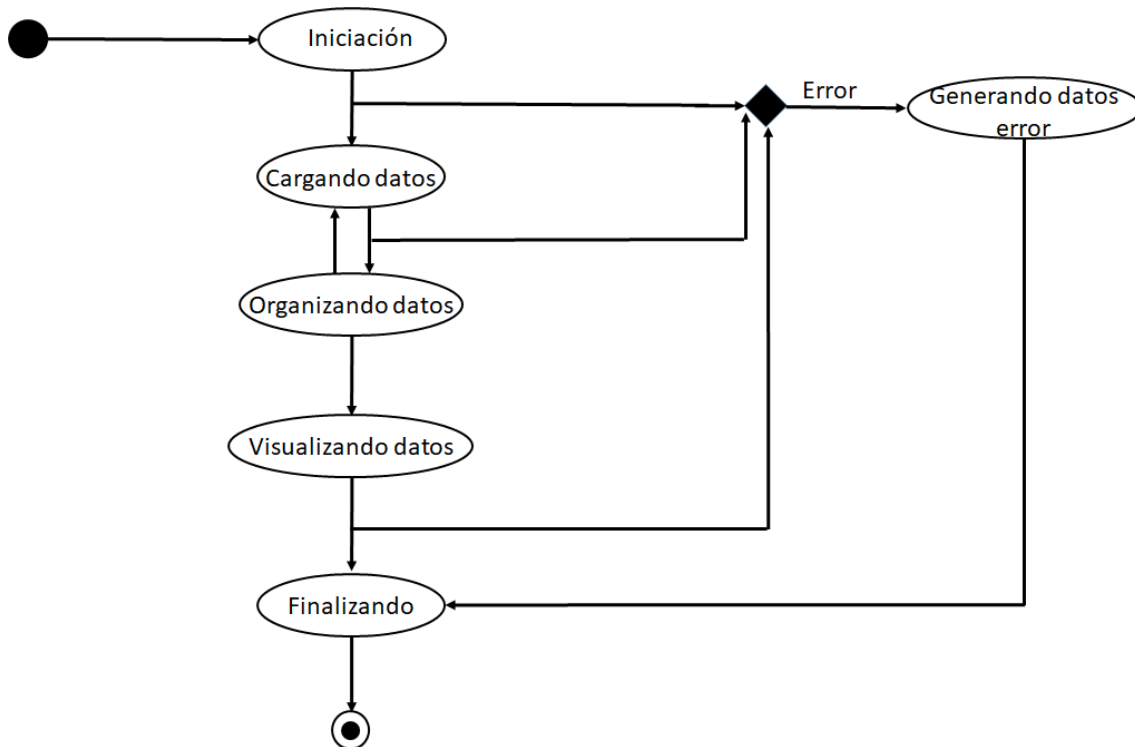


Figura 66 Diagrama de estados de la aplicación web de visualización.

El estado de la aplicación puede obtenerse mediante la invocación del método *getStatus*, de la instancia de la clase *VisualizationApplication*, que devuelve un string con el estado en el que se encuentra. En la Figura 66 se muestra el diagrama de estados de la aplicación. Los diferentes estados en los que se puede encontrar la aplicación web son:

- Iniciación (INITIALIZING). El estado inicial, hasta que se procede a cargar datos.
- Carga de datos (LOADING\_DATA). Este estado está vigente desde que comienza la carga de datos, hasta que finaliza.
- Organización de datos (ADAPTING). Tras la carga de los datos, además de inicializar las estructuras básicas, se generan nuevas estructuras para agilizar procesos como relaciones, filtros y agrupaciones. Este estado, se puede realizar, de forma sincronizada al de carga, ya que tras organizar la información puede solicitarse información adicional.
- Visualización (VISUALIZATING). Se activa cuando el componente de visualización ha recibido todos los datos y comienza la visualización de datos.



**CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO**

- Finalizando (CLOSING). El estado está activo cuando el usuario decide cerrar la aplicación de visualización, o el cierre se ha causado por un evento del sistema y termina cuando se han terminado las operaciones de liberación de memoria.
- Generando datos error (ERROR). En el caso de que haya algún error, la aplicación pasa por este estado y a continuación por el estado de Finalización.

Tabla 31 API de la Interfaz pública de la clase VisualizationApplication .

| Función                 | Parámetros   | Retorno | Funcionalidad   |
|-------------------------|--|---------|---|
| <u>init</u>             | VisualizationApplication app<br>string scene<br>string stevoAddress<br>string domainAddress<br>string domainData | bool    | Iniciar la aplicación de visualización.                                   |
| <u>getStatus</u>        | void   | string  | Devuelve el estado actual de la aplicación web de visualización.          |
| <u>getErrorMsg</u>      | void   | string  | Devuelve el mensaje de error generado por la aplicación de visualización. |
| <u>getErrorCode</u>     | void   | int     | Devuelve el código de error generado por la aplicación de visualización.  |
| <u>close</u>            | void   | void    | Cierra la aplicación de visualización.                                    |
| <u>setError</u>         | int code<br>string message   | void    | Indicar a la interfaz que se ha generado un error en la aplicación.       |
| <u>getScene</u>         | void   | string  | Devuelve el nombre de la escena a visualizar especificada en STEVO.       |
| <u>getSTEVOAddress</u>  | void   | string  | Devuelve la información de acceso a STEVO.                                |
| <u>getDomainAddress</u> | void   | string  | Devuelve la información de acceso al grafo de conocimiento.               |
| <u>getDomainData</u>    | void   | string  | Devuelve la información sobre los datos a representar.                    |

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

En el caso de que exista algún error, el mensaje de error y el código de error generado, pueden ser consultados con los métodos *getErrorMessage* y *getErrorCode* de *VisualizationApplication*.

En todo momento, la aplicación de visualización puede cerrarse el método *close* de *VisualizationApplication*.

En la Tabla 31 se muestra el nombre de las funciones, parámetros, retorno y funcionalidad de las funciones de la API de la Interfaz de la aplicación web de visualización accesible a través de la clase *VisualizationApplication*.

### Gestor de datos.

El gestor de datos se inicia creando, desde la instancia de *VisualizationApplication*, una instancia de la clase JavaScript *DataManager*. Una vez creada, se invocará al método *init*, el cual recibe como parámetro la instancia de *VisualizationApplication*. A través de esa instancia, conocerá la escena y los datos a representar, además de cómo acceder a STEVO y al grafo de conocimiento.

El método *init* devuelve como resultado un booleano, si todo es correcto, la aplicación procede a enviar paquetes de datos al componente de visualización desde el Gestor de Datos. Si hay algún problema, la aplicación finaliza en estado de Error.

Cuando el componente de visualización tiene los datos básicos, puede proceder a visualizar la información, pero durante el proceso de carga puede generar estructuras intermedias de visualización, e interfaz.

Hay cinco paquetes de datos: Escena (SCENE), marcadores (MARKERS), visualización (VISUALIZATION), interfaz (INTERFACE) y relación (RELATIONSHIP). En cada paquete se transmite la siguiente información:

- Escena, este paquete tiene la información necesaria para saber el tipo de escena que se va a visualizar, así como la información de los comportamientos definidos en STEVO.
- Marcadores, se transmite la información gráfica necesaria para representar los diferentes marcadores definidos en STEVO en la escena seleccionada, agrupados por concepto.
- Visualización, la información necesaria para representar los datos (URI y posicionamiento (x, y,z)), agrupados por concepto.
- Interfaz. Este paquete de datos contiene la información que el componente de visualización necesita para mostrar la interfaz. Datos sobre las propiedades filtrables y relacionables. Esta información contiene, para cada propiedad sus diferentes posibles valores.
- Relación. Conjunto de propiedades relacionables y el número de instancias que tienen el mismo valor en cada propiedad.

Los datos de los diferentes paquetes pueden obtenerse con el método *getPackage(string packageName)*, de la clase *Datamanager*, el cual devuelve una instancia de *DataPackage* con 3 posibles valores:

- Nulo: Aún no hay datos disponibles.
- Instancia de *DataPackage*, con datos, con un paquete del conjunto de paquetes obtenidos del grafo de conocimiento.

CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

- Instancia de `DataPackage`, sin datos, indicando que ya no hay más paquetes con información de ese tipo.

De esta forma, el componente de visualización puede invocar los datos como desee, cargando los paquetes de cada tipo uno a uno, conforme se vayan obteniendo desde el grafo de conocimiento.

Además, la API del gestor de datos, ofrece la función: `getDataFromURI(string URI)`, que devuelve un string en formato JSON con los valores de las diferentes propiedades del Concepto Visual de STEVO cuya URI sea la especificada como parámetro.

Por último, en la API del gestor de datos se invoca al método `close`, para liberar los datos y recursos creados por el gestor de datos.

En la Tabla 32 se listan las funciones, parámetros, retorno y funcionalidad de las diferentes funciones de la API del Gestor de Datos.

Tabla 32 API del Gestor de Datos de la aplicación web de visualización.

| Función                            | Parámetros   | Retorno     | Funcionalidad  |
|------------------------------------|--|-------------|--|
| <u><code>init</code></u>           | string scene<br>string stevoAddress<br>string domainAddress<br><br>string domainData | bool        | Iniciar el gestor de datos y comenzar la carga de datos.                                 |
| <u><code>getPackage</code></u>     | string packageName   | DataPackage | Devuelve un paquete del conjunto de paquetes obtenidos, del tipo citado en el parámetro. |
| <u><code>getDataFromURI</code></u> | string uri   | string      | Devuelve información sobre la instancia cuya URI se especifica.                          |
| <u><code>close</code></u>          | void   | void        | Cierra el Gestor de datos para que comience a liberar memoria.                           |

En el Gestor de Datos, habría que desarrollar la clase `DataManager`, que soportaría el núcleo del componente, además de la comunicación con el resto de módulos de la aplicación. Al estar este componente desarrollado en JavaScript habría que desarrollar las siguientes clases:

- `Scene`, que contendría la información sobre la escena, obtenida de la ontología STEVO.
- `Concept`, asociado a `Scene`, que representaría los conceptos visuales de la escena.
- `Property`, a su vez representa las propiedades visuales de STEVO, asociadas a cada concepto.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

- **PropertyManager.** Sería un gestor de propiedades, como su nombre indica, y mantendría información sobre los posibles valores de las propiedades filtrables, así como los objetos relacionados con el mismo valor de cada propiedad.
- **QueryBuilder.** Se trataría de una clase, cuya función sería la de construir de forma adecuada la consultas para ser lanzadas sobre el grafo de conocimiento y/o STEVO. Esta clase tendría una especialización para el tratamiento de consultas en lenguaje SPARQL, en caso de que hubiera que usar otro lenguaje, habría que crear otra especialización.
- **WebServiceBuilder.** Se encargaría de construir la expresión de un servicio web a partir de los datos obtenidos de la interfaz de la aplicación web, en caso de que el acceso a la información fuera a través de este canal. Tendría una especialización, **RESTBuilder**, que se encargaría de generar servicios web RestFul.
- **DataPackage.** Representa un paquete de datos que ha de suministrarse al componente de visualización. Permite construir dichos paquetes, a partir de la información suministrada por la interfaz de la aplicación y el contenido de STEVO. Habría una especialización para cada uno de los cinco tipos de paquetes generados por el Gestor de Datos.

### Componente de visualización

El componente de visualización tiene una API muy reducida, ya que es el componente principal de la aplicación y el que soporta más interacción con el usuario, a través de la interfaz gráfica, y además un alto nivel de procesamiento debido a la representación gráfica de los datos.

Además, este componente usa dos tecnologías, un módulo gráfico, desarrollado en WebGL, generado con Unity, según la propuesta de esta investigación, y un módulo desarrollado en JavaScript, que se encarga de la comunicación entre el módulo WebGL y el resto de componentes de la aplicación web.

El API, desarrollado en JavaScript, únicamente tiene tres funciones, soportadas por la clase *VisualizationComponent* :

- *init*, que se encarga de iniciar el componente de visualización. Este método recibe como parámetro una instancia de la clase *DataManager* y otra de la clase *VisualizationApplication*, de esta forma puede comunicarse con los dos componentes de la aplicación. Y una vez iniciado, se encarga de acceder de forma ordenada los paquetes de datos al Gestor de Datos. En el caso de que haya algún error se lo comunica a la interfaz de la aplicación. Una vez tenga la información necesaria procederá a representar gráficamente la información. Si una vez comenzado el proceso de visualización e interacción con el usuario, necesita más información, consultará periódicamente al gestor de datos para obtener los datos en cuanto estén disponibles. En caso de que hubiera algún error se lo comunicaría a la interfaz de la aplicación, además de proceder al cierre del componente.
- *close*. Este método se encarga de destruir el componente, liberando previamente la memoria utilizada.

En la Tabla 33 se listan las funciones, parámetros, retorno y funcionalidad de las diferentes funciones de la API del Gestor de Datos.

Tabla 33 API del Componente de Visualización de la aplicación web de visualización

| Función | Parámetros | Retorno | Funcionalidad |
|---------|------------|---------|---------------|
|---------|------------|---------|---------------|

CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

|              |  |      |   |
|--------------|--|------|---|
| <u>init</u>  | VisualizationApplication app<br>DataManager dm | void | Iniciar el componente de visualización, y comenzar la representación gráfica en cuanto tenga la información necesaria desde el Gestor de Datos. |
| <u>Close</u> | void   | void | Cierra el componente de visualización y libera los recursos del sistema utilizados.   |

#### 8.2.6. Componente de visualización

Tal y como se ha descrito a lo largo de este capítulo, el componente de visualización tiene dos módulos:

- Un módulo gráfico, implementado en WebGL, desarrollada en C#, con Unity, que se encarga de la representación gráfica y la interacción con el usuario.
- Un módulo de comunicación desarrollado en JavaScript, compuesto por la clase VisualizationComponent, que básicamente servirá de interfaz entre el componente WebGL y el resto de módulos de la aplicación.

Primero se describirá la comunicación entre los dos módulos del componente y a continuación se detallará la parte desarrollada mediante C#.

#### Interfaz

Tal y como se especifica en la arquitectura de la aplicación, la interacción entre estos dos módulos se compone de una serie de mensajes destinados a intercambiar información sobre cómo representar los datos y que datos representar, así como la comunicación de operaciones de finalización y error. En Tabla 34 se especifican las funciones que publican estos dos módulos para poder realizar dichas operaciones.

Tabla 34 API de la Interfaz entre los diferentes módulos del Componente de Visualización.

| Nombre  | Parámetros | Funcionalidad   |
|---|------------|---|
| <b>Peticiones desde Módulo C# (Unity) a Módulo JavaScript</b> |            |   |
| readyForData()  |            | Indicar al módulo JavaScript que está preparado para recibir más datos.       |
| getDataForURI()   | String     | Solicitar información de la instancia cuya URI está indicada en el parámetro. |
| getDataFromRegion()   | String     | Solicitar el envío de datos de visualización de una zona                      |

CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

|  |        |   |
|--|--------|---|
|  |        | definida por la cadena de texto en formato JSON.  |
| cancelDataFromRegion()                                 | String | Indicar al sistema la cancelación de petición de datos sobre una zona codificada en formato JSON. |
| visualizationError()                                   | String | Indicar al módulo JavaScript que ha sucedido un error, codificado mediante el parámetro.          |
| Closed()   |        | Indicar que el módulo se ha cerrado debidamente.  |
| <b>Peticiones desde Módulo JavaScript a C# (Unity)</b> |        |   |
| sendDataScene  | String | Envío de datos de definición de la Escena. Los datos están codificados en formato JSON            |
| sendDataVisualization                                  | String | Envío de datos de visualización. Los datos están codificados en formato JSON.                     |
| sendDataInterface                                      | String | Envío de datos de interfaz codificados mediante una cadena de texto en formato JSON.              |
| sendDataRelationShip                                   | String | Envío de datos sobre las relaciones entre los objetos codificados en formato JSON.                |
| sendDataForURI   | String | Envío de datos sobre la URI solicitada en formato JSON.   |
| Close  |        | Solicitud de finalización desde la aplicación.  |

Para el desarrollo de la comunicación entre el código C# de Unity y el resto de componentes JavaScript de la aplicación web, Unity requiere la especificación de dos archivos. Un archivo JavaScript con las funciones, que directamente llaman a las funciones de la página web que contiene el módulo gráfico, y una clase en c#, que hace de contenedor de las funciones del archivo en JavaScript. El código en Unity sólo tiene que llamar a las funciones de la clase contenedor.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

A continuación, se muestra un fragmento del archivo WebGLPlugin.jslib con el código JavaScript de las funciones que se pueden invocar desde el módulo gráfico WebGL hacia el resto de módulos de la aplicación.

```
// Functions called by Unity module
mergeInto(LibraryManager.library, {
  // Methods to send message to the page
  // Method to get data from a URI specified in the parameter
  getDataForURI: function (text) {
    // Convert bytes to the text
    var convertedText = Pointer_stringify(text);
    // Pass message to the page
    getDataForURI_FromUnity(convertedText);
    // This function is embeded into the page
  }

  // ----- . . . -----
});
```

Seguidamente se expone el fragmento del archivo WebGLPlugin.cs en C# , que sirve de contenedor de las funciones JavaScript especificadas en el archivo JSLIB.

```
using System.Runtime.InteropServices;

// -----
// Class to wrap the JSLIB file functions
// -----
public static class WebGLPluginJS
{
    // Importing "getDataForURI"
    [DllImport("__Internal")]
    public static extern void getDataForURI(string text);

    // ----- . . . ---
}
```

Estos dos archivos forman parte de la implementación del módulo gráfico desarrollado en Unity.

Para que la comunicación funcione, se ha de desarrollar en el código JavaScript de la página que contiene el módulo la función especificada en el archivo JSLIB. Una vez allí, al estar en el espacio JavaScript de la aplicación Web ya puede usar las variables e instancias de la aplicación, e invocar instancias de la clase *VisualizationComponent*.

Una vez se han iniciado los componentes de la aplicación, el gestor de datos comienza a acceder a la información en el orden adecuado. Desde la clase *VisualizationComponent* se acceden a los diferentes paquetes de datos, que son debidamente procesados, transformados a string en formato JSON para enviarlos al módulo gráfico a través del método *sendData* de su interfaz.

Cuando el módulo gráfico está preparado para recibir más información se lo indica a la instancia de *VisualizationComponent* mediante el método *ReadyForData*. El proceso continúa, hasta que no hay más información que transmitir.

Cuando hay un error durante el procesamiento de datos, o visualización de la información, se comunica a *VisualizationComponent*, y se finaliza el módulo gráfico.

Procedente de *VisualizationComponent* puede llegar una petición de cierre, que conlleva la finalización del módulo gráfico.

En la Figura 67 se muestra un esquema con el proceso de comunicación entre VisualizationComponent y el módulo gráfico, así como la principales operaciones de cada módulo, y el paso de mensajes entre ellos a través de la página principal (Global) que tiene embebida el componente WebGL y tiene acceso a los módulos de la aplicación.

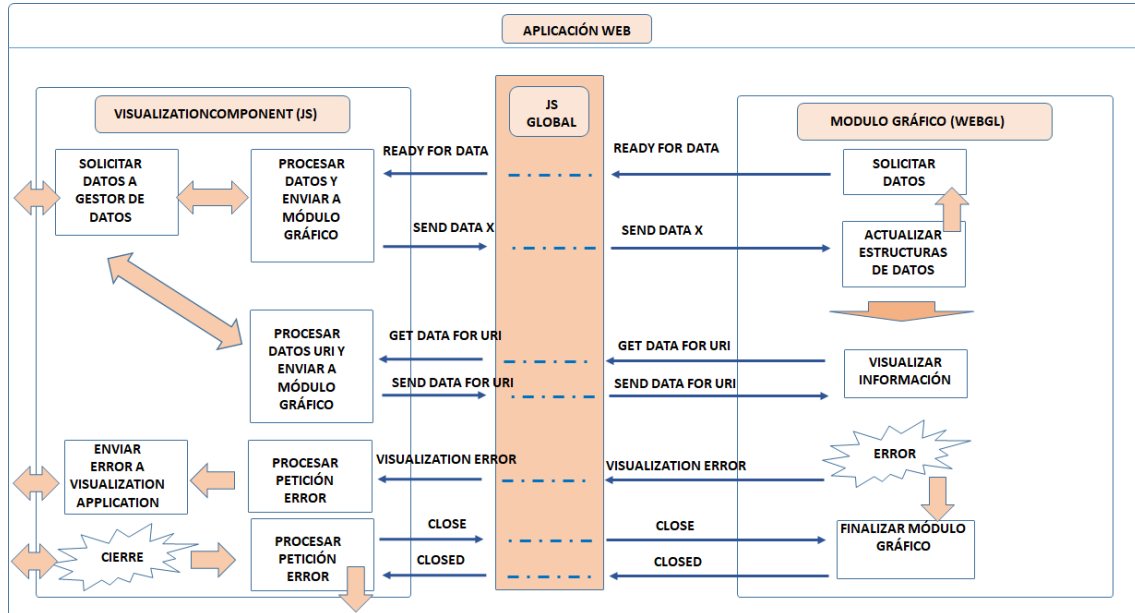


Figura 67 Diagrama de la comunicación entre VisualizationComponent y el Módulo gráfico.

En esta investigación, el desarrollo implementado para el proceso de procesamiento de datos y su comunicación al módulo gráfico es secuencial, pero el sistema está diseñado para optimizar cálculos de las relaciones, e incluso evitar transmitir toda la información, que hace que el sistema esté limitado a una cantidad determinada por los recursos hardware de la máquina que representa la información. Por esta razón se han definido el método *getDataFromRegion*, que permitirían al módulo gráfico solicitar datos de una región concreta, y no de toda la escena. De esta forma, el módulo gráfico podría solicitar la información de los datos próximos a la zona actual del observador, incluso predecir las posibles zonas, en función de su forma de navegar. En caso de cambio de ruta, se cancelarían las peticiones mediante el método *CancelDataFromRegion*. Como se ha comentado en varias ocasiones, estas optimizaciones están fuera de esta investigación y quedarían pendientes de implementar en futuros trabajos, pero el diseño y la definición del API ya lo soportan.

### 8.3. MÓDULO GRÁFICO

El módulo gráfico es el componente que soporta la representación gráfica de la información y permite la interacción con el usuario.

A continuación, se enumerarán los procesos que el módulo va a llevar a cabo. Primero se describirán los necesarios para poder representar la escena y permitir la interacción con el usuario. A continuación, se detallarán el resto de procesos, que son necesarios para llevar a cabo la funcionalidad del módulo, pero que para activarse requieren una interacción del usuario.

En la Figura 68 se muestra un diagrama con los procesos que se ejecutan en el Módulo gráfico y los flujos de comunicación que se producen entre los procesos, así como hacía y desde el exterior del módulo gráfico.



**CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO**

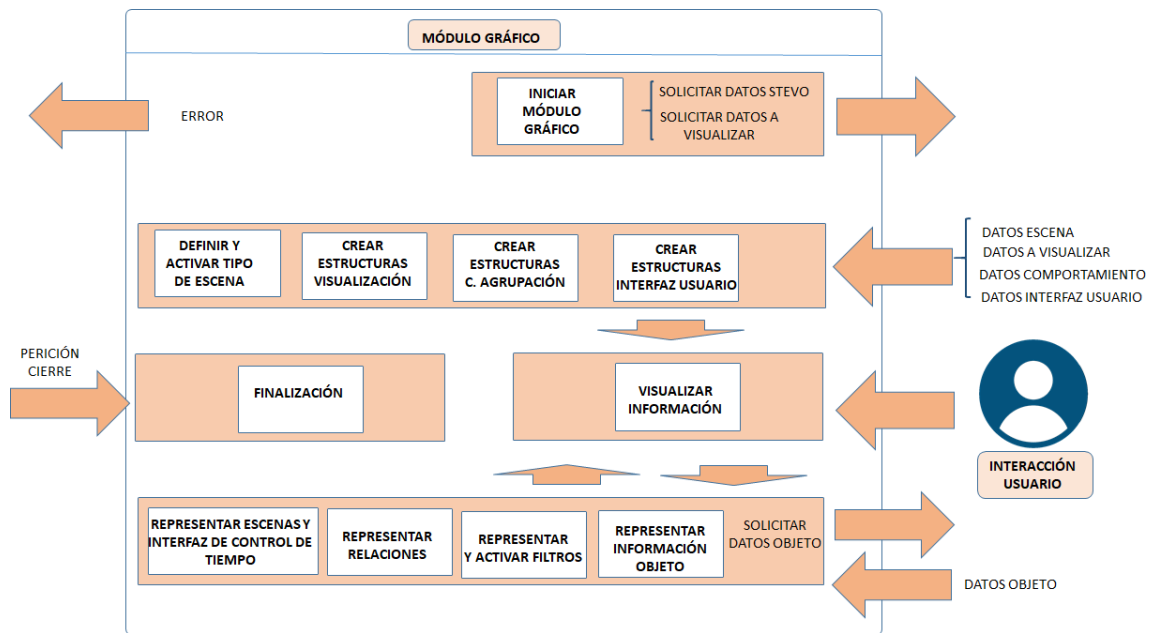


Figura 68 Diagrama con los procesos que se realizan en el Módulo gráfico y los flujos de comunicación que emiten y reciben.

En la Figura 69 se muestra el diagrama con las clases principales del Módulo Gráfico. Estas clases y sus relaciones se citan a continuación, en la descripción de los procesos del módulo.

### 8.3.1. Procesos necesarios para la visualización inicial

#### Iniciar el Módulo gráfico.

El inicio del módulo gráfico implica la instanciación de una clase que gestione el comportamiento del módulo (GraphicModuleManager). Esta se encargará de recibir los datos de forma conveniente, así como arrancar los diferentes procesos para que se represente la visualización, comience la interacción con el usuario y se mantenga la comunicación con el resto de módulos de la aplicación web.

#### Definir y activar el tipo de Escena.

Este módulo se diseñará para ser capaz de representar varios tipos de escenas. De esta forma, cuando reciba información sobre la escena a representar, activará los procesos necesarios para solicitar la información necesaria sobre dicha escena y representar los datos.

Para ello habrá que definir una clase Escena (Scene) que tenga las propiedades y métodos comunes a todas las escenas, así como las especializaciones pertinentes a los tipos de escena que se pueden implementar. En la implementación propuesta se van a desarrollar dos tipos de escena, cada una en dos y tres dimensiones respectivamente.

Por una parte, se desarrollará la escena cartesiana (Cartesian), que es la escena más simple, donde los datos se representan en un sistema de coordenadas cartesiano.

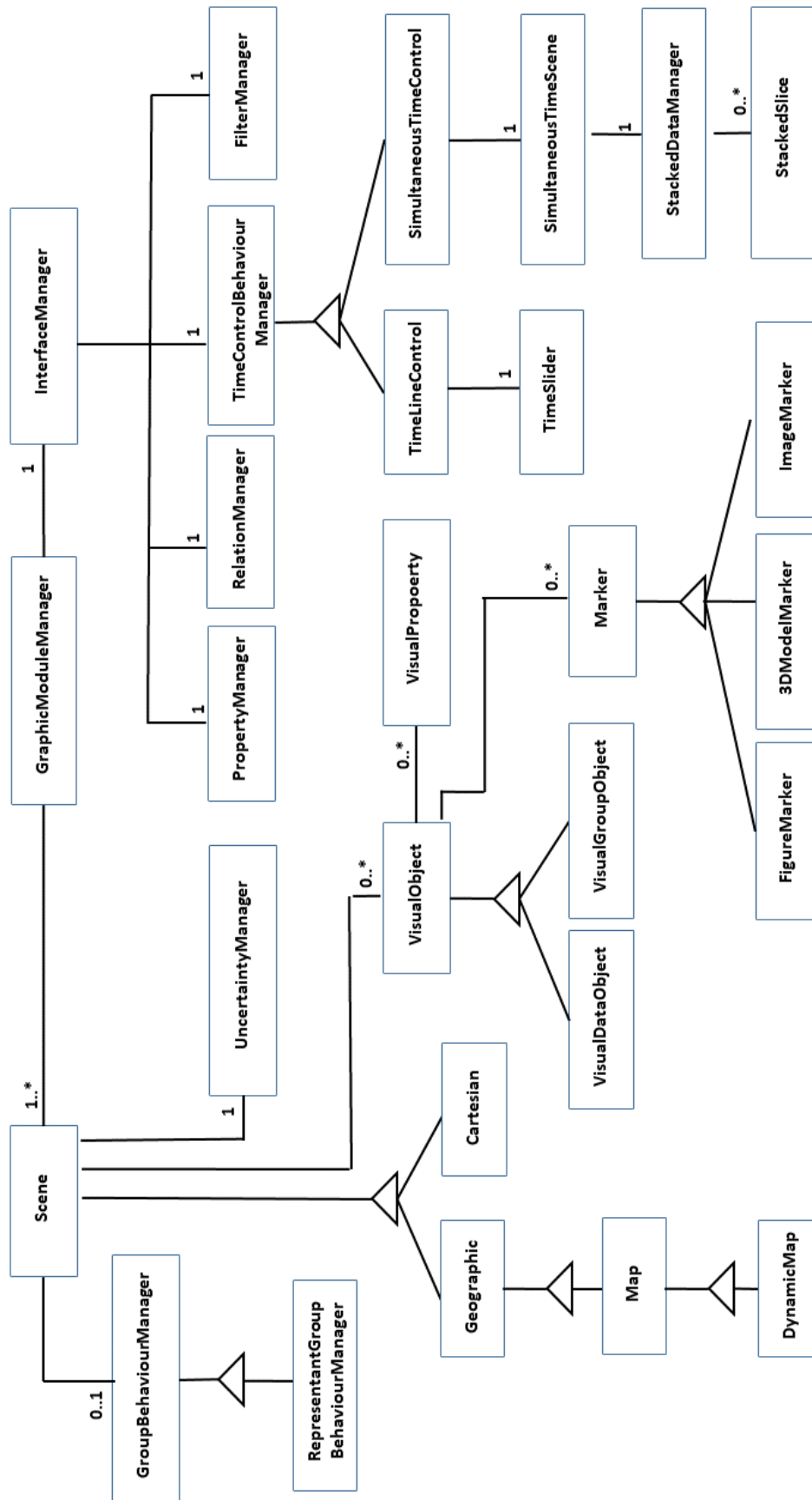


Figura 69 Diagrama de clases con las principales clases del Módulo Gráfico.

Por otra parte, también se implementará la escena Mapa (Map), que extiende la escena Geográfica (Geographic) diseñada para representar datos georeferenciados, en la que los datos se representan sobre un mapa. La escena Map tiene además la especialización dinámica (DynamicMap). En el caso de que sea dinámico, el mapa contempla la opción de niveles de detalle, descritos en el capítulo 6. En este caso, la resolución y datos del mapa se modificarán, conforme el usuario requiera ver más detalle de los contenidos del mapa. Esto implica que el usuario parte de una altura espacial, que determina su separación del mapa, y conforme la altura disminuye, el usuario se acerca más al mapa y esto hace que se actualice su contenido en la zona que se está visualizando.

Dentro de la información de la escena, además de los datos a representar, es muy importante obtener información sobre los diferentes comportamientos especificados en STEVO (agrupación de datos, relaciones, etc.) Esto es importante, ya que puede requerir la solicitud de más datos a los módulos de la aplicación web, además de cambios en la interfaz de usuario y en procesamiento de la información.

En esta investigación estos procesos se van a implementar dentro de este módulo, pero sería mucho más óptimo hacerlo en módulos externos. Al hacerlo en módulos externos se podría realizar un procesamiento en paralelo. Los procesos a realizar en este módulo son los que representan los diferentes comportamientos de agrupación, y de relación definidos en STEVO.

#### Creación de Estructuras de datos de visualización.

En este proceso se crean las estructuras necesarias para representar la información de los datos a visualizar. Estas estructuras consisten en la instanciación de las clases para representar los objetos visuales (VisualObject) y sus especializaciones para datos independientes (VisualDataObject) y representantes de agrupaciones de datos (VisualGroupObject). En la propuesta de desarrollo, también se mantiene información sobre sus propiedades (VisualProperty).

Estos objetos llevan asociadas unas características gráficas vinculadas al concepto Marcador de la ontología STEVO. Por lo tanto, estos objetos estarán asociados a una instancia de la clase Marcador (Marker), que en el caso de la implementación con Unity se trata de una extensión de la clase GameObject, que en el API de Unity es la clase base para representar todas las entidades de una escena. Además de esta clase principal, se desarrollan extensiones para representar figuras FigureMarker, en dos y tres dimensiones, modelos 3D (3DModelMarker), e imágenes (ImageMarker)

A la hora de representar y construir los marcadores es importante conocer cómo se va a representar la incerteza en los datos. Para ello, durante el proceso de carga de datos, cada elemento debe de tener el nivel de granularidad. Este nivel, junto con el número de valores posibles, determina el nivel de incerteza del elemento, por lo que sólo habrá que invocar el gestor de incerteza (UncertaintyManager) a través de la escena, para conocer cómo representar el dato.

#### Creación de Estructuras de datos para representar el Comportamiento de Agrupación.

En STEVO los datos se pueden agrupar por marcador representante y por mapa de calor. En esta investigación, se desarrollará la agrupación por marcador representante. Esta agrupación requiere el diseño de un algoritmo que como mínimo recorra todos los datos para analizar su

proximidad, realizar las agrupaciones, y a su vez, procesar estas agrupaciones, para crear nuevas agrupaciones, y así con tantos niveles como se defina en STEVO.

En el capítulo 4 de esta investigación se describen diferentes tipos de agrupamientos y se enumeran varios de los algoritmos diseñados para este proceso que tiene un coste mínimo de  $O(n) + \sum_{l=1}^{num\ niveles} k/l$

Dado el volumen de datos que puede llegar a tener el ámbito de patrimonio cultural es obvio que el coste temporal de este algoritmo puede ser un problema. Por lo que sería ideal que este procesamiento fuera dinámico, sobre una región y/o conjunto de niveles de la escena.

En esta propuesta de desarrollo, el agrupamiento se ha realizado sobre todos los datos de la escena y se ha utilizado el algoritmo MarkerCluster (Leaver 2012). Este algoritmo básicamente sigue los siguientes pasos.

```
Para todo punto P a agrupar
    Encuentra_Grupo = false;
    i = 0;
    Mientras no (Encuentra_Grupo) Y i < numero_Grupos
        Encuentra_Grupo = (distancia( grupo[ i ], P) < k);
        i = i +1;
    Fin Mientras;
    Si Encuentra_Grupo
        Introduce P en grupo[ i - 1]
    Sino
        Crea grupo[i];
        Introduce P en grupo[i];
    Fin Si;
Fin Para;
```

De esta forma, se crearían varios grupos, que comprenderían todos los puntos de la escena. Después en función de los niveles de profundidad, o zoom, que se hayan definido en la escena, realiza el mismo proceso, pero con los centros de los grupos que se han creado en cada nivel. Así, hasta que llegue al nivel superior, o bien sólo haya un único grupo.

A nivel gráfico hay que tener en cuenta que pueden existir grupos formados por un único punto. En ese caso no se representarán como grupo, sino como puntos independientes.

De este modo, se conocen los grupos que existen en cada nivel de profundidad de la escena, y conforme se va profundizando en la escena, se van mostrando los grupos que lo componen hasta llegar al nivel inferior, que todo son puntos independientes.

## CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

Para llevar a cabo estos procesos la escena tiene acceso a una instancia de la clase gestora del comportamiento de agrupación (GroupBehaviourManager) y su especialización de gestor de agrupación por marcadores representantes (RepresentantGroupBehaviourManager)

[Creación de estructuras de datos para la representación de la interfaz de usuario.](#)

La interfaz del usuario con el módulo gráfico de visualización está compuesta de dos partes principales:

- La representación gráfica que le permitirá navegar por la escena, mediante el teclado, ratón. A través de esta representación podrá acceder a elementos independientes de la escena, y solicitar información adicional sobre ellos.
- Los controles de interfaz que estarán accesibles junto a la representación gráfica. Estos controles, permitirán al usuario realizar operaciones que son fundamentales para cumplir las especificaciones de la ontología STEVO.

La interacción gráfica estará definida por el tipo de escena, y los controles de navegación y otras opciones serán mostradas mediante un botón de ayuda.

Los controles de interfaz van a permitir realizar las siguientes operaciones: Filtrado de datos, Visualizar relaciones y activar los diferentes comportamientos de control de tiempo especificados en STEVO.

La finalidad de este proceso es conocer las diferentes opciones de interfaz que se van a ofrecer al usuario, y conseguir que se puedan representar, además de obtener la información necesaria para que se puedan representar.

Para centralizar la captura de información, así como atender a las peticiones de las clases de Unity para representar los diferentes controles, se desarrollará un gestor de interfaz (InterfaceManager). Este gestor, debe de conocer si van a existir propiedades filtrables para poder mostrar en la interfaz de usuario un botón de filtrar datos, lo mismo con las relaciones y los diferentes comportamientos de control de tiempo. Para ello, desde el gestor de Interfaz se instanciará la clase gestora de las propiedades, que mantiene para cada información. Esta clase mantiene una relación con instancias de la clase gestora de propiedades (PropertyManager) que indica las propiedades que son filtrables y sus posibles valores, y con RelationManager que se encarga de lo propio con las propiedades que son relacionables.

Además, también existe una clase gestora de los filtros activos (FilterManager) para poder activar y desactivar filtros de forma incremental.

El gestor de interfaz también está conectado a la clase gestora de comportamiento de control de tiempo (TimeControlBehaviourManager) y sus extensiones para representar el control de forma simultánea (SimultaneousTimeControl) o secuencial (TimeLineControl). Además de indicar que posibilidades de comportamiento están definidas, y sus características, tienen un API para activar la escena adecuada en el caso de control Simultáneo (SimultaneousTimeScene) y obtener los datos correspondientes a cada ítem de tiempo en los comportamientos de Línea de Tiempo. Este tipo de visualizaciones se describen con detalle en el capítulo 6 de esta investigación. Tal y como se discute en dicho capítulo, la complejidad de las escenas de tiempo simultaneo, requieren la representación de datos en un escenario diferente, donde los datos se reparten a lo largo de varias capas, con el fin de ver en paralelo que datos se corresponden a cada capa.

### 8.3.2. Procesos activados mediante la interacción con el usuario en el Módulo gráfico

#### Representación y activación de filtros

Durante los procesos previos se ha instanciado debidamente la clase FilterManager. Esta puede activar filtros sobre los datos de la escena activa. Esta clase proporciona los datos que están filtrados, así como los diferentes filtros activos. Sin embargo, para mostrar la información el sistema accede a otras clases que se encargan de mostrar todas las propiedades filtrables y los posibles valores de cada propiedad, para que el usuario pueda activar los valores a filtrar.

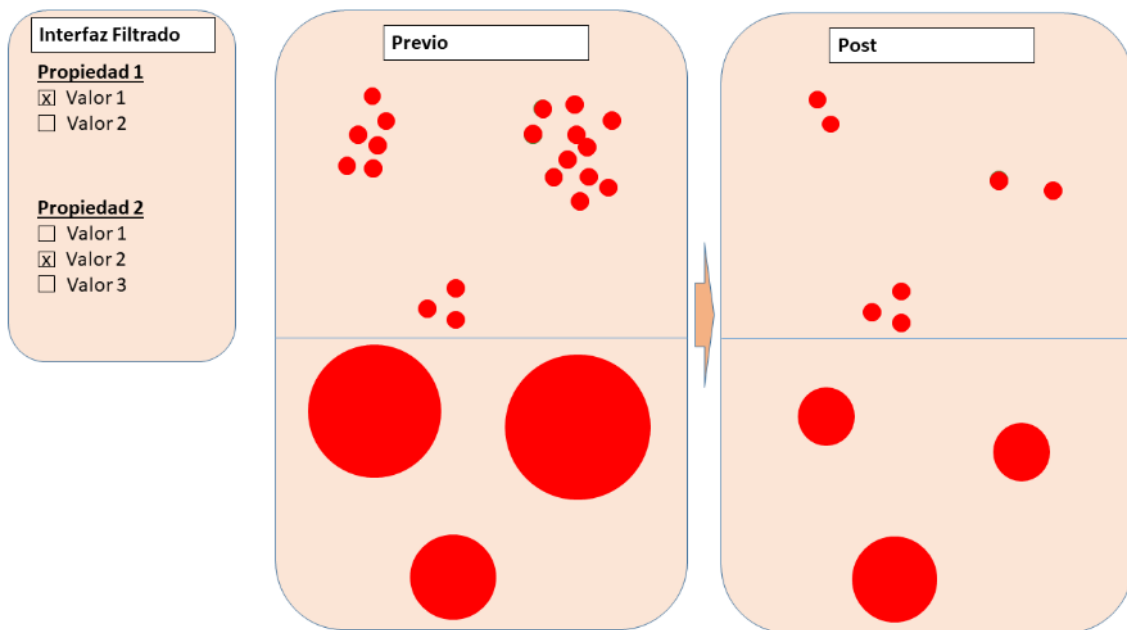


Figura 70 Esquema de funcionamiento de Interfaz de filtrado en Módulo Gráfico.

Una vez se selecciona un valor mediante la interfaz, se aplica el nuevo filtro, sobre los ya existentes de forma incremental, de forma que el resultado final realiza una operación Y (AND) sobre los valores en O (OR) de cada propiedad. Este cambio implica que los datos que no verifican el filtrado dejan de verse y por lo tanto las agrupaciones también cambiarán de tamaño. En la Figura 70 se muestra un esquema de cómo se muestra la interfaz de usuario y cómo afectaría a la visualización de los datos previo al filtrado y posterior al filtrado.

#### Representación de controles y escenas de los comportamientos de control de tiempo.

En la interfaz de usuario se necesita conocer datos sobre los comportamientos de control de tiempo. Esto es necesario para mostrar los controles adecuados en la interfaz de usuario.

Estos controles, una vez activados por el usuario activan los procesos que se describen a continuación:

- Interacción con Línea de Tiempo. Si la escena tiene asociado un comportamiento de control de tiempo secuencial, se adquirirán los datos necesarios para mostrar un control de interacción del tiempo a través de la clase TimeSlider que interactúa con los métodos de FilterManager para mostrar la información pertinente. Esto es debido a que realmente, la barra de control de tiempo lo que realmente realiza es un filtro de los objetos con intervalos de tiempo.

- Interacción con control de tiempo simultáneo. Esta interacción es mucho más compleja desde el punto de vista de la representación gráfica. A través de las especificaciones de control, el módulo gráfico debe de conocer los diferentes intervalos posibles, y en función de lo que elija el usuario, hacer los filtros adecuados para conocer los datos a representar en cada capa asociado a un intervalo. Estos datos se obtendrán a través de la clase FilterManager, y se gestionarán desde la clase gestora de este tipo de comportamiento: StackedDataManager, que obtendrá el fragmento de escena visualizado con la clase CloneVisualizationScene y representará los datos de cada capa mediante la clase StackedSlice.

#### Visualización de datos adicionales del objeto seleccionado.

Este proceso se activa al seleccionar un objeto sobre el que se desea obtener más información. Cuando se visualiza un objeto individual (sin agrupación), se puede clicar con el ratón, o mediante teclado, y ver toda la información sobre dicho objeto en una ventana de información. Para mostrar toda la información del objeto especificada en STEVO, tendrá que conocer las propiedades visuales del objeto y sus valores en el dominio de datos. el gestor del módulo gráfico (GraphicModuleManager), accesible desde la Escena, se encarga de solicitar esta información y transmitir la respuesta.

Esta ventana informativa tiene doble función, por una parte, mostrar datos, pero por otra, dado que se visualizan los valores de las diferentes propiedades, usarla como interfaz de usuario para seleccionar ver los objetos que tienen el mismo valor en dicha propiedad. De esta forma se puede buscar objetos relacionados con el objeto seleccionado de forma sencilla. La filosofía de la interfaz de usuario es encontrar datos mediante la navegación y filtrado, para luego encontrar objetos del conjunto con el mismo valor en una, o varias propiedades del mismo. En la Figura 71 se representa un esquema de esta pantalla y cómo se visualizan las relaciones, con arcos, en objetos independientes y agrupados.

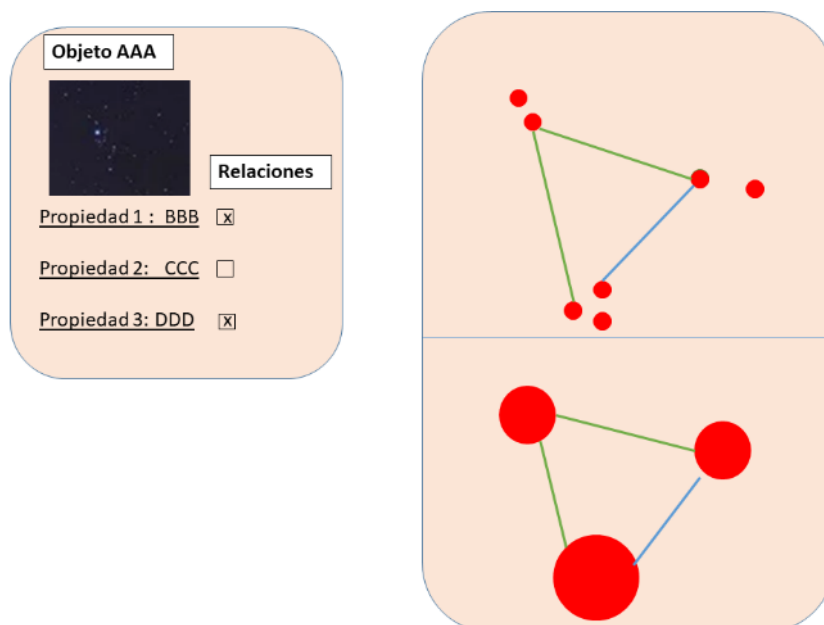


Figura 71 Esquema de pantalla de información de objeto del Módulo gráfico, e interfaz para solicitar datos relacionados.

#### Representación de relaciones.

Este proceso, al igual que el anterior, tiene dos posibilidades de visualización. En función de las especificaciones de la escena, la interfaz de usuario mostrará los controles adecuados para activar los siguientes procesos.

- Visualización de objetos con el mismo valor. Como se ha descrito en el apartado anterior, al solicitar información sobre un objeto de la escena, el usuario puede especificar que desea visualizar los objetos que tienen el mismo valor en determinadas propiedades. Una vez definida, esta información llega al gestor de interfaces (InterfaceManager), que a su vez invoca al gestor de relaciones (RelationManager) para que se encargue de representar en la escena los arcos que unen los objetos relacionados en la escena.
- Visualizador de porcentaje de relaciones. En la ontología STEVO se puede especificar que la escena tenga asociado un comportamiento de visualización de relaciones en Anillo. En este caso, el módulo gráfico tiene que permitir al usuario poder visualizar este tipo de relaciones mediante la interfaz del usuario. De esto se encarga el Gestor de Interfaces (InterfaceManager) ofreciendo el control adecuado en la interfaz. Al activar este control, el mismo control debe de coordinarse con el Gestor de Relaciones, para averiguar los datos de porcentajes de relaciones y mostrarlos mediante la estructura de visualización soportada por la clase RelationshipItemVisualizer. Esta clase se encarga de dibujar una corona circular con tantos sectores como propiedades a visualizar, y su longitud proporcional al porcentaje de objetos relacionados. Esta corona circular se representa circunscribiendo al objeto. Esto hace que introduzca mucho ruido en la escena, aunque por otra parte también introduce mucha información. Lo habitual es usarla temporalmente para detectar una serie de objetos y desactivarla posteriormente.

#### 8.3.3. Procesos de finalización

Los procesos de finalización se invocan mediante el gestor de módulo gráfico por dos motivos. Una solicitud de cierre invocada desde la aplicación, o bien, por un error generado dentro del módulo gráfico.

En ambos casos se tienen que invocar los procesos que liberan los recursos utilizados hasta el momento para la visualización.

En caso de que el error haya sido generado por el módulo gráfico, el gestor gráfico informará del error a la aplicación.

Una vez finalizada la destrucción de procesos, se comunica a la aplicación web que la ejecución del módulo gráfico ha terminado.

#### 8.4. CONCLUSIONES

En este capítulo se ha diseñado un marco de trabajo que permite acceder al contenido de un grafo de conocimiento con datos de patrimonio cultural y visualizar el contenido de este grafo, o parte de él, mediante una aplicación web.

Se ha definido la arquitectura de esta aplicación web, y su integración en una web, así como la comunicación con el grafo de conocimiento, la ontología STEVO y el resto de módulos web.

Se ha expuesto una propuesta de implementación de una aplicación web siguiendo las directrices de la arquitectura. Debido a las limitaciones de esta investigación, existen elementos



CAPÍTULO 8. DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE VISUALIZACIÓN WEB DEL CONTENIDO DE UN GRAFO DE CONOCIMIENTO, ESPECIFICADA CON LA ONTOLOGÍA STEVO

de STEVO que no se han podido implementar, así como mejoras sustanciales en la implementación que permitirían visualizar los datos de forma más rápida, incluso manejar un mayor volumen de datos. Sin embargo, la propuesta de aplicación alcanza los objetivos propuestos y está diseñada de forma que las mejoras detectadas afecten lo mínimo posible a los desarrollos realizados.

En la arquitectura de la aplicación existe un módulo gráfico que permite representar escenas gráficas bidimensionales y tridimensionales en una página web. Siguiendo los trabajos realizados en el capítulo 5 de esta investigación, se han analizado diferentes tecnologías que permiten desarrollar código en WebGL, dada su funcionalidad, integración en aplicaciones web y soporte se concluyó en dicho capítulo que era la mejor opción. Estas tecnologías se han analizado y se ha decidido realizar los desarrollos de este módulo gráfico usando la tecnología Unity.

Se han expuesto los procesos que intervienen, la comunicación entre ellos, así como las principales clases, y los diagramas de clases tanto de la aplicación web, como del módulo gráfico.

## SECCIÓN IV. OTROS RESULTADOS

## CAPÍTULO 9. APLICACIÓN DE LOS RESULTADOS DE LA INVESTIGACIÓN

### 9.1. RESUMEN

En este capítulo, se van a presentar los resultados del marco de trabajo diseñado e implementado en esta investigación sobre dos proyectos informáticos. Estos proyectos están basados en datos de patrimonio cultural soportados por grafos de conocimiento.

El primero es el proyecto SILKNOW, que maneja datos de tejidos de seda del último milenio relacionados con la ruta de la seda en Europa. En uno de los resultados de este proyecto, que es el portal web, tenía que desarrollarse un mapa espacio temporal. Un mapa que tenía que ser capaz de visualizar sobre el mapa los diferentes tejidos resultados de una búsqueda sobre el portal web. Además de visualizarlos geográficamente, debía de permitir realizar filtrados, así como ver sus propiedades, las relaciones existentes entre ellos y evaluar la fluctuación de datos en el tiempo, tanto de forma simultánea como de forma lineal.

El proyecto ya está terminado y en este capítulo se expone el trabajo realizado para implementar dicho mapa espacio temporal, basándose en la herramienta diseñada e implementada en esta investigación. Este trabajo, principalmente consiste en la creación de instancias de conceptos de la ontología STEVO, para que la herramienta pueda procesar este contenido y visualizar los datos en la aplicación web incluida en el portal, que es el resultado software de esta investigación.

El segundo es el proyecto de l'Arxiu del Disseny de València. Este proyecto está relacionado con los documentos y objetos de patrimonio cultural del diseño valenciano. En este proyecto hay que desarrollar una aplicación web que permita realizar filtrados y visualizar los datos de la información contenida en l'Arxiu. Además, tiene que incluir una herramienta gráfica denominada Mapa de Producto, que permita visualizar en dos y tres dimensiones los resultados de variables relacionadas con un conjunto de datos de l'Arxiu, que sean el resultado de una búsqueda realizada sobre el portal web.

El proyecto aún está en desarrollo por lo que no se tiene permiso para mostrar la ontología con las instancias de los conceptos de STEVE, así como datos técnicos del mismo. Sin embargo, sí que se ha obtenido el permiso para mostrar información del proyecto y algunas capturas de pantalla del prototipo del Mapa de producto, que está desarrollándose con la herramienta resultado de esta investigación.

### 9.2. APLICACIÓN DE LOS RESULTADOS EN EL PROYECTO SILKNOW

#### 9.2.1. Proyecto SILKNOW

El proyecto SILKNOW (Silk heritage in the Knowledge Society: from punched cards to bigdata, deep learning and visual/tangible simulations) (Portalés, Sebastián et al. 2018), financiado por la Unión Europea, dentro del programa de investigación e innovación, con el código 769504.

El objetivo es desarrollar un sistema informático diseñado para mejorar la comprensión actual del patrimonio cultural europeo relacionado con los tejidos realizados con seda. Este sistema parte de una tarea base, consistente en el diseño e implementación de un grafo de conocimiento basado en el modelo CIDOC-CRM. Este grafo, ha de integrar la información relativa a estos tejidos que se encuentra digitalizada en la actualidad.

El grafo de conocimiento es un resultado del proyecto, que como todos los del proyecto, es abierto y de acceso público. Además de este, el resto de resultados del proyecto se dividen en dos grandes grupos, uno orientado a introducir datos en el grafo de conocimiento, y otro conjunto de herramientas que permitirán buscar y visualizar la información del mismo.

El grupo dirigido a la introducción de información en el grafo de conocimiento está compuesto por las siguientes herramientas:

- Un sistema de análisis semántico de texto. Este sistema utiliza como base las reglas del grafo de conocimiento, la información del mismo, y otras fuentes, como WikiData, y el uso de herramientas de análisis morfológico, sintáctica y de desambiguación. Su función es analizar el contenido de un texto procedente de las fuentes de datos digitalizadas para actualizar e introducir nuevos datos en el grafo de conocimiento.
- Un sistema de análisis de imágenes inteligente basado en redes neuronales. Este sistema permite analizar las imágenes digitalizadas de los tejidos y utilizar la información obtenida para actualizar el grafo con datos sobre los motivos, colores, técnicas de tejido, llegando incluso a obtener, en ocasiones, datos sobre la época y el lugar dónde se confeccionó, con una alta probabilidad.
- Una herramienta de mapeo, que permite conectar una fuente de datos digitalizada con las herramientas anteriores y actualizar el grafo de conocimiento.
- El desarrollo de un tesoro multilingüe sobre las técnicas de tejido de seda empleadas en Europa (León, Gaitán et al. 2020). El tesoro, extiende a nivel jerárquico el tesoro de Getty AAT (Getty 1997a). El tesoro tiene una elevada complejidad de desarrollo, ya que aplica la traducción de términos usados en contextos muy específicos y con varios siglos de antigüedad.

Entre los resultados que tienen la finalidad de mostrar la información del grafo al usuario y permitir interactuar con esta, se han desarrollado las siguientes herramientas:

- Un buscador web que permita a usuarios con diferentes perfiles, tanto de conocimiento, como de intereses, obtener la información sobre estos tejidos que forman parte del patrimonio cultural. En este portal web, tienen que desarrollarse e integrarse las siguientes herramientas:
- Un telar virtual, que permita la visualización interactiva en tres dimensiones de la aplicación de diversas técnicas de tejido que representan cómo ha sido producido. Esta herramienta permite ver, partiendo de una imagen, y de la información como se ejecutan estas técnicas, los diferentes hilos que se han usado y como se entrelazan entre sí.
- Un mapa espacio temporal interactivo que muestra la distribución geográfica y temporal de los objetos resultantes de realizar una búsqueda mediante el buscador web. Este mapa debe de permitir realizar filtrados adicionales, navegación espacial, visualizar las relaciones entre los diferentes objetos y analizar la fluctuación en el tiempo de esta información.

En la Figura 72 se muestra un diagrama con los diferentes módulos y resultados del proyecto SILKNOW, estructurado a tres objetivos principales: OB1: Obtener información, OB2: Introducir la información en un grafo de conocimiento, OB3: Visualizar e interactuar con la información.

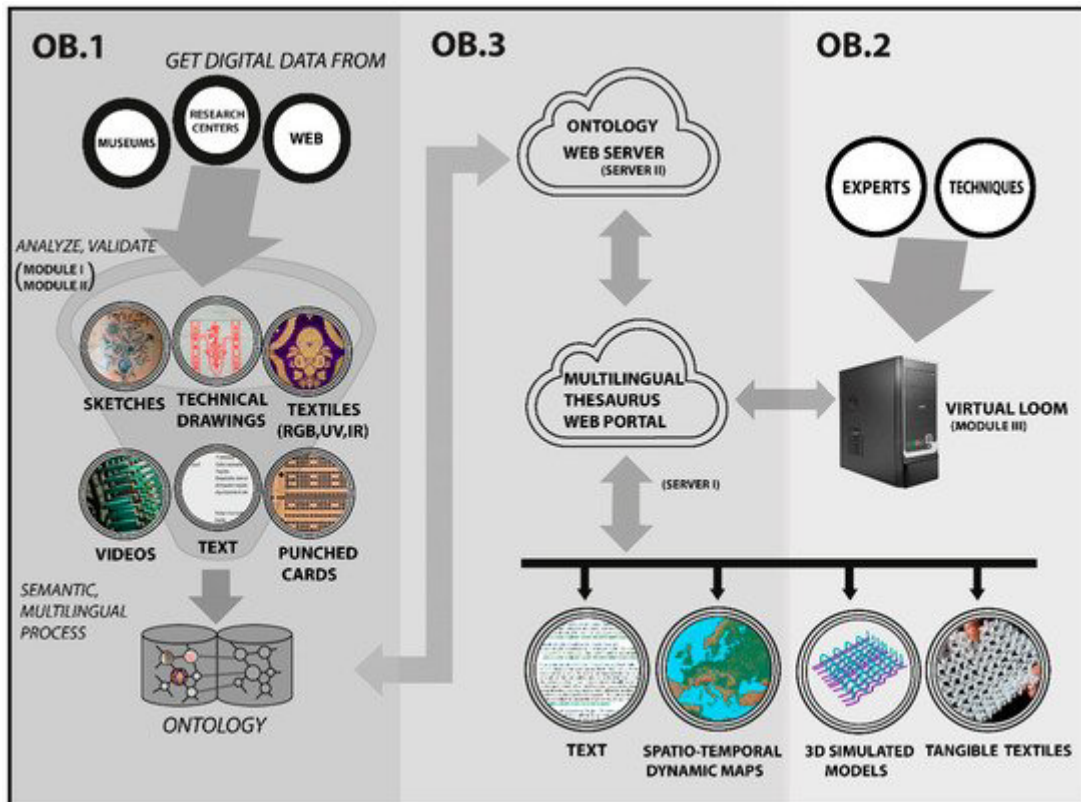


Figura 72 Diagrama con los resultados del proyecto SILKNOW. Fuente: (Portalés, Sebastián et al. 2018)

En el proyecto SILKNOW hay dos herramientas de visualización de datos, que toman como fuente de datos el contenido de un grafo de conocimiento con datos de patrimonio cultural, el telar virtual y el mapa espacio temporal interactivo.

En el telar virtual el objetivo es visualizar la aplicación de una técnica de tejido sobre la imagen de un objeto. En este caso no tiene mucho sentido aplicar la ontología STEVO, ya que no se van a visualizar datos de patrimonio cultural donde su posición en la escena dependa de los datos del grafo de conocimiento. En esta herramienta, el grafo de conocimiento sirve para extraer la imagen del objeto, así como los datos asociados a la misma: colores, motivos, etc., y la técnica utilizada para su tejido. Con esa información el sistema muestra un tejido de forma tridimensional. En la Figura 73 se muestra en la parte superior la imagen de un tejido que sería una de las entradas del telar virtual, tras ser corregidos aspectos como la inclinación, o fragmentos deteriorados, se selecciona un fragmento y la herramienta muestra la distribución de los hilos en un entorno tridimensional en la parte inferior.

La otra herramienta de visualización de SILKNOW, el mapa espacio temporal, tiene como objetivo la visualización de varios objetos de patrimonio cultural en una escena espacial, representada por un mapa, donde además se tiene que visualizar su fluctuación en el espacio. Esta herramienta encaja perfectamente con el objetivo del marco de trabajo de visualización de datos, mediante una aplicación web, diseñado y desarrollado en este trabajo.

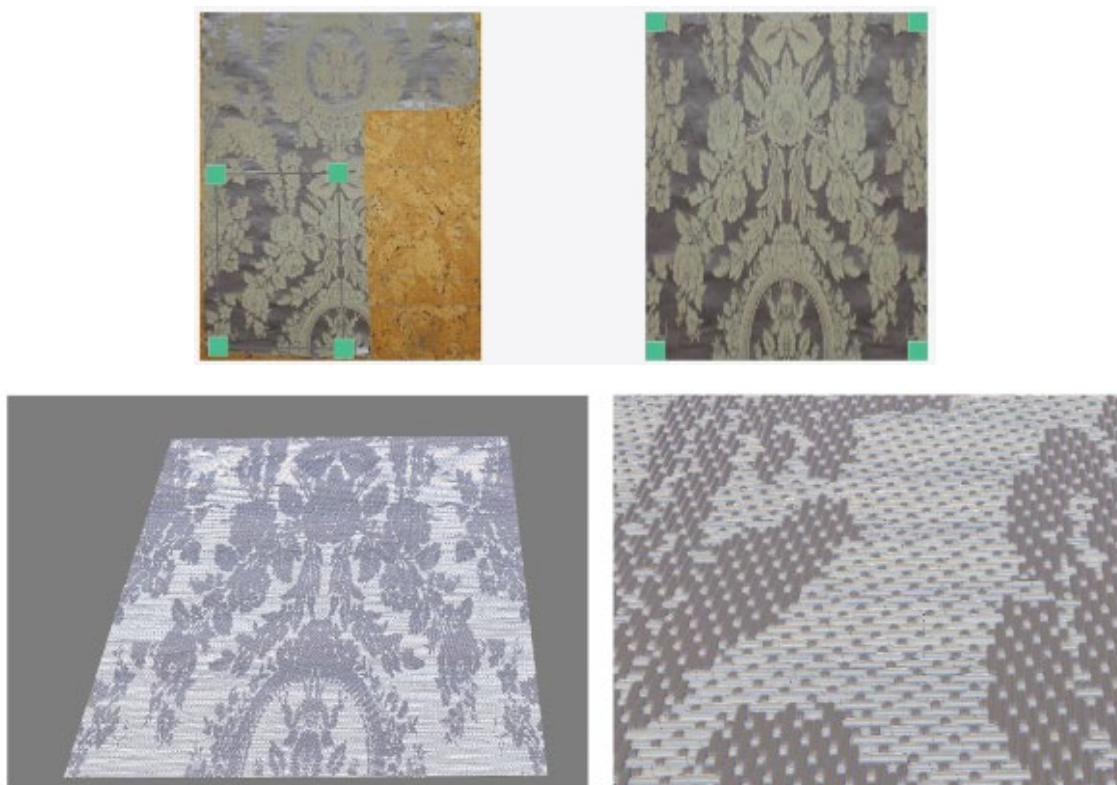


Figura 73 Captura de pantalla de los resultados del telar virtual del proyecto SILKNOW: Fuente: (Portalés, Pérez et al. 2021)

En el mapa espacio temporal de SILKNOW, los datos a representar son tejidos realizados en seda, que además pertenecen a diferentes tipos, en función del empleo que se hacía de estos tejidos, por ejemplo: ropa, decoración del hogar, decoración de muebles, religioso, etc. Todos los objetos tienen una localización geográfica que indica su producción, además de una referencia temporal, tomando como unidad de referencia el siglo. Como es habitual, existen datos donde hay incerteza, tanto a nivel geográfico, como temporal, además de una presencia de incerteza importante debida a la granularidad, principalmente espacial, aunque también temporal.

La totalidad de objetos del grafo de conocimiento de SILKNOW era en enero de 2022 de 37832. Dado este número de datos es necesario la realización de agrupamientos para su visualización. En este caso, se prefirió un agrupamiento por clusters, en los que un objeto gráfico representa a un conjunto de elementos, y conforme el usuario interactúe con el mapa se vayan separando estos grupos, hasta llegar a un nivel que permita la visualización de objetos de forma individual.

Otro de los requisitos del proyecto sobre la herramienta es la realización de filtros dinámicos sobre el mapa. Aunque el contenido del mapa es el resultado de una búsqueda inicial, una vez visualizados se ha de ofrecer la posibilidad de realizar nuevos filtros sobre los datos del mapa. Por ejemplo, se puede requerir la visualización de objetos que usen oro en sus hilos, y una vez visualizados, desde el mapa espacio temporal, ver sólo los que se hayan producido en Italia en el siglo XVI.

También es una funcionalidad requerida el hecho de que se pueda filtrar en cada momento y de forma sencilla, por el siglo en que fueron producidos los objetos. De esta manera, debe de ser posible ver sobre la misma pantalla el cambio de datos de un siglo a otro. El objetivo de este

requisito es el de poder ver la variación espacial de los datos, casi como si de una película se tratara con el fin de poder inferir razones, o comportamientos que causen esta variación.

Con el fin de reforzar este último objetivo aún más, el mapa espacio temporal debe de poder visualizar de forma simultanea los datos de diferentes periodos de tiempo en una región espacial. Este requisito, permite ver esta variación espacial de forma directa. Por último, se tienen que poder mostrar en el mapa espacio temporal, en cada momento, los objetos del mapa que tengan el mismo valor para una propiedad dada. Esta asociación, o relación quedará marcada gráficamente, enlazando los objetos de una forma concreta, mediante arcos, o líneas. Además, el mapa espacio temporal, debe de permitir mostrar ver que porcentaje de objetos de los que se están visualizando tienen el mismo valor en sus diferentes propiedades. Con esta funcionalidad se puede ver de forma sencilla objetos que siguen comportamientos anómalos. Por ejemplo, si en el mapa, todos los objetos tienen un porcentaje similar de objetos en sus propiedades (material usado, técnica, etc.), pero hay uno que tiene un porcentaje mucho más bajo, o mucho más alto, no sigue el patrón habitual, y puede ser interesante ver el porqué de este comportamiento.

Todos estos requisitos de visualización son representables mediante la ontología STEVO, y por lo tanto podrían visualizarse con el marco de trabajo desarrollado en este trabajo. En la Figura 74 se muestra una captura del mapa espacio temporal de SILKNOW, desarrollado con los resultados de este trabajo, donde se describen las partes principales de la interfaz de usuario del sistema de visualización.

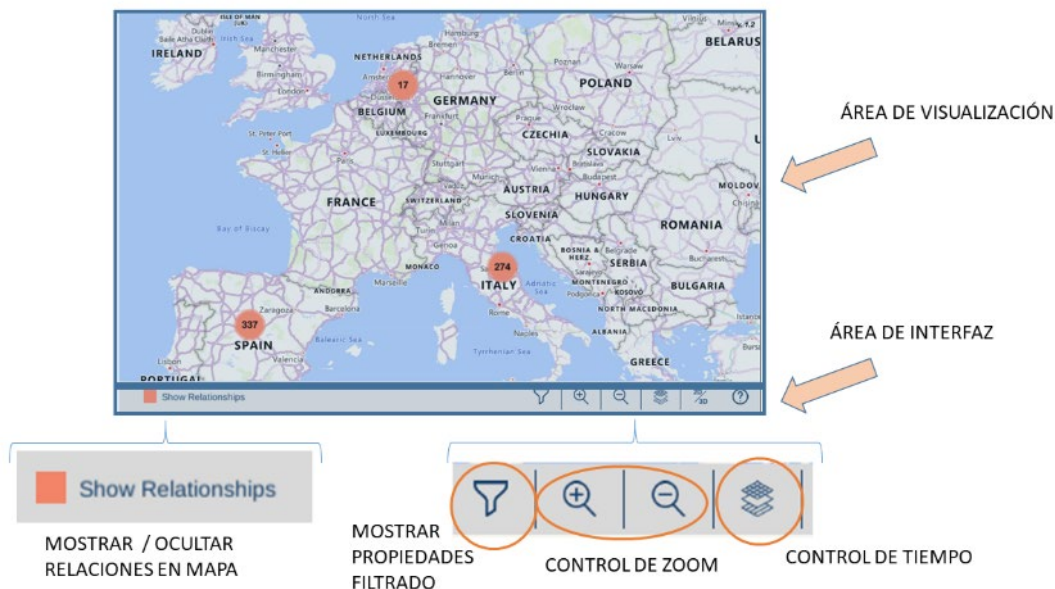


Figura 74 Diagrama sobre captura de pantalla del mapa espacio temporal para indicar las diferentes áreas del componente. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

A continuación, se expondrán las consideraciones que se han seguido para introducir instancias en la ontología STEVO que permitan representar esta visualización. Para ello, se determinará el tipo de escena, agrupamiento, los diferentes conceptos visuales, así como sus propiedades, y por último el comportamiento.



El grafo de conocimiento del proyecto SILKNOW es público y existe un portal web con varios puntos de acceso a su contenido, desde un punto SPARQL, hasta un buscador de facetas. Puede consultarse en la dirección <http://data.silknow.org>.

#### 9.2.2. Instanciación de STEVO

Para definir como instanciar la ontología STEVO con el fin de representar un sistema de visualización hay que determinar los siguientes aspectos:

- Cómo acceder al contenido del grafo de conocimiento.
- Definir la escena a representar.
- Definir los conceptos visuales que se van a representar.
- Definir las propiedades visuales de los conceptos seleccionados y sus características, es decir, cuales determinan las posiciones espaciales y temporales, y como, además de cuáles podrán visualizarse por el usuario, así como por cuales podrá filtrar y relacionar con otros objetos.
- Definir la representación gráfica de los conceptos.
- Especificar si tiene que utilizarse una agrupación de los datos y su representación gráfica.
- Definir si hay que instanciar el resto de comportamientos definidos por STEVO (relación, visualización temporal, etc.)

#### Acceso al grafo de conocimiento.

El grafo de conocimiento en SILKNOW está soportado por la plataforma Virtuoso. Por lo que a nivel interno del proyecto se tiene acceso directo a un punto de acceso SPARQL directo contra el servidor que contiene el grafo de conocimiento. Además, para permitir el acceso a aplicaciones externas, de forma pública, se tiene acceso a una serie de servicios web generados mediante la herramienta GRLC (Meroño-Peñuela, Hoekstra 2016). Esta herramienta, transforma consultas SPARQL en servicios web RESTful de forma automatizada. En el rendimiento del sistema se observó que el rendimiento de estos servicios web no es tan bueno como el directo sobre el punto SPARQL de Virtuoso.

En cualquier caso, el acceso al contenido del grafo de conocimiento de SILKNOW se hará mediante servicios web especiales, o mediante la ejecución de consultas SPARQL.

#### Definición de la escena.

La escena del mapa espacio temporal de SILKNOW debe de estar basada en un mapa. Sobre ese mapa se deben de representar los datos en el punto adecuado. Además, conforme el usuario se acerque a una zona, esto puede interpretarse como que hace zoom, o que está descendiendo, el contenido del mapa debe de actualizarse y mostrar texturas con más resolución y dónde se aprecian más detalles, como carreteras, ríos, etc.

Por lo tanto, habrá que instanciar el concepto Scene (Escena), con la instancia que se denominará *“sceneSilknow”*

Esta escena, tendrá que usar las siguientes representaciones:

- *Map*. Esta relación es obvia, ya que es una escena donde la base especial es un mapa.
- *Link Diagram*. Los objetos se van a enlazar entre sí, ya que se tienen que visualizar las relaciones gráficamente.



- *TimeLine*, para poder filtrar los datos mediante una línea de tiempo.
- *Time Layer*, para poder visualizar los datos en un mismo espacio, pero mostrando varias capas del mismo espacio, donde cada capa estará relacionada con un intervalo de tiempo diferente.

Como el mapa ha de partir de una visión mundial y descender a una visión que se aprecien las calles principales de las ciudades, sin necesidad de llegar a un detalle más alto, se estima inicialmente definir 12 niveles de zoom para asegurar un cambio aceptable por el usuario, pero que tampoco sature demasiado el sistema. De los cuales, los 3 últimos niveles no serían accesibles en la navegación, ya que no hace falta llegar a un nivel máximo de detalle. Estos valores pueden ser ajustados posteriormente. Debido a esto, los valores de las propiedades nivel de zoom máximo y mínimo tendrán los 12 y 3 respectivamente.

En la Figura 75 se representa un diagrama con las relaciones y propiedades básicas de la instancia “sceneSILKNOW” que representará la escena del mapa espacio temporal en el proyecto SILKNOW. Más adelante se añadirán más propiedades y relaciones, conforme se vayan instanciando más conceptos en la ontología STEVO relacionados con el mapa espacio temporal de SILKNOW.

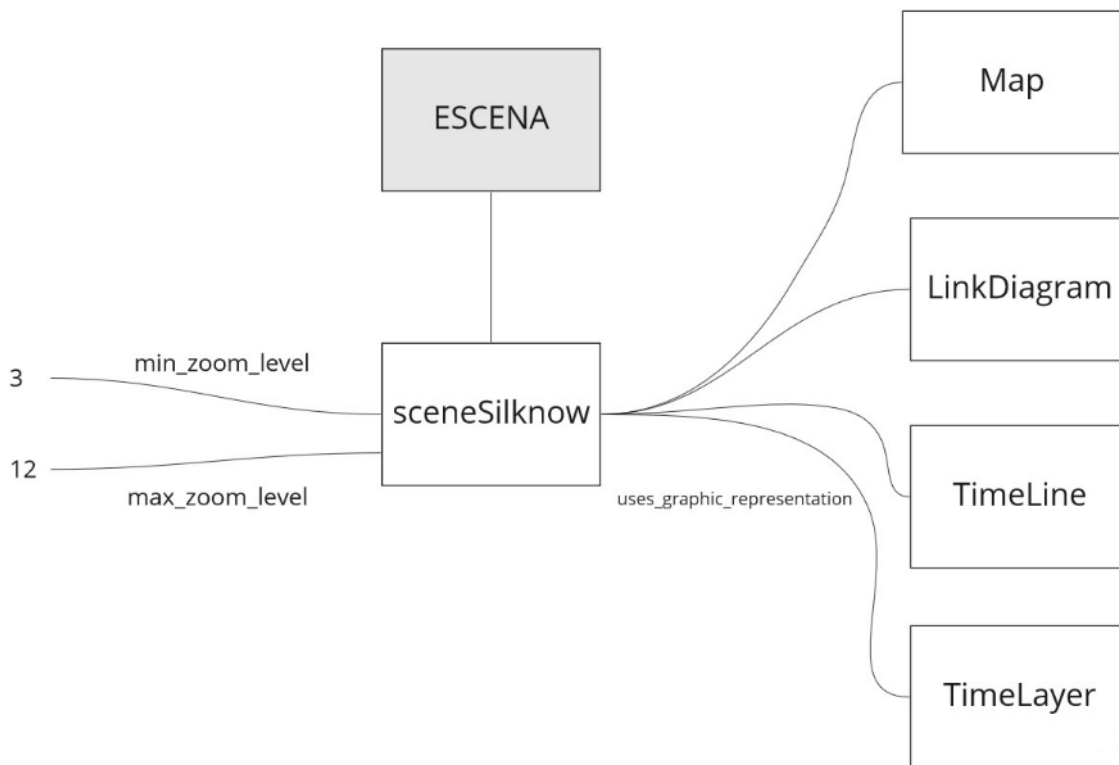


Figura 75 Diagrama con relaciones y propiedades de instancia de Escena en STEVO de la escena de SILKNOW

#### Definición de los conceptos visuales.

En el mapa espacio temporal del proyecto SILKNOW se tienen que representar tejidos.

Los tejidos en el grafo de conocimiento de SILKNOW, al estar basado en el modelo CIDOC-CRM, podrían ser instancias, o extensiones del concepto *E22\_Man-made\_Object*, del modelo. En el caso del proyecto SILKNOW se optó por instanciar el concepto directamente.

Entonces, se instanciará el concepto *Visual\_Concept* de la ontología STEVO, con la instancia *visualTextile*, la cual tendrá en la propiedad “*represents domain concept*” el valor:

"http://erlangen-crm.org/current/E22\_Human-Made\_Object"^^xsd:anyURI

Este valor corresponde al concepto E22\_Man-made\_Object en la implementación del modelo CIDOC-CRM con la ontología de la Universidad de Erlangen.

Para representar este concepto visual, hay que instanciar el concepto *Visual Concept*. La instancia tendrá el label *visualTextile.*, será representado por la instancia de escena (*Scene*) *sceneSilknow.*

#### Definición de las propiedades visuales.

Las propiedades de los tejidos que se tienen que visualizar son los materiales de los que está hecho el objeto, la técnica de tejido con la que fue confeccionado y tipo o categoría del objeto. Además, debido a la incerteza de los datos, también se podrá visualizar el tiempo y el lugar de producción. Esto es debido a que en SILKNOW se decidió representar la incerteza con un marcador especial y representar un único objeto. De esta forma, si se desea saber los diferentes datos del objeto, no hay otra opción que poder visualizar el valor de estas propiedades.

En la Figura 76 se muestra una captura de pantalla del mapa espacio temporal de la web del proyecto SILKNOW. Este mapa ha sido generado con el marco de trabajo implementado en esta investigación. Al existir propiedades por las que se puede filtrar, el sistema muestra un botón de filtrado en la interfaz (ver parte inferior de la figura). Al activar este botón se muestra una ventana con las diferentes propiedades que son filtrables (propiedad “es\_filtrable” del concepto Propiedad\_Visual) y sus posibles valores del conjunto de datos que se visualizan.

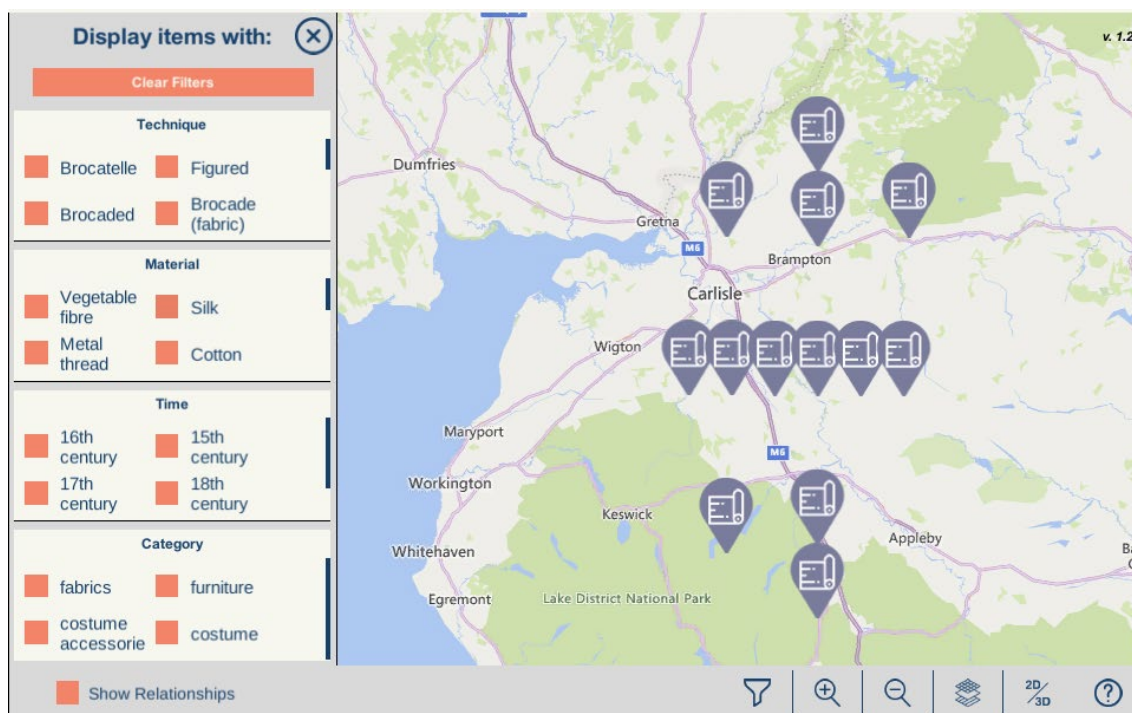


Figura 76 Interfaz de usuario de mapa espacio temporal mostrando las propiedades por las que se puede filtrar y sus posibles valores. Fuente: ADASILK (SILKNOW Consortium 2022)

En la Figura 77 se muestra una captura de pantalla del mapa espacio temporal de SILKNOW, en el que, tras seleccionar un objeto, se muestra información de las diferentes propiedades visuales del mismo que son visibles (propiedad “es\_visible” del concepto Propiedad\_Visual). En esta

ventana, también se muestra un control de tipo check, sobre aquellas propiedades que son relacionables (Propiedad “es\_relacionable” del concepto Propiedad\_Visual) .

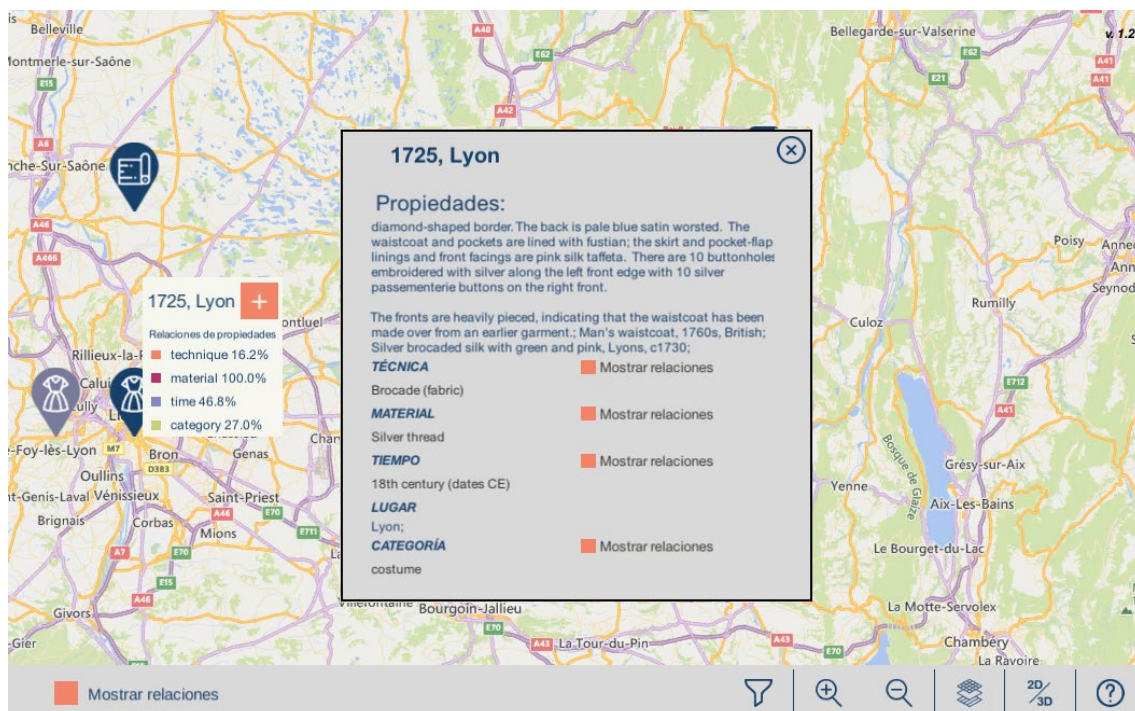


Figura 77 Captura de pantalla con ventana emergente con información del objeto representado en el mapa espacio temporal de SILKNOW. Fuente: ADASILK (SILKNOW Consortium 2022)

### Material del objeto

En el dominio de datos de SILKNOW, para conocer el material de que está hecho el tejido, se aplica el modelo de CIDOC-CRM. Por lo tanto, es necesario instanciar el concepto E12\_Production relacionado con el tejido (E22\_Man-made\_Object), a través de la relación P128\_has\_produced. En dicha instancia, se puede obtener el valor del material empleado a través de la relación P126\_employed, que enlaza con una, o varias instancias del concepto E57\_Material. La instancia de E57\_Material tiene la información sobre el material, o materiales empleados para elaborar el tejido.

En la ontología STEVO para representar esta propiedad hay que instanciar el concepto *Visual Property* con la instancia *visualTextileMaterial*. En los requisitos del proyecto se indicaba que en el mapa espacio temporal se mostrará la información del material del objeto, además de poder filtrar por esta propiedad y mostrar las relaciones con otros objetos que tengan el mismo valor en esta propiedad. Las propiedades y relaciones de esta instancia se muestran en la Tabla 35.

La instancia del concepto Query de la ontología STEVO, queryForTextileMatyTec, referenciada por visualTextileMaterial y por visualTextileTechnique, que se detalla en el apartado siguiente, tiene en su expresión una Query en SPARQL, que permite obtener los valores del material y de la técnica del objeto. Las propiedades de la instancia queryForTextileMatyTec se muestran en la Tabla 36.

Tabla 35 Propiedades y relaciones de la propiedad visual *visualTextileMaterial*

| Tipo      | Nombre                   | Valor                          |
|-----------|--------------------------|--------------------------------|
| Propiedad | is_relatable             | True                           |
| Propiedad | Is_filtered              | True                           |
| Propiedad | Has_name                 | "material"                     |
| Propiedad | Uses_query_variable_name | "materialLabel"                |
| Relación  | Represented_by_color     | blueColor (Color)              |
| Relación  | Value_per_query          | queryForTextileMatyTec (Query) |
| Relación  | Is_domain_property       | visualTextile (VisualConcept)  |

Tabla 36 Propiedades de la instancia de Query, *queryForTextileMatyTec*

| Propiedad      | Valor   |
|----------------|---|
| Has_expression | <pre> SELECT distinct ?techniqueLabel ?materialLabel WHERE {   ?s     rdf:type     &lt;http://erlangen-crm.org/current/E12_Production&gt; .   ?s     &lt;http://erlangen-crm.org/current/P108_has_produced&gt;     this .   ?s     &lt;http://erlangen-crm.org/current/P32_used_general_technique&gt;     ?p .   ?p     &lt;http://www.w3.org/2004/02/skos/core#prefLabel&gt;     ?techniqueLabel   ?s     &lt;http://erlangen-crm.org/current/P126_employed&gt;     ?q .   ?q     &lt;http://www.w3.org/2004/02/skos/core#prefLabel&gt;     ?materialLabel }</pre> |

En la Figura 78 se representa un diagrama con los conceptos de la ontología STEVO involucrados en la representación de los conceptos visuales y sus propiedades visuales requeridos por el proyecto SILKNOW.

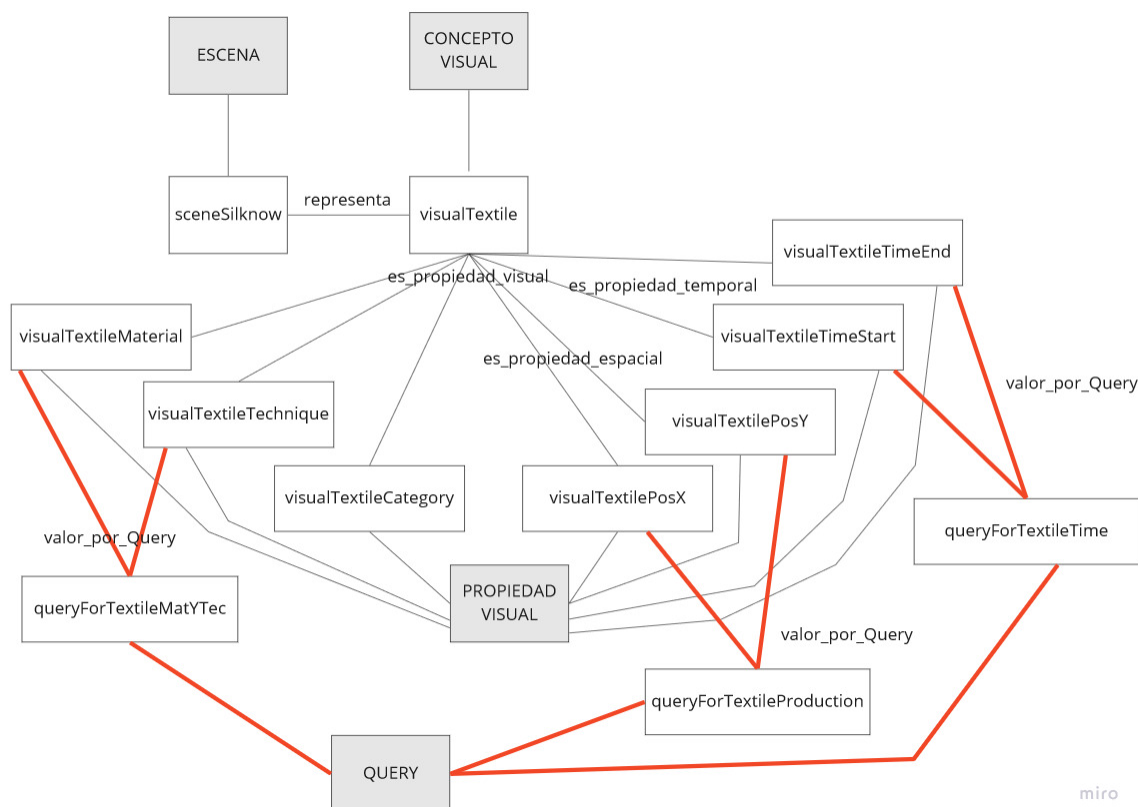


Figura 78 Diagrama con conceptos (fondo gris) e instancias de la ontología STEVO para representar los conceptos visuales y sus propiedades visuales para representar los datos del proyecto SILKNOW

#### Técnica del objeto.

Con la técnica, al igual que el material, en SILKNOW se aplica el modelo de CIDOC-CRM. Al igual que en el material, es necesario instanciar el concepto E12\_Production, y en esta instancia, mediante la relación P32\_used\_general\_technique, se accede a una o varias instancias del concepto E55\_Type. La instancia de E55\_Type se usa en CIDOC-CRM para enlazar con términos de tesauros. En este caso, se enlaza con el tesauro de SILKNOW que clasifica las diferentes técnicas de tejido con seda.

En la ontología STEVO para representar esta propiedad hay que instanciar el concepto *Visual Property* con la instancia *visualTextileTechnique*. El valor de esta propiedad debe de poder visualizarse en el mapa espacio temporal, además de poder filtrar datos por este valor, y visualizar los objetos relacionados con el mismo valor de esta propiedad. Las propiedades y relaciones de esta instancia se muestran en la Tabla 37.

Tabla 37 Propiedades y relaciones de la propiedad visual *visualTextileTechnique*

| Tipo      | Nombre                   | Valor                          |
|-----------|--------------------------|--------------------------------|
| Propiedad | is_relatable             | True                           |
| Propiedad | Is_filtered              | True                           |
| Propiedad | Has_name                 | "technique"                    |
| Propiedad | Uses_query_variable_name | "techniqueLabel"               |
| Relación  | Represented_by_color     | greenColor (Color)             |
| Relación  | Value_per_query          | queryForTextileMatyTec (Query) |
| Relación  | Is_domain_property       | visualTextile (VisualConcept)  |



### Categoría del objeto.

El tipo, o categoría del objeto, se resuelve en el proyecto SILKNOW mediante la relación *P41\_classified\_of*, que relaciona la instancia del concepto *E22\_Man-made\_Object* con instancias de *Object\_type\_assignment*, que hace referencia un lenguaje controlado de términos con los diferentes tipos de textiles de SILKNOW.

Esta propiedad requerirá la instanciación de *Visual\_Property*, que se denominará *visualTextileCategory*. En este caso, al ser una propiedad accesible mediante una relación directa del concepto del dominio de datos, no es necesario usar una consulta para acceder a su valor. Para ello se dará valor a la propiedad *property\_URI*. El valor de esta propiedad visual del objeto no será observable, ni dará opción a filtrar, ni relacionar. Eso es así porque el usuario podrá ver el valor directamente sobre el objeto, ya que el objeto tendrá una imagen u otra, en función del valor de esta propiedad. En la Tabla 38 se muestran las propiedades y relaciones de la instancia de *visualTextileCategory* (*Propiedad Visual*).

Tabla 38 Propiedades y relaciones de la instancia de propiedad visual *visualTextileCategory*

| Tipo      | Nombre             | Valor   |
|-----------|--------------------|---|
| Propiedad | is_relatable       | False   |
| Propiedad | Is_filtered        | False   |
| Propiedad | Has_name           | "category"  |
| Propiedad | Property_URI       | <a href="http://erlangen-crm.org/current/P41_classified">http://erlangen-crm.org/current/P41_classified</a> |
| Relación  | Is_domain_property | visualTextile (VisualConcept)   |

### Posición del objeto.

La posición del objeto es una propiedad especial, ya que se enlaza con el concepto visual a través de la propiedad *position\_domain\_property*.

Por lo demás, sigue una definición similar a la del resto de propiedades.

En el modelo de SILKNOW, siguiendo la filosofía del modelo CIDOC\_CRM, al igual que sucedía con el material y la técnica, la ubicación de producción de un objeto se hace mediante la relación con *E12\_Production*, y dentro de esta entidad, se relaciona, a través de la propiedad *P8 took place on or within* con una instancia de la entidad *Feature*<sup>1</sup> del grafo de conocimiento de Geonames.org. De esta entidad se referenciarán en STEVO las siguientes propiedades:

- *geonames:name* → Nombre principal del lugar, con el prefijo del idioma.
- *geo:lat* → latitud del lugar.
- *geo:long* → longitud del lugar.

Esto implica, la instanciación de tres propiedades visuales en STEVO:

- *visualTextilePositionName*, con el nombre/nombres de la ubicación del lugar de producción.
- *visualTextilePosX*, con la longitud del objeto.
- *visualTextilePosY*, con la latitud del objeto.

<sup>1</sup> URI del concepto Feature del grafo de conocimiento de Geonames.org.  
<<http://www.geonames.org/ontology#Feature>>

La propiedad *visualTextilePositionName*, debido a la incerteza de la producción de los objetos, debe de poder visualizarse, además de filtrar y visualizar los objetos relacionados con el mismo valor en dicha propiedad. En la Tabla 39 se enumeran las propiedades y relaciones de esta propiedad visual.

Las propiedades *visualTextilePosX* y *visualTextilePosY*, se usarán únicamente para la representación del objeto, no se podrá visualizar su contenido, ni filtrar, ni relacionar.

Las propiedades y relaciones de *visualTextilePosX* se detallan en la Tabla 40, las de *visualTextilePosY* son análogas, pero cambiando X por Y en el nombre.

Tabla 39 Relaciones y propiedades de la instancia de propiedad visual *visualTextileProductionName*

| Tipo      | Nombre                   | Valor                             |
|-----------|--------------------------|-----------------------------------|
| Propiedad | is_relatable             | True                              |
| Propiedad | Is_filtered              | True                              |
| Propiedad | Has_name                 | "production"                      |
| Propiedad | Uses_query_variable_name | "production"                      |
| Relación  | Represented_by_color     | greenColor (Color)                |
| Relación  | Value_per_query          | queryForTextileProduction (Query) |
| Relación  | Is_domain_property       | visualTextile (VisualConcept)     |

Tabla 40 Relaciones y propiedades de la instancia de propiedad visual *visualTextilePosX*

| Tipo      | Nombre                        | Valor                             |
|-----------|-------------------------------|-----------------------------------|
| Propiedad | is_relatable                  | false                             |
| Propiedad | Is_filtered                   | false                             |
| Propiedad | Has_name                      | "longitud"                        |
| Propiedad | Uses_query_variable_name      | "longitud"                        |
| Relación  | Value_per_query               | queryForTextileProduction (Query) |
| Relación  | Is_Position_x_domain_property | visualTextile (VisualConcept)     |

También hay que instanciar el concepto de *Query*, para poder consultar los datos de la producción del objeto. La instancia se denomina *queryForTextileProduction* y en la se describen las relaciones y propiedades de la misma. En la Tabla 41 se enumeran las relaciones y propiedades de esta instancia.

#### Tiempo de producción del objeto.

Para tratar el tiempo en SILKNOW, se usa una instancia del concepto E12\_Production, como para representar la técnica y el material. Desde esa instancia, con la propiedad P4-has\_time-span, se accede a una, o varias instancias del concepto E52\_Time-Span del modelo CIDOC-CRM. Esta instancia tiene las propiedades hasBeginning y hasEnd, que permiten obtener los valores de inicio y fin del intervalo.

En SILKNOW, aunque se usan etiquetas alternativas, la etiqueta preferente aplica una unidad temporal anual para expresar el intervalo.

Tabla 41 Relaciones y propiedades de la instancia de Query, *queryForTextileProduction*

| Propiedad | Valor |
|-----------|-------|
|-----------|-------|

|                              |  |
|------------------------------|--|
| Has_variable<br>for_instance | this   |
| Has_expression               | <pre> SELECT distinct ?latitud ?longitud WHERE {   ?s   rdf:type   &lt;http://erlangen-crm.org/current/E12_Production&gt; .   ?s   &lt;http://erlangen-crm.org/current/P108_has_produced&gt;   this .   ?s   &lt;http://erlangen-crm.org/current/P8_took_place_on_or_within&gt;   ?p .   ?p   rdf:type   &lt;http://www.geonames.org/ontology#Feature&gt; .   ?p   &lt;http://www.w3.org/2003/01/geo/wgs84_pos#lat&gt;   ?latitud .   ?p   &lt;http://www.w3.org/2003/01/geo/wgs84_pos#long&gt;   ?longitud   ?p   &lt;http://www.geonames.org/ontology#name&gt;   ?production. } </pre> |

#### Tiempo de producción del objeto.

Para tratar el tiempo en SILKNOW, se usa una instancia del concepto E12\_Production, como para representar la técnica y el material. Desde esa instancia, con la propiedad P4-has\_time-span, se accede a una, o varias instancias del concepto E52\_Time-Span del modelo CIDOC-CRM. Esta instancia tiene las propiedades hasBeginning y hasEnd, que permiten obtener los valores de inicio y fin del intervalo.

En SILKNOW, aunque se usan etiquetas alternativas, la etiqueta preferente aplica una unidad temporal anual para expresar el intervalo.

En la ontología STEVO para representar estas propiedades hay que instanciar el concepto *Visual\_Property* con la instancia *visualTextileTimeStart* y *visualTextileTimeEnd*. Los valores de estas propiedades deben de poder visualizarse en el mapa espacio temporal, sobre todo debido a la incerteza, que es muy alta en esta propiedad. En SILKNOW se determinó que no era muy eficiente la visualización de objetos relacionados, también por la incerteza de los datos, y por tratarse de intervalos. Era poco probable en el dominio de datos del proyecto hallar objetos relacionados que tuvieran el mismo valor de intervalo temporal. Podría tener sentido si la relación fuera efectiva en el momento en que hubiera un porcentaje de solapamiento en el intervalo temporal, pero es un dato que habría que investigar. Las propiedades y relaciones de la instancia *visualTextileTimeStart* se enumeran en la Tabla 42, los de *visualTextileTimeEnd* son totalmente análogos.



Tabla 42 Relaciones y propiedades de datos de la instancia *visualTextileTimeStart*

| Tipo      | Nombre                        | Valor                         |
|-----------|-------------------------------|-------------------------------|
| Propiedad | is_relatable                  | false                         |
| Propiedad | Is_filtered                   | true                          |
| Propiedad | Has_name                      | "from"                        |
| Propiedad | Uses_query_variable_name      | "start"                       |
| Relación  | Value_per_query               | queryForTextileTime (Query)   |
| Relación  | Is_time_start_domain_property | visualTextile (VisualConcept) |

En la Tabla 43 se enumeran las relaciones y valor de las propiedades de la instancia del concepto *Query*, *queryForTextileTime*, usada para obtener los valores de las instancias de propiedades visuales *visualTextileTimeStart* y *visualTextileTimeEnd*, que obtiene los valores de tiempo de producción asociados a un objeto.

Tabla 43 Relaciones y propiedades de datos de la instancia *queryForTextileTime*

| Propiedad                 | Valor  |
|---------------------------|--|
| Has_variable_for_instance | this   |
| Has_expression            | <pre> SELECT distinct ?this, ?start, ?end WHERE {   ?s   rdf:type   &lt;http://erlangen-crm.org/current/E12_Production&gt; .   ?s   &lt;http://erlangen-crm.org/current/P108_has_produced&gt;   ?this .   ?s   &lt;http://erlangen-crm.org/current/P4_has_time-span&gt;   ?p .   ?p   &lt;http://www.w3.org/2006/time#hasBeginning&gt;   ?st .   ?p   &lt;http://www.w3.org/2006/time#hasEnd&gt;   ?ed .   ?st   &lt;http://www.w3.org/2006/time#inXSDDate&gt;   ?start .   ?ed   &lt;http://www.w3.org/2006/time#inXSDDate&gt;   ?end .   FILTER (?uriInstance in (this.value)) }</pre> |

#### Definición de la representación gráfica de los conceptos.

En el mapa espacio temporal de SILKNOW únicamente se representan los objetos y sus agrupaciones.

Las agrupaciones se representan con un círculo, por lo que es muy sencilla su representación, ya que sólo necesitan la definición de un Marcador cuya representación gráfica sea un círculo.

En este caso habrá que crear una instancia de Marcador, que se denominará *silknowClusterMarker*, asociada al comportamiento de agrupación, que será un marcador del tipo *Circle\_Marker*, representado por una figura geométrica circular.

Sin embargo, el concepto representado por la instancia *visualTextile*, es similar, pero tiene una complejidad asociada. El marcador del objeto que representa el concepto, en función del tipo de categoría del objeto tiene asociado una textura que lo define. Para ello, se crea una instancia del concepto Marcador, que tiene asociados tantos modificadores como diferentes categorías tenga el objeto. Además, al marcador se le asociará la instancia de representación gráfica primitiva (Primary Graphic Representation, de ontología VISO) *circleTextile*, para definir diferentes propiedades de la figura. En la Tabla 44 se enumeran las propiedades y relaciones de dicha instancia.

Tabla 44 Relaciones y propiedades de la Instancia de *circleTextile*

| Tipo      | Nombre                     | Valor   |
|-----------|----------------------------|---|
| Propiedad | has_image                  | defaultTextileTexture (instancia de Picture con valor de textura con defecto en la propiedad has_URL) |
| Propiedad | Has_graphic_representation | circle (Shapes, ontología VISO)   |
| Propiedad | color_rgb                  | LightBlueColor ( Color, ontología VISO)   |

La instancia de Marcador, se denomina *silknowTextileMarker*. En la Tabla 45 se enumeran las propiedades y relaciones de esta instancia del concepto Marcador. Este marcador tiene tantas relaciones con instancias de modificador, como tipos de tejidos diferentes se van a representar. En la tabla se representan los modificadores tejidos de tipo Ropa, Religioso, Desconocido, Hogar, y Complementos.

Tabla 45 Relaciones y propiedades de la instancia de Marcador *silknowTextileMarker*

| Tipo     | Nombre                     | Valor  |
|----------|----------------------------|--|
| Relación | has_graphic_representation | circleTextile (Marcador)                               |
| Relación | Usa_modificador            | correspondingTextileTypeClothesPicture (Modificador)   |
| Relación | Usa_modificador            | correspondingTextileTypeReligiousPicture (Modificador) |
| Relación | Usa_modificador            | correspondingTextileTypeUnkownPicture (Modificador)    |
| Relación | Usa_modificador            | correspondingTextileTypeHomePicture (Modificador)      |
| Relación | Usa_modificador            | correspondingTextileTypeCompPicture (Modificador)      |

A modo de ejemplo, se detallan las propiedades y relaciones de la instancia del modificador usado para representar los objetos de tipo Ropa en la Tabla 46 .

Tabla 46 Relaciones y propiedades de la instancia correspondingTextileTypeClothes del concepto Modificador.

| Tipo      | Nombre                       | Valor  |
|-----------|------------------------------|--|
| Relación  | Modifica_sobre_valor         | visualTextileType  |
| Propiedad | Tiene_representación_gráfica | PictureTextileTypeClothes (instancia de Picture con valor de textura del tipo Ropa la propiedad has_URL) |
| Propiedad | Valor_de_propiedad           | http://data.silknow.org/category/40  |

En la Figura 79 se muestra la representación gráfica de varios objetos del mapa espacio temporal de SILKNOW. Como se puede apreciar en la imagen, cada objeto tiene un tipo de textura que lo identifica en función del tipo al que pertenece el objeto.

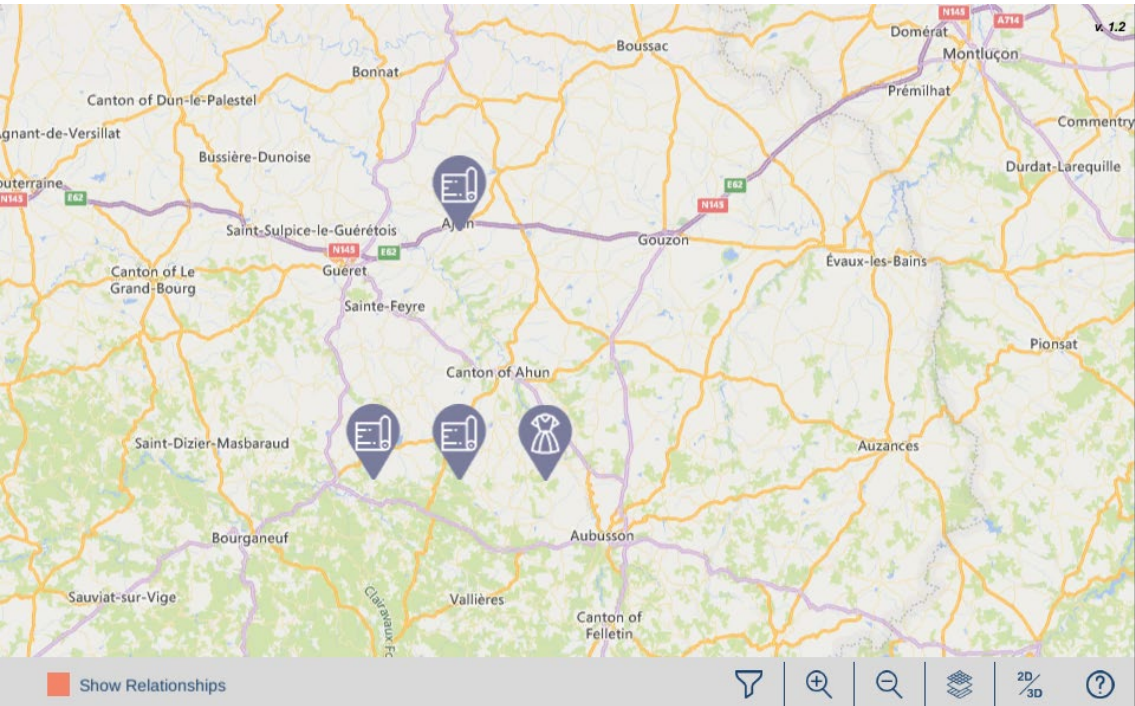


Figura 79 Representación gráfica de los conceptos representados en el mapa espacio temporal. Fuente: ADASILK (SILKNOW Consortium 2022)

Definición de los comportamientos.

Dada la complejidad de los requerimientos del mapa espacio temporal del proyecto SILKNOW será necesario instanciar varios conceptos de comportamiento STEVO para que se pueda representar la interacción de forma adecuada.

A continuación, se detallan las instancias realizadas en STEVO, y sus relaciones, para representar los comportamientos de agrupación, de control de tiempo, y de visualización de relaciones entre objetos.

Agrupaciones.

Dado que los datos se agruparán en grupos, y no en mapas de calor se hará uso del concepto *Clusters\_Group\_Behaviour*, que extiende a *Group\_Behaviour*, creando la instancia *silknowGroupBehaviour*. Esta instancia usará la instancia de marcador *silknowClusterMarker* comentada previamente.

En la Tabla 47 se exponen las relaciones y propiedades de la instancia *silknowGroupBehaviour*.

Tabla 47 Propiedades y relaciones de la instancia *silknowGroupBehaviour* (*Clusters\_Group\_Behaviour*) para representar cómo visualizar los agrupamientos.

| Tipo     | Nombre                        | Valor                                  |
|----------|-------------------------------|--|
| Relación | Marcador_representante        | <i>silknowClusterMarker</i> (Marcador) |
| Relación | Es_comportamiento_de_grupo_de | <i>sceneSilknow</i> (Escena)           |

En la Figura 80 se muestran varias capturas de pantalla del mapa espacio temporal de SILKNOW. Las capturas muestran el resultado de un proceso de navegación en el que el usuario se acerca cada vez más a la ciudad de París. Inicialmente se ven agrupamientos en toda Europa, pero conforme se acerca, además de ver el mapa con mayor información, se muestran diferentes agrupamientos, resultado de los agrupamientos de nivel superior.

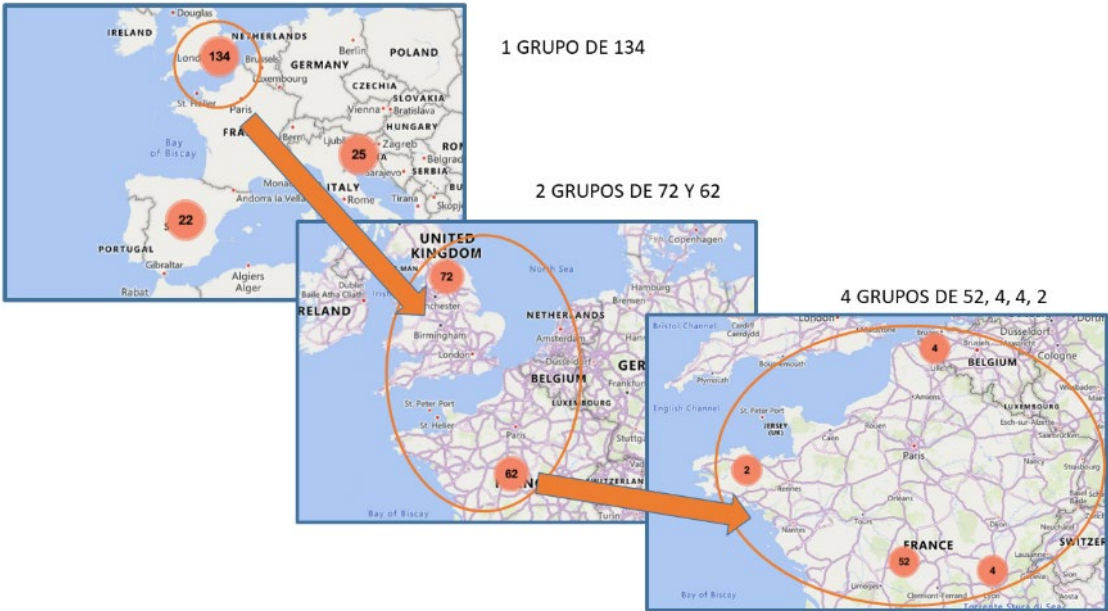


Figura 80 Zoom con agrupamientos sobre mapa espacio temporal de ADASILK. Composición con capturas de pantallas de ADASILK.Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

Control de tiempo.

El control de tiempo en el mapa espacio temporal de SILKNOW se tiene que poder hacer de dos formas. Primero con un control de línea de tiempo (TimeLine), de forma que tomando como unidad de tiempo el siglo, se pueda ver cuáles de los objetos se han producido en siglo concreto, y de forma sencilla, ver los que se han producido en el siguiente, o en el anterior. Usando el control con cierta rapidez se puede observar un efecto película, ya que en cada interacción se verían los datos de cada siglo de forma correlativa.

Por otra parte, también se desea poder observar los datos de cada siglo, o un intervalo de siglos, de forma simultánea a los de otro siglo, o intervalo. Con esta función se puede observar de forma simultánea los datos de cada intervalo de tiempo.

Para definir el control de tiempo será necesario instanciar dos conceptos de STEVO, el primero será *Control\_Individual*, para representar el TimeLine, y el segundo *Control\_Simultáneo*.

La instancia de *Control\_Individual* será *silknowTimeLineControl*, cuyas relaciones y propiedades se exponen en la Tabla 48.

En la Figura 81 se muestra una captura de pantalla del mapa espacio temporal de SILKNOW con el detalle de la interfaz de usuario para activar los diferentes controles de tiempo, que en este caso son dos: Individual (Timeline) y Simultáneo.

Tabla 48 Propiedades y relaciones de la instancia *silknowTimeLineControl* (*Control\_Individual*)

| Tipo     | Nombre                         | Valor   |
|----------|--------------------------------|---|
| Relación | Frecuencia_control_tiempo      | unityCentury (Unidad Temporal)<br>Instancia predefinida en STEVO. |
| Relación | Es_comportamiento_de_tiempo_de | sceneSilknow (Escena)   |

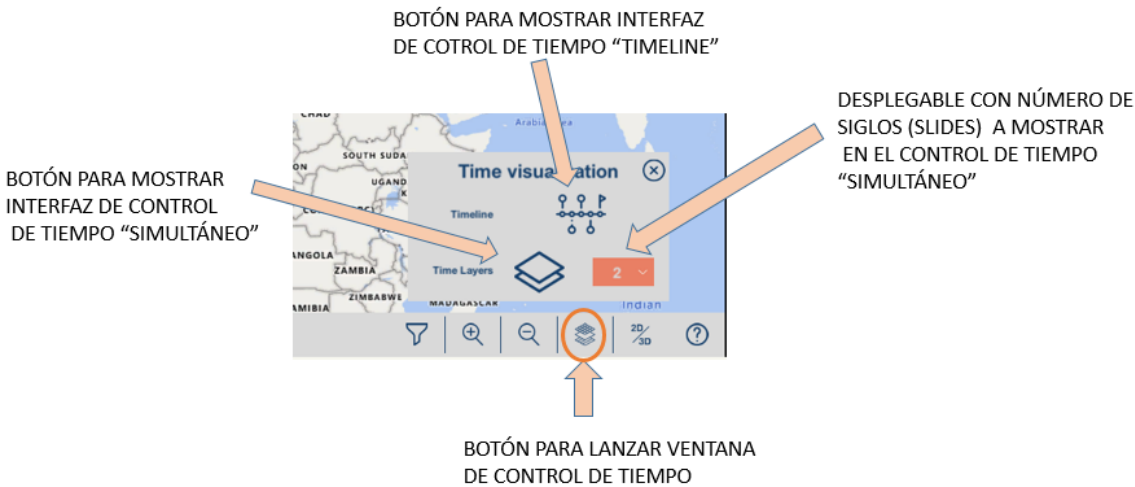


Figura 81 Captura de ADASILK mostrando la interfaz para seleccionar los diferentes modos de control de tiempo y sus parámetros. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022).

En la Figura 82 se muestran diferentes capturas de pantalla del mapa espacio temporal de SILKNOW con la interfaz de control de tiempo Individual, o Timeline. Esta interfaz consiste en un control de tipo slider, que permite activar uno u otro siglo de forma sencilla. De esta forma



se ven de izquierda a derecha los datos que han sido producidos en los siglos 15, 16 y 17 respectivamente.

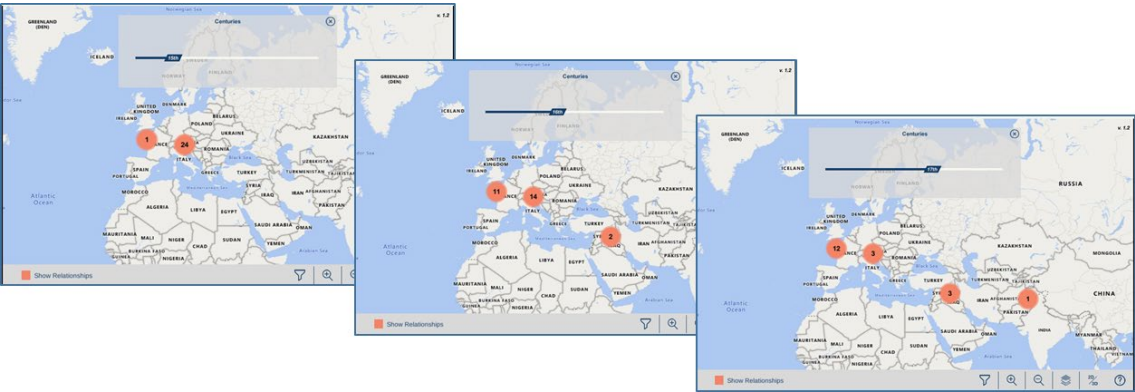


Figura 82 Diferentes capturas de pantallas con el control de tiempo "TimeLine" activo en la parte superior sobre un mismo conjunto de datos. Cada captura corresponde a un siglo diferente. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

La instancia de *Control\_Simultáneo* será *silknowSimultaneousTimeControl* y sus relaciones y propiedades se enumeran en la Tabla 49.

Tabla 49 Relaciones y Propiedades de la instancia *silknowSimultaneousTimeControl* (*Control\_Simultaneo*)

| Tipo      | Nombre                         | Valor   |
|-----------|--------------------------------|---|
| Relación  | Frecuencia_control_tiempo      | unityCentury (Unidad Temporal)<br>Instancia predefinida en STEVO. |
| Relación  | Es_comportamiento_de_tiempo_de | sceneSilknow (Escena)   |
| Propiedad | Usa_pasos                      | True  |
| Propiedad | Número_máximo_de_pasos         | 4   |
| Propiedad | Visualiza_zona_espacial        | True  |

En la Figura 83 se muestran dos capturas con el modo de control de tiempo simultáneo del mapa espacio temporal de SILKNOW. En las dos capturas se visualiza el mismo conjunto de datos, mostrando para la misma zona espacial tres capas de datos. En la figura superior la capa activa es la inferior que comprende el intervalo del siglo 17 al 18, aunque también se observan los datos del intervalo definido por los siglos 16 a 17. En la figura inferior, tras activar la capa activa, la comprendida entre los siglos 16 y 17, las capas de desplazan hacia abajo, mostrando perfectamente la distribución de datos entre los siglos 16 y 17, y dejando ver los datos entre el 15 y 16, así como mostrando una gran parte de la capa definida por los siglos 16 y 17.

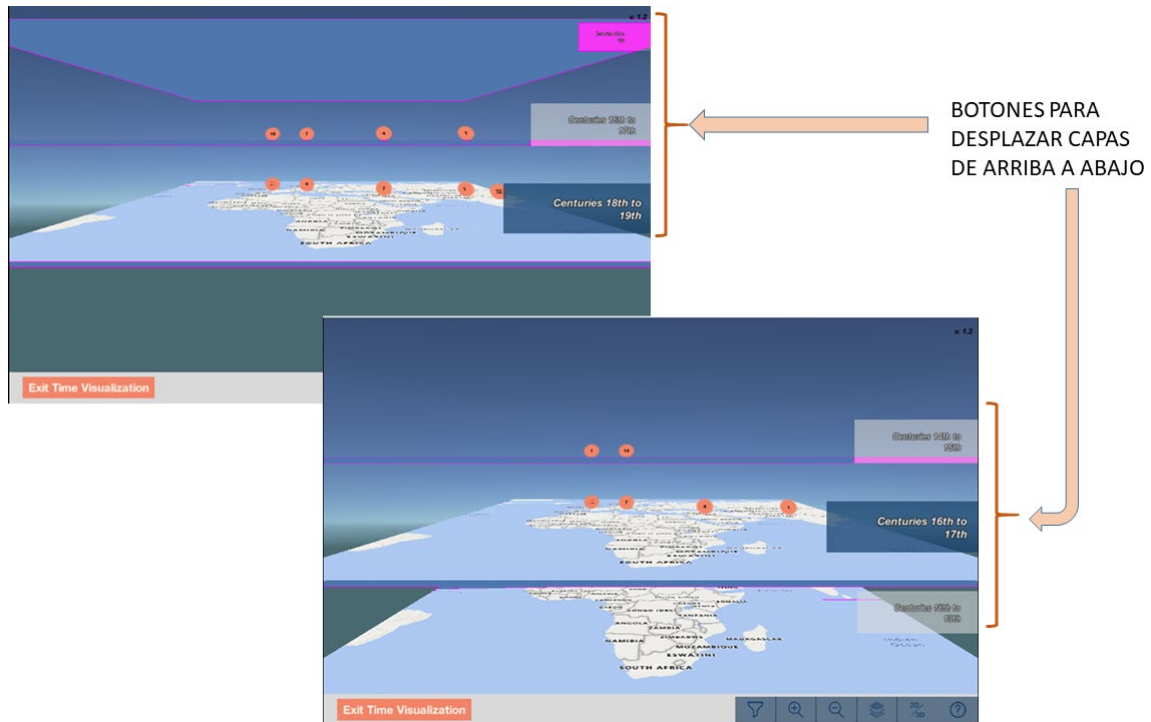


Figura 83 Capturas de pantalla de visualización de tiempo simultáneo en mapa SILKNOW. Cada capa representa un intervalo de tiempo (en siglos). Los botones de la derecha permiten mover las capas de arriba a abajo para mejor visualización. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

#### Relaciones.

En el mapa espacio temporal de SILKNOW, representado a través de la ontología STEVO, se visualizan dos tipos de relaciones entre los objetos, usando las propiedades que sean relacionables (propiedad “relacionable” del concepto Propiedad\_Visual).

El primer tipo dibuja líneas entre los objetos que tengan el mismo valor de las propiedades marcadas para ver sus relaciones. El segundo tipo representa el control de anillo, que permite ver el porcentaje de objetos que tienen el mismo valor para cada una de sus propiedades relacionables.

Para el primer tipo de relaciones, habrá que instanciar el concepto *Relación\_Enlaces* mediante la instancia *silknowLinkedRelationBehaviour*. En la Tabla 50 se definen las propiedades y relaciones de esta instancia.

Tabla 50 Relaciones y propiedades de la instancia *silknowLinkedRelationBehaviour* (Relacion\_Enlaces)

| Tipo      | Nombre                        | Valor                 |
|-----------|-------------------------------|-----------------------|
| Relación  | Es_comportamiento_de_Relación | sceneSilknow (Escena) |
| Propiedad | Aplica_Técnica_de_líneas      | True                  |
| Propiedad | Número_máximo_de_enlaces      | 2000                  |
| Propiedad | Aplica_Técnica_de_Color       | False                 |

En la Figura 84 se muestran dos capturas de pantalla del mapa espacio temporal de silknow, tras activar la visualización de objetos relacionados a través de la propiedad material. En este caso, todos los objetos que se están visualizando tienen el mismo material que el objeto seleccionado, por lo que está relacionado con todos. En la parte superior, el nivel de zoom es alto, y se ve la relación con los objetos próximos. En la parte inferior, se aleja más la vista del usuario, teniendo

un nivel de zoom más bajo, y se observan las relaciones, pero a través de los agrupamientos, ya que el objeto activado y otros no se visualizan.

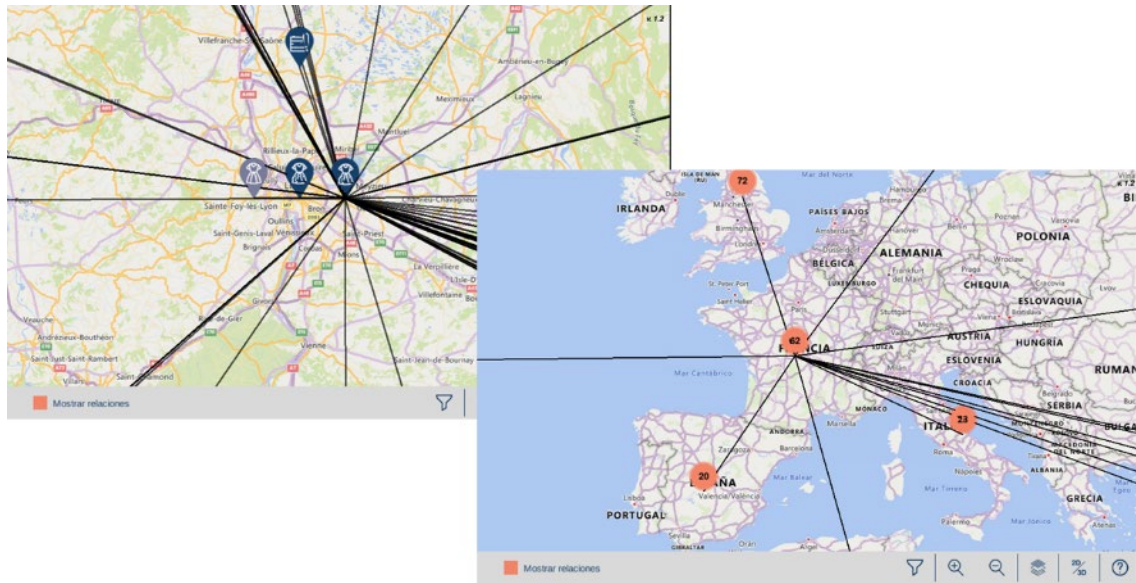


Figura 84 Figura compuesta por dos capturas de pantalla en las que se muestran las relaciones de un objeto con el resto. La imagen superior es a nivel de zoom alto, y la inferior es a un nivel de zoom más bajo, las relaciones se muestran entre las agrupaciones. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

Para representar el segundo tipo de relaciones con anillos, o corona circular, que permita saber el porcentaje de objetos que tienen el mismo valor en las propiedades de todos los objetos, para cada objeto que se visualice, se requiere instanciar el concepto *Relacion\_Anillo*. Se crea la instancia *silknowMarkerRelationBehaviour* y se relaciona con la instancia *sceneSilknow*, de Escena, mediante la relación “*Tiene\_Comportamiento\_de\_Relación*” y su inversa “*Es\_Comportamiento\_de\_Relación*”.

En la Figura 85 se muestra una captura de pantalla del mapa espacio temporal de SILKNOW, tras activar la visualización de relaciones en anillo de los objetos que se están visualizando.

La zona espacial es la misma que en la representada en la Figura 84, y los objetos son los mismos. Se ve que efectivamente el objeto está relacionado con todos los objetos (100%) a través de la propiedad “material”, como se veía en la otra figura con las líneas. Este porcentaje se ve visualmente, que también sucede con el objeto de al lado, pero en los otros dos, aunque es alto, ya no es del 100%.

#### Incerteza

Al tener datos de patrimonio cultural en el proyecto SILKNOW, no es nada extraño que también tengan incerteza. En este caso, los datos presentan incerteza principalmente en espacio y tiempo. Al tratarse de tejidos, en ocasiones de casi un milenio de antigüedad, y al carecer en numerosas ocasiones de documentos de compra y venta, así como firmas o etiquetas de autoridad, es complejo saber quién, cuándo y dónde se produjeron. Estos datos tienen que obtenerse a partir de los materiales, técnica, los motivos del tejido y los diferentes lugares de procedencia, lo que conlleva una incerteza.



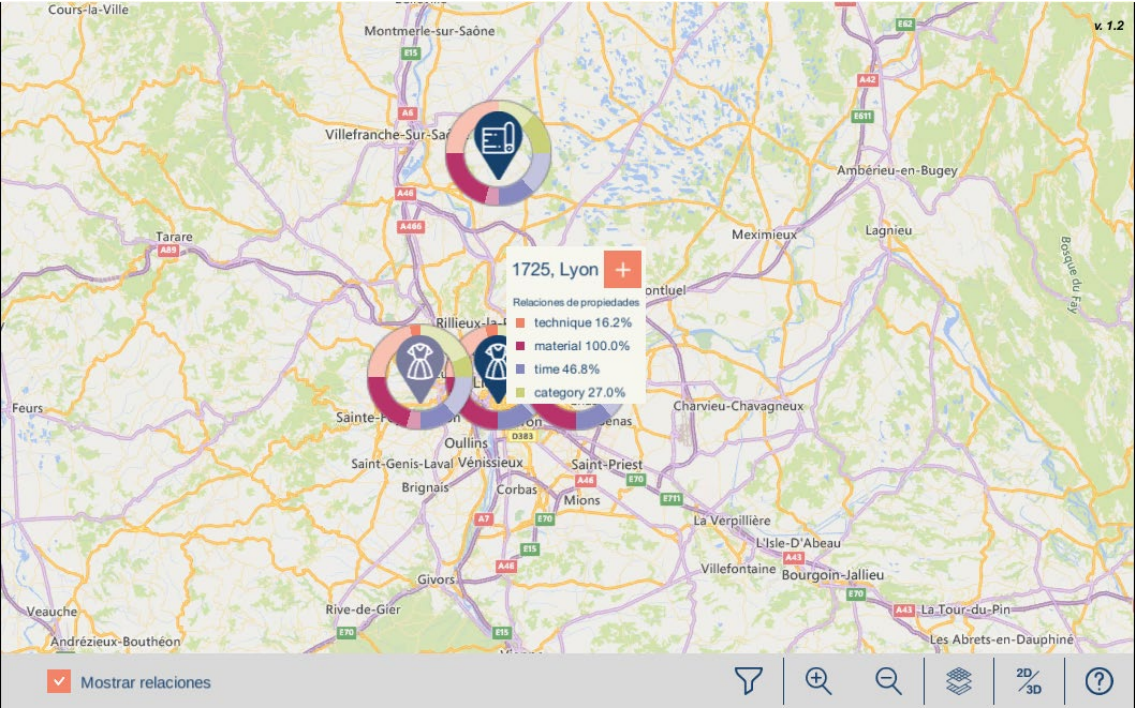


Figura 85 Captura de pantalla del mapa espacio temporal de SILKNOW. En esta imagen se aprecia el anillo de con las relaciones totales de los objetos. Se aprecia la leyenda de propiedades que muestra que el 100% tienen el mismo material. Fuente: Adaptada de ADASILK (SILKNOW Consortium 2022)

Tal y como se ha discutido en la definición del modelo de esta investigación (ver capítulo 6), hay varias formas de gestionar este problema. En el proyecto SILKNOW se decidió hacerlo mediante replicación espacial y decoloración de marcador. Es decir, si un objeto puede proceder de dos, o más lugares diferentes, su ubicación espacial en la representación del dato será una para cada posible lugar. Esto puede generar confusión, ya que un resultado puede generar un número X de datos que cumplen una consulta, pero en la representación pueden existir X + N datos, siendo N un incremento generado por la duplicación de incerteza. Esta situación puede suceder fácilmente en conjuntos relativamente pequeños de datos y es confusa porque con las agrupaciones se representa el número de datos. En una visualización con un nivel bajo de zoom, con pocas agrupaciones, es sencillo de detectar, mediante la suma de las agrupaciones, que el número de datos es mayor que el esperado y esto puede confundir al usuario.

Para representar la incerteza del mapa espacio temporal de SILKNOW, se instancia el concepto de STEVO *Tratamiento\_Incerteza\_Con\_Replicacion*, mediante la instancia *silknowUncertaintyBehaviour*. En la Tabla 51 se detallan las propiedades y relaciones de esta instancia.

Tabla 51 Relaciones y propiedades de la instancia *silknowUncertaintyBehaviour* (*Tratamiento\_Incerteza\_Con\_Replicacion*)

| Tipo      | Nombre                      | Valor                 |
|-----------|-----------------------------|-----------------------|
| Relación  | Es_tratamiento_de_incerteza | sceneSilknow (Escena) |
| Propiedad | Aplica_Decoloracion         | True                  |
| Propiedad | Aplica_Desplazamiento       | False                 |

La aplicación de esta técnica en el mapa espacio temporal de SILKNOW puede apreciarse en la Figura 85 y en la Figura 84, donde se aprecia claramente un dato que tiene un color más claro

que el resto, aunque sea del mismo tipo, este dato presenta incerteza. Si se quiere conocer el tipo de incerteza, que puede ser por el espacio, tiempo, o ambos datos, hay que visualizar la información del objeto, como se aprecia en la Figura 77, para ver en la interfaz las propiedades “Lugar” y “Tiempo”, que en el caso de que tengan varios datos, sería la causa de la incerteza.

En la Figura 76 se muestra que hay objetos, donde el icono tiene un color claro, esto es debido a que estos presentan incerteza de algún tipo.

#### Unidades de tiempo, espacio y sistemas de referencia.

Las unidades de tiempo, espacio, así como los espacios de referencia usados en el mapa espacio temporal de SILKNOW, se pueden representar con instancias predefinidas en la ontología STEVO.

Para definir el entorno temporal se ha creado una instancia denominada *silknowTime*, del concepto *Entorno\_Temporal*, indicando que el intervalo de tiempo de trabajo de SILKNOW (desde el año 0 de nuestra era hasta el siglo XX) y la unidad de tiempo, que es el siglo. En la Tabla 52 se enumeran las propiedades y relaciones de esta instancia.

Tabla 52 Relaciones y propiedades de la instancia *silknowTime* (*Entorno\_Temporal*)

| Tipo     | Nombre             | Valor                         |
|----------|--------------------|-------------------------------|
| Relación | Unidad_Temporal    | unitCentury (Unidad_Temporal) |
| Relación | Comienzo_de_Tiempo | startSilknowTime (Instant)    |
| Relación | Fin_de_Tiempo      | endSilknowTime (Instant)      |

Para referenciar el entorno espacial del mapa, se usa la instancia predefinida en STEVO del concepto *Entorno\_Espacial*: “Unity3D”

### 9.3. APLICACIÓN DE LOS RESULTADOS EN EL PROYECTO ARXIU VALENCIÀ DE DISSENY

#### 9.3.1. Proyecto Arxiu Valencià del Disseny

L’Arxiu Valencià del Disseny (U. València 2022), se erige como un centro de custodia, conservación, catalogación y digitalización de la documentación generada por los diseñadores, los estudios de diseño, y los departamentos de diseño de las principales empresas valencianas, para que puedan ser consultados, estudiados y puestos a disposición de los investigadores sobre el diseño y sobre la cultura moderna, analizando, a partir de documentación original, los procesos de creación, producción, difusión, venta y uso de los objetos. Todo ello con la finalidad, no solo de custodia y estudio, sino también con el interés de poner al alcance del tejido productivo e industrial unos fondos digitalizados y relacionados a través de herramientas analíticas y de mapas conceptuales y de producto capaces de fomentar la innovación y la creatividad del sector industrial.

El archivo tiene como una de sus funciones primordiales, facilitar el acceso a sus fondos a través de la red difundiendo on-line la documentación, así como el desarrollo de tecnologías 3D y una

web con un buscador con contenido semántico aplicadas a la visualización de los objetos de diseño y su proceso de creación a través de la plataforma web de la Universitat de València.

L'arxiu Valencià del Disseny, con una metodología basada en la descripción de sus fondos siguiendo estándares internacionales y ante la gran variedad, transversalidad de la documentación y la multitud de soportes, pretende constituirse como un laboratorio de innovación y creatividad.

Con este proyecto se definen y se crean las bases para llevar a cabo una difusión innovadora del Arxiu a través de la configuración de una serie de herramientas que, en base a la catalogación y digitalización de sus fondos, creando de manera paralela un Archivo de la Memoria del Diseño (Investigación etnológica) y desarrollando una Web con un buscador semántico y capaz de enlazar semánticamente los archivos digitalizados. De esta forma los usuarios pueden acceder más intuitivamente tanto a los objetos de diseño, a la documentación de empresa, catálogos, etc., como a las relaciones significativas entre ellos, configurándose como una fuente sustancial para la promoción y el desarrollo de la cultura de la innovación.

Una de las herramientas software diseñadas e implementadas en el proyecto es un mapa de producto. Esta herramienta permite asignar variables a las diferentes instancias del Arxiu y representar mediante gráficos 2D y 3D dichas instancias. El sistema debe de permitir realizar búsquedas sobre los datos del dominio y así seleccionar una serie de datos que cumplan una serie de restricciones.

El proyecto actualmente está en desarrollo, por lo que no se puede publicar, ni la ontología del mismo, ni partes como la ontología STEVO que se va a usar para desarrollar el mapa de producto.

Pero se trata de un proyecto cuyo grafo de conocimiento, al igual que SILKNOW, está basado en CIDOC-CRM, y gestionará decenas de miles de datos.

El mapa de producto, es una herramienta del ámbito de diseño que consiste en usar los valores de ciertas variables de un producto de diseño que pertenece al patrimonio cultural, para representar estos datos como puntos espaciales en un espacio cartesiano de dos, o tres dimensiones, en función del número de variables.

#### *9.3.2. Instanciación en STEVO*

Al estar este proyecto en desarrollo, no se pueden publicar enlaces, ni detalles técnicos, pero esta visualización es mucho más sencilla que la realizada en el proyecto SILKNOW. En este caso sólo hay que enlazar las propiedades espaciales que definen un objeto del dominio, con las coordenadas de visualización. Como existen predefinida en STEVO instancias del concepto Entorno\_Espacial y no hay ningún control de tiempo, incerteza, ni relacional, la instanciación de STEVO es sencilla. Únicamente hay que instanciar la escena, un concepto visual, las propiedades espaciales, y otras que se usarán únicamente para su visualización.

En la Figura 86 se expone una captura de pantalla con el resultado de una consulta sobre el buscador del Arxiu.

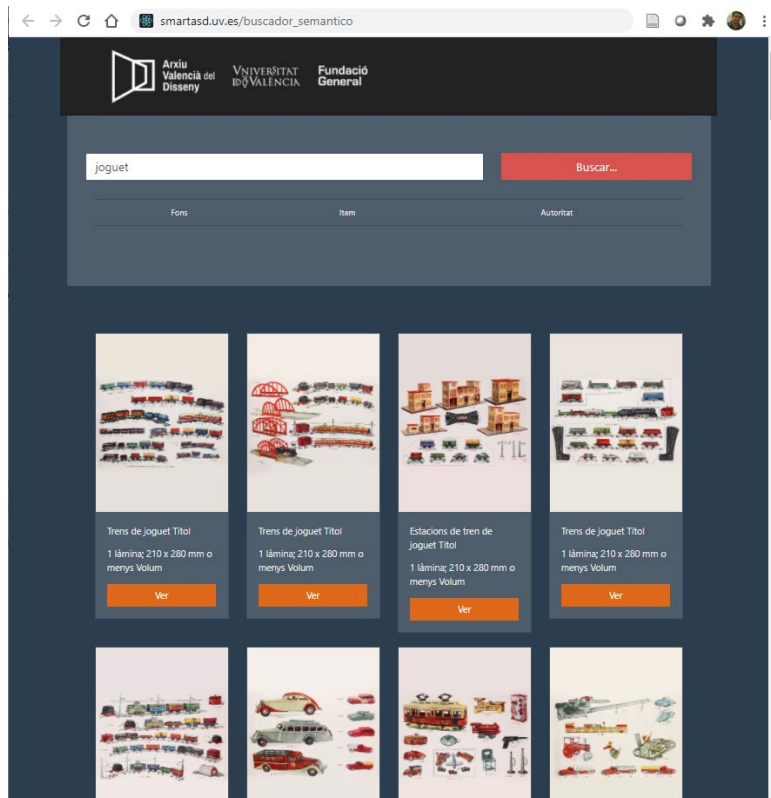


Figura 86 Captura de pantalla del portal web prototipo de la web d' l'arxiu. Fuente: Web de AVD (U. València 2022)

Sobre el resultado de una búsqueda se visualiza, tras seleccionar las propiedades que van a servir de coordenadas espaciales y se visualiza la posición de estos datos en un eje cartesiano, que permite la navegación interactiva por el espacio del mismo. Esta visualización ha sido realizada basándose mediante el marco de trabajo definido en esta investigación, con la ontología STEVO.

En la Figura 87 se muestra una captura de pantalla con el mapa de producto del portal de l'Arxiu en el que se usa un espacio tridimensional para representar la información.



Figura 87 Herramienta de visualización del mapa de producto del portal de l'Arxiu. La visualización ha sido representada con la ontología STEVO. Fuente: Web de AVD (U. València 2022)

#### 9.4. CONCLUSIONES

En este capítulo se han detallado con profundidad los resultados de incluir el marco de trabajo definido en esta investigación con el objeto de visualizar un mapa espacial interactivo. Los datos, tal y como es el objetivo de esta investigación, pertenecen al dominio del patrimonio cultural y están soportados en un grafo de conocimiento.

Se ha observado que es viable incluir el componente en un portal web, no sólo físicamente, sino que también es capaz de interactuar con el portal, recogiendo como datos el resultado de una búsqueda realizada sobre el portal, y además es capaz de comunicarse de forma síncrona para obtener información de un dato del mapa.

La herramienta implementada se ha basado en el contenido, e información, de la ontología STEVO, para representar la información y ofrecer las interacciones adecuadas al usuario.

Se ha expuesto con detalle las instancias de los diferentes conceptos que intervienen, así como los valores de las propiedades y relaciones.

Además, se han expuesto capturas de pantallas del componente de visualización, para demostrar las posibilidades gráficas del control.

En la versión actual están desarrollados los agrupamientos por marcadores representantes, así como los dos tipos de comportamientos relacionales, los de control de tiempo y los de incerteza. Además, se ha desarrollado una interfaz de filtros, de visualización de información del objeto.

También se han desarrollado las escenas interactivas de navegación por mapas, y la navegación en ejes cartesianos en dos y tres dimensiones.

En el siguiente capítulo se va a evaluar los resultados de los trabajos realizados, tanto desde el punto de vista de usabilidad, como la resistencia del control a pruebas de estrés.

## CAPÍTULO 10. VALIDACIÓN DE LOS RESULTADOS: ARQUITECTURA E IMPLEMENTACIÓN

### 10.1. RESUMEN

En este capítulo se describen las pruebas realizadas a los desarrollos del marco de trabajo realizados en esta investigación.

Tal y como se describió en el capítulo anterior, uno de los resultados de esta investigación es la visualización de un mapa espacio temporal dentro del proyecto SILKNOW (Portalés, Sebastián et al. 2018).

En este proyecto, una de las tareas clave era la validación de los resultados del mismo. Las pruebas y resultados realizados sobre el mapa espacio temporal se encuentran en los entregables del proyecto:

- D.7.3. Usability evaluation by online users of the system.
- D5.7. Test report of Virtual Loom and deployable components.

La participación del autor de este trabajo en estos entregables fue muy activa, tanto en desarrollo, como preparación de pruebas y redacción del entregable.

Por este motivo, se incorporará de forma abreviada la parte del trabajo que ha sido realizada por el autor de esta investigación, aquella que está relacionada con el mapa espacio temporal, y de forma abreviada, como parte de este capítulo.

Las pruebas de validación y usabilidad del mapa espacio temporal consistirán en una pruebas basadas en el Sistema de Escala de Usabilidad (SUS) (Brooke 2013), que será aplicado a varios conjuntos de usuarios.

En lo referente a las pruebas de carga se han realizado con un servidor dedicado, realizando con diferentes cargas de datos y sistemas hardware, la medición de tiempos en los diferentes procesos de carga y visualización de datos.

Estas pruebas son la validación del marco de trabajo diseñado e implementado en esta investigación.

### 10.2. PRUEBAS DE ACEPTACIÓN Y USABILIDAD

#### 10.2.1. Introducción

Para la evaluación de la aceptación y usabilidad del mapa espacio temporal del proyecto SILKNOW se usará un método popular en la literatura para evaluar la usabilidad de un producto. El Sistema de Escala de Usabilidad (SUS) (Brooke 2013) , es un método que se utiliza desde hace décadas, y el gran número de cuestionarios recogidos con sus puntuaciones ofrece la posibilidad de analizar con un buen grado de fiabilidad los resultados de un análisis SUS. El SUS tiene muchas ventajas, una de las cuales es que ofrece las respuestas utilizando la escala Likert (Joshi, Kale et al. 2015) , lo que representa un alto grado de objetividad, pero aun así, como todos los métodos que utilizan cuestionarios en lenguaje natural, está sujeto a cierta ambigüedad y a la posibilidad de que el contenido de las preguntas no se comprenda del todo bien. También existe el riesgo de que las personas que desarrollen la prueba no entiendan bien las preguntas. Por este motivo,



tras recoger todas las respuestas a los cuestionarios se han aplicado técnicas para identificar las respuestas inconsistentes e incoherentes. Por esta razón se obtiene el parámetro alfas de Cronbach (Taber 2018) para evaluar la fiabilidad de los resultados.

La fase de ejecución de las pruebas ha sido llevada a cabo por el Departamento de Informática de la Universidad de Palermo, que forma parte del consorcio del proyecto SILKNOW.

Para evaluar las pruebas de aceptación y usabilidad de los mapas espacio temporales representados con la ontología STEVO se han llevado a cabo las siguientes actividades:

- Preparación de los cuestionarios.
- Selección de los usuarios.
- Preparación de los materiales de formación.
- Envío de los cuestionarios.
- Recogida y evaluación de los resultados

#### *10.2.2. Preparación de los cuestionarios*

Los cuestionarios realizados constan de tres partes: las preguntas de la escala SUS, las preguntas generales y un espacio de observaciones, o comentarios, que el entrevistado puede introducir si lo desea. Además, pueden proporcionar algunos datos personales, como su formación y edad, además de su declaración de consentimiento a realizar el cuestionario.

Las preguntas de la escala SUS están estructuradas de tal manera, que las preguntas con numeración impar tienen un sesgo positivo y las preguntas con numeración par tienen un sesgo negativo. Son las siguientes:

1. Creo que me gustaría utilizar este sistema con frecuencia.
2. El sistema me pareció innecesariamente complejo.
3. Me pareció que el sistema era fácil de usar.
4. Creo que necesitaría el apoyo de un técnico para poder utilizar este sistema.
5. Me pareció que las distintas funciones de este sistema estaban bien integradas.
6. Me pareció que había demasiadas incoherencias en este sistema.
7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente.
8. El sistema me pareció muy engorroso de utilizar.
9. Me sentí muy seguro utilizando el sistema.
10. Tuve que aprender muchas cosas antes de ponerme en marcha con este sistema.

Las preguntas generales complementan los cuestionarios SUS, y tienen como objetivo explorar la eficacia y de las herramientas por parte de los usuarios finales. Estas preguntas se han dividieron en tres grupos principales:

- Un primer grupo con el objetivo de Investigar la comodidad del usuario, basándose en términos de eficacia y fiabilidad de las funciones esperadas.



- El segundo grupo pretende que el usuario exprese su opinión sobre la herramienta.
- El tercer grupo busca conocer la opinión del usuario sobre la posible utilización de la herramienta en entornos concretos.

En la se Tabla 53 se exponen las preguntas realizadas en los cuestionarios de aceptación y usabilidad.

*Tabla 53 Preguntas de los cuestionarios de aceptación y usabilidad. Fuente: Adaptado de Deliverable D.7.3 de proyecto SILKNOW.*

|   |
|---|
| <b>Preguntas generales sobre el mapa espacio temporal.</b>  |
| <b>Grupo 1: Sobre la comodidad del usuario, en base a la eficacia y fiabilidad.</b>                                 |
| Puedo encontrar los objetos que busco de forma adecuada mediante los filtros.                                       |
| Los objetos se mostraron en el mapa de forma óptima.  |
| El tiempo de ejecución es apropiado.  |
| Encuentro fácilmente una forma de hacer las cosas.  |
| La información proporcionada es adecuada.   |
| La herramienta cumple con mis expectativas  |
| El sistema puede ser útil para el aprendizaje textil.   |
| <b>Grupo 2: Opinión sobre la herramienta.</b>   |
| Me gusta la representación lineal de la relaciones entre los objetos del mapa.                                      |
| Me gusta la representación de corona circular para expresar el porcentaje de relaciones entre los objetos del mapa. |
| Gracias a la visualización de datos en el mapa es fácil de comprender la relación existente entre los objetos.      |
| Los conceptos utilizados son apropiados para mi formación.  |
| Compartiré esta herramienta con compañeros y amigos.  |
| <b>Grupo 3: Opinión sobre su aplicación en entornos concretos.</b>  |
| ¿Dónde crees que podría usarse la herramienta en un museo?  |
| ¿Crees que la visualización interactiva de mapas puede ser útil en un museo?  |
| ¿Crees que la herramienta puede ser útil para la investigación en historia?   |
| ¿Crees que el mapa espacio temporal puede ser útil para la investigación en arte?                                   |
| ¿Crees que el sistema puede aplicarse para encontrar inspiración en diseños tradicionales?                          |
| Me gustaría usar el mapa espacio temporal en un museo cuando visito una ciudad.                                     |

### 10.2.3. Selección de los usuarios

Para identificar a las personas que debían participar en la evaluación, que se realizó durante dos meses, se recurrió a todos los posibles usuarios del proyecto. Se elaboró una lista de personas de los sectores de Patrimonio, Investigación y Educación, Industrias Creativas, Turismo y Medios de Comunicación. Además, se recurrió a usuarios del sector de las TIC (Tecnologías de la Información y Comunicación). Participaron personas de las cuatro lenguas utilizadas en el proyecto SILKNOW (Italiano, Francés, Alemán y Español)

Además, se llevaron a cabo tres sesiones de evaluación específicas por parte de los socios del proyecto:

- Universidad de Palermo. 60 estudiantes de los cursos de ingeniería de software ofrecidos por el Departamento de Ingeniería y unos 15 historiadores del arte de los cursos de humanidades.

- Por parte de Josef Stefan Institute, se involucraron 30 investigadores y estudiantes de investigación que trabajan en el área de Inteligencia Artificial.
- Por parte de la Universidad de Valencia, participaron 60 estudiantes de la asignatura de visualización de datos dentro del grado de Ciencia de Datos y del Máster de Patrimonio Cultural.

En la Figura 88 se muestra una serie de gráficos que indican la distribución de participantes en la evaluación en términos de género, lenguaje, formación y rango de edad.

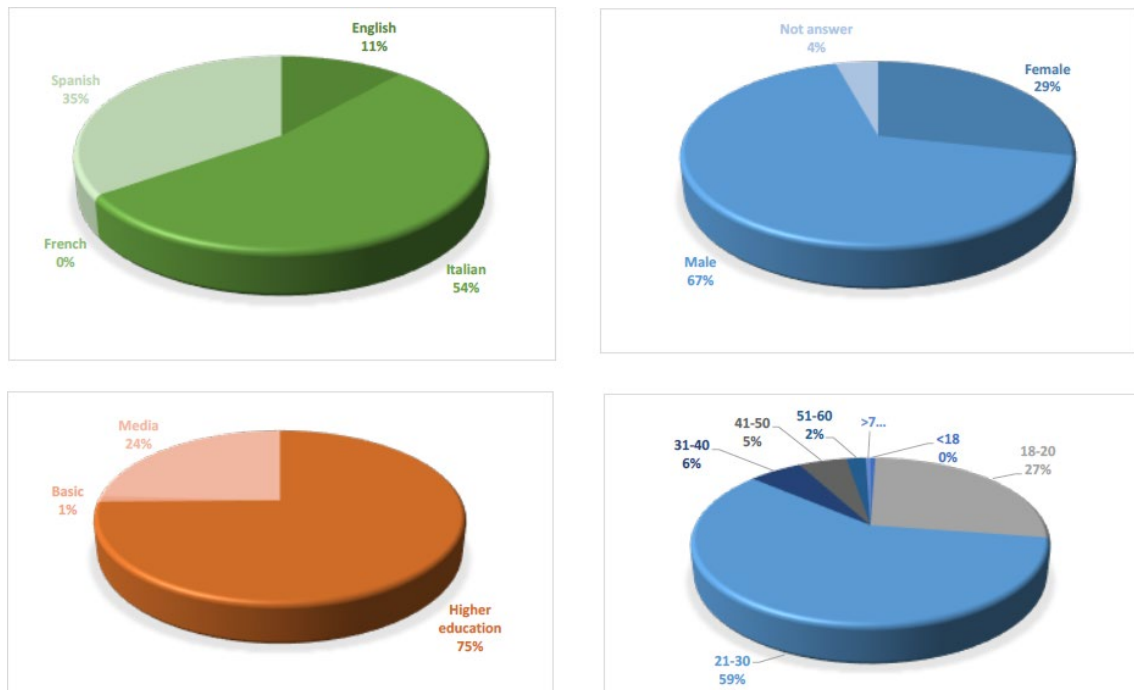


Figura 88 Distribución de los participantes de la evaluación por género, lenguaje, formación y edad. Fuente: Deliverable D7.3 proyecto SILKNOW

#### 10.2.4. Preparación de los materiales de formación y envío de los cuestionarios

Se creó una página web, donde los usuarios podrían encontrar en el lenguaje escogido materiales de formación sobre la herramienta a evaluar. A todos los usuarios se les pidió que leyeran los manuales y materiales de formación.

Se prepararon varios cuestionarios online, que se solicitó rellenar a los usuarios una vez habían completado la fase de formación.

#### 10.2.5. Evaluación de los resultados

La evaluación se realizó durante un periodo de aproximadamente 3 meses y debido a la situación de pandemia generada por la COVID-19, se decidió que fuera online.

A continuación, se exponen los resultados de la evaluación, en función del lenguaje de los entrevistados (español, inglés e italiano), ver Tabla 54, así como por el ámbito profesional de los entrevistados, ver Figura 89.

Teniendo en cuenta que la evaluación se realizó en marzo de 2021, cuando los desarrollos estaban en estado prototipo, y que una puntuación de 70 en la escala SUS es un buen dato, se concluye que los resultados de la evaluación han sido positivos, ya que, tanto en los entrevistados en español, como en inglés, los resultados fueron superior a 70, mientras que en

italiano la puntuación fue de 54,25. Muy baja en este último caso, y este sesgo baja la media, pero esto es debido probablemente a que los usuarios eran técnicos en informática, que se centraron más en los aspectos técnicos y seguramente tenían más altas expectativas, desde el punto de vista tecnológico, y acusan más las inconveniencias derivadas del estado prototipo del sistema.

Tabla 54 Resultados de las pruebas SUS por idioma de entrevista. Fuente: Deliverable D.7.3 de proyecto SILKNOW.

|  | TOTAL |
|--|-------|
| Número de respuestas                     | 126   |
| Puntuación SUS entrevistados en inglés.  | 71    |
| Puntuación SUS entrevistados en italiano | 54,25 |
| Puntuación SUS entrevistados en español  | 73,57 |
| Puntuación SUS total                     | 65,62 |

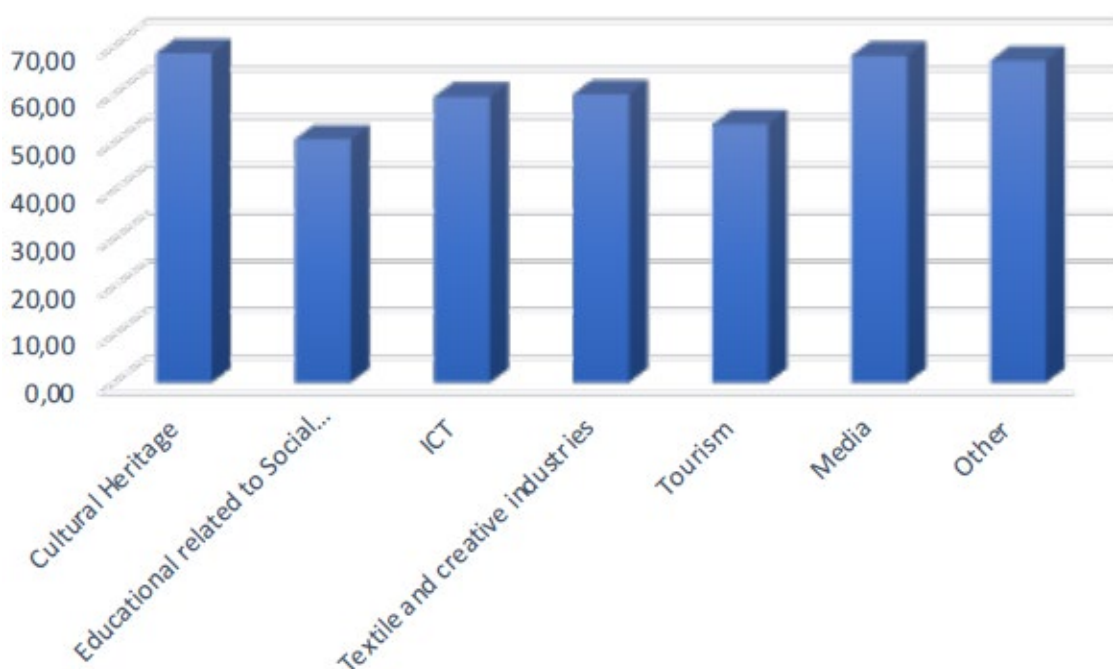


Figura 89 Resultados de pruebas SUS por ámbito profesional del entrevistado. Fuente: Deliverable D7.3 de proyecto SILKNOW.

En lo referente al ámbito profesional de los encuestados se observa que la media de las puntuaciones, por ámbito profesional, está entre 59 y 65. La puntuación más baja ha sido en el ámbito educativo, que llega a 49. Una vez más, teniendo en cuenta que las evaluaciones han sido realizadas sobre un prototipo, que entre otras cosas tenía un tiempo de carga demasiado elevado, se puede concluir que han sido positivas. Sin los problemas de funcionalidad derivados de la versión prototipo y tras desarrollar muchas de las pequeñas sugerencias, generalmente de interfaz de usuario, que en estos cuestionarios se dijeron, probablemente los resultados serían mejores.

### 10.3. PRUEBAS DE ESTRÉS

#### 10.3.1. Introducción

El objetivo de las pruebas de estrés del mapa espacio temporal es determinar su solidez a la hora de mostrar un enorme número de objetos, incluyendo sus relaciones. Para ello, se han preparado varios experimentos aumentando el número de objetos a considerar, lo que requeriría más recursos computacionales, y haciendo uso de diferentes sistemas hardware.

A continuación, se definen los experimentos durante los cuales se medirán diferentes parámetros. Estos parámetros servirán para evaluar el correcto funcionamiento de la herramienta, mostrando diferente número de objetos e interactuando con ellos, así como determinar cuáles son los límites de la herramienta.

Una vez realizados los experimentos se recogieron los datos y se exponen mediante gráficas y tablas. Para finalizar las pruebas, se analizan estos resultados y se redactan las conclusiones sobre ellos.

#### 10.3.2. Descripción de las pruebas

En el mapa espacio temporal, una vez que se cargan todos los objetos en la escena y se realiza la visualización de datos, no hay diferencias especiales entre su rendimiento cuando se aplica a diferentes conjuntos de datos. Esto se debe al proceso de agrupación y al hecho de que el número de objetos visualizados a la vez es independiente del número de objetos del conjunto de datos. Sin embargo, existen considerables diferencias en el contexto del proceso de carga, que depende del número de objetos contenidos en el conjunto de datos.

Por lo tanto, para realizar las pruebas de estrés de la herramienta de visualización, se van a tomar mediciones relacionadas con el proceso de carga de los datos y la generación de la visualización consistente en un mapa con agrupaciones de los datos. Para obtener una evaluación cuantitativa sobre la importancia del número de objetos, se preparan cuatro conjuntos de datos diferentes. Estos conjuntos de datos se diferencian en el número de objetos, y por lo tanto en el número de marcadores mostrados en el mapa y el número de visualizaciones de relaciones basadas en anillos (el número de relaciones anillo que se muestran en el mapa si se activa su visualización), como se indica en la Tabla 55 que muestra la información de los diferentes conjuntos de datos usados en la prueba de carga. Este número varía dependiendo del conjunto de datos, ya que los marcadores con anillos no aparecen en los marcadores de agrupación.

Tabla 55 Conjunto de datos de prueba para realizar las pruebas de carga de la herramienta. Fuente: Deliverable D.5.7 de SILKNOW.

| Conjunto de datos | Número de objetos | Núm. De marcadores | Núm. de marcadores relación anillo. |
|-------------------|-------------------|--------------------|-------------------------------------|
| 1                 | 301               | 342                | 26                                  |
| 2                 | 3001              | 3321               | 295                                 |
| 3                 | 15001             | 16286              | 1257                                |
| 4                 | 32249             | 34800              | 1600                                |

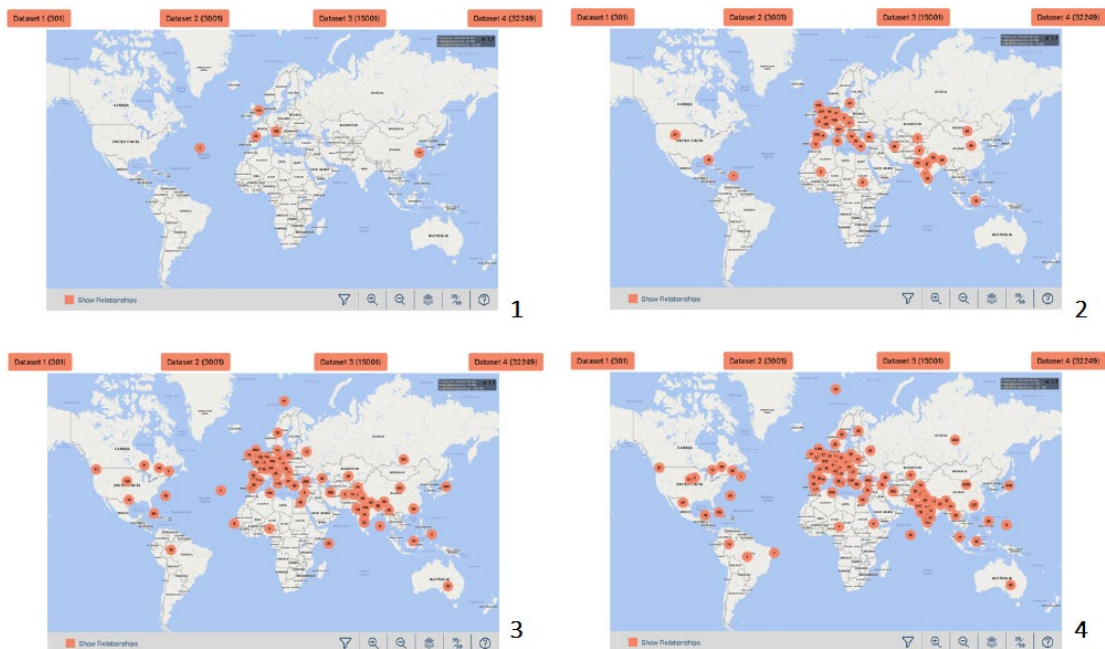
Para realizar las pruebas se ha implementado una página web de Evaluación de la herramienta (Sevilla, J., Casanova, P. 2021), accesible públicamente, que permite realizar las mediciones con los diferentes conjuntos de datos e interactuar con la herramienta de forma sencilla. En la Figura

90 se muestran varias capturas de pantalla de la herramienta de evaluación con los diferentes conjuntos de datos.

Aunque la herramienta de evaluación es la misma página web que se usó en el entregable D.5.7 de SILKNOW, la versión de la herramienta de visualización es más avanzada y se han corregido errores, a la vez que se han optimizado procesos. Las pruebas se han vuelto a repetir, sobre los mismos sistemas hardware, obteniendo nuevos y mejores datos. La descripción de los sistemas hardware usados en las pruebas de carga se detallan en la Tabla 56 .

*Tabla 56 Descripción de hardware usado en pruebas de carga*

| Ordenador   | Procesador             | Memoria             | Tarjeta gráfica |
|-------------|------------------------|---------------------|-----------------|
| Ordenador 1 | i5-3470 3.2 Ghz        | 4 Gb DDR2 1333 Mhz  | Nvidia GT 730   |
| Ordenador 2 | i7-920 2.6 Ghz         | 8 Gb DDR 1066 Mhz   | Nvidia GT 730   |
| Ordenador 3 | I5-6500 3.2 Ghz        | 8Gb DDR4 2133 Mhz   | AMD Radeon VII  |
| Ordenador 4 | Intel i7-7700HQ 2.8Ghz | 16 Gb DDR4 2666 Mhz | GeForce 1050 Ti |



*Figura 90 Capturas de pantalla de la herramienta de evaluación con los diferentes conjuntos de datos (1:301, 2:3001, 3:15001, 4: 32249)*

### 10.3.3. Parámetros a evaluar

Los parámetros que se van a evaluar durante las pruebas de carga de la herramienta de visualización son los siguientes:

- **Memoria usada:** La utilización de memoria siempre es un parámetro importante a tener en cuenta. La memoria es un recurso limitado, y si la aplicación sobrepasa el límite, puede bloquearse, o su rendimiento puede disminuir drásticamente. En un componente WebGL este parámetro es muy importante, ya que esta tecnología no libera la memoria no utilizada hasta que finaliza el proceso de dibujo del cuadro. En la herramienta de visualización, muchos procesos deben ejecutarse antes de que se dibuje la representación de datos. Además, si estos procesos requieren más que la memoria disponible, el componente se bloqueará.

- **Velocidad de refresco (Frame rate).** Este parámetro indica el número de veces que se redibuja el contenido del área de visualización por segundo. Este parámetro está directamente relacionado con la calidad de la experiencia del usuario, si es muy baja (menos de 15 FPS (Frames Per Second) se considera una mala experiencia de usuario. Si es superior a 30 FPS, la experiencia de usuario se considera buena.

- **Tiempo requerido para la realización de los procesos iniciales.** Para comenzar a visualizar los datos en la herramienta de visualización es necesario que se ejecuten un cierto número de procesos. Este tiempo es muy importante, y más en una aplicación web, que es el marco donde se va a ejecutar la herramienta. Este tiempo, a nivel general, según Nielsen (Nielsen 1994) es de 10 segundos el máximo tiempo que el usuario puede esperar una respuesta de un sistema informático, a partir de este tiempo ya se considera intolerable. Pero estudios más recientes (Nah 2004), vinculados con la interacción a través de la web, concluyen que a partir de 2 segundos de espera ya comienza a generar una experiencia desagradable en el usuario. Los tiempos que se van a medir en cada ejecución de la prueba son:

- **Adquisición y procesamiento de datos.** Consiste en el tiempo de obtener los datos del grafo de conocimiento y estructurarlos internamente para que puedan ser procesados por el componente de visualización.
- **Generación de marcadores.** Tiempo de generación de estructuras gráficas para la representación de los objetos y agrupamientos.
- **Procesamiento de agrupamientos.** Tiempo de ejecución del algoritmo de clustering.
- **Procesamiento de relaciones (general).** Tiempo de procesamiento para la creación de estructuras de datos asociadas a los objetos para agilizar la visualización de relaciones.
- **Procesamiento de relaciones de anillo.** Tiempo de procesamiento para obtener datos y crear estructuras gráficas para representar los datos y elementos gráficos de las relaciones anillo.

#### *10.3.4. Resultados de las pruebas*

Después de ejecutar las pruebas de carga, los datos obtenidos de las diferentes pruebas con los diferentes sistemas hardware y conjuntos de datos, referentes a elementos gráficos a representar (objetos, marcadores, relaciones anillo) , recursos de memoria y velocidad de refresco, se muestran en la Tabla 57.

En la Tabla 58, para cada uno de los ordenadores, con los diferentes conjuntos de datos, se muestran los datos de tiempo de ejecución para los procesos iniciales (Procesamiento de datos, generación de marcadores, proceso de agrupación, proceso de relaciones general y de marcadores de anillo). Estos datos son los valores medios, para cada combinación de conjuntos de datos y ordenador, las pruebas se han repetido cinco veces. El valor de cada celda de la tabla se corresponde con el valor medio obtenido durante las cinco pruebas.

En las dos tablas se aprecian que hay unas celdas con asteriscos, en lugar de valores numéricos. Se trata de las pruebas de carga realizadas con los ordenadores 1, 2 y 3, para el conjunto de datos 4, el de mayor volumen. En esta combinación de ordenador y conjunto de datos, el proceso se ha interrumpido por problemas de memoria durante el procesamiento de relaciones.

Tabla 57 Recursos y velocidad obtenidos para las pruebas de carga de herramienta de visualización.

| Ordenador | Conjunto de datos | Objetos | Marcadores | Relaciones Anillo | Memoria Usada (MB) | Memoria Reservada (MB) | FPS |
|-----------|-------------------|---------|------------|-------------------|--------------------|------------------------|-----|
| 1         | 1                 | 301     | 342        | 26                | 15                 | 23                     | 75  |
| 1         | 2                 | 3001    | 3321       | 295               | 27,8               | 86                     | 75  |
| 1         | 3                 | 15001   | 16286      | 1257              | 92,4               | 280                    | 75  |
| 1         | 4                 | 32249   | 34800      | *****             | ***                | ***                    | *   |
| 2         | 1                 | 301     | 342        | 26                | 15                 | 24                     | 60  |
| 2         | 2                 | 3001    | 3321       | 295               | 27,8               | 86                     | 60  |
| 2         | 3                 | 15001   | 16286      | 1257              | 91,6               | 282                    | 60  |
| 2         | 4                 | 32249   | 34800      | *****             | ***                | ***                    | *   |
| 3         | 1                 | 301     | 342        | 26                | 16                 | 24                     | 120 |
| 3         | 2                 | 3001    | 3321       | 295               | 28,8               | 86                     | 120 |
| 3         | 3                 | 15001   | 16286      | 1257              | 87,4               | 283                    | 120 |
| 3         | 4                 | 32249   | 34800      | *****             | ***                | ***                    | *   |
| 4         | 1                 | 301     | 342        | 26                | 16                 | 24                     | 120 |
| 4         | 2                 | 3001    | 3321       | 295               | 29,6               | 87                     | 120 |
| 4         | 3                 | 15001   | 16286      | 1257              | 78                 | 284                    | 120 |
| 4         | 4                 | 32249   | 34800      | 1600              | 145                | 570                    | 120 |

Tabla 58 Tiempo de procesos iniciales obtenidos para las pruebas de carga de la herramienta de visualización

| Ordenador | Conjunto de datos | Procesar datos (s) | Generación Marcador (s) | Proc. Agrupación (s) | Proc. Relaciones (s) | Proc. Rel.Anillo (s) |
|-----------|-------------------|--------------------|-------------------------|----------------------|----------------------|----------------------|
| 1         | 1                 | 0,33               | 0,13                    | 0                    | 0,03                 | 0                    |
| 1         | 2                 | 1,92               | 0,62                    | 0,01                 | 0,53                 | 0,04                 |
| 1         | 3                 | 8,14               | 2,37                    | 0,08                 | 4                    | 0,15                 |
| 1         | 4                 | 16,61              | 5,36                    | 0,3                  | *****                | *****                |
| 2         | 1                 | 0,72               | 0,24                    | 0                    | 0,05                 | 0,01                 |
| 2         | 2                 | 2,53               | 0,65                    | 0,01                 | 0,55                 | 0,05                 |
| 2         | 3                 | 10,7               | 3,16                    | 0,09                 | 4,68                 | 0,2                  |
| 2         | 4                 | 21,98              | 6,77                    | 0,31                 | *****                | *****                |
| 3         | 1                 | 0,2                | 0,14                    | 0                    | 0,02                 | 0                    |
| 3         | 2                 | 0,93               | 0,39                    | 0,01                 | 0,36                 | 0,02                 |
| 3         | 3                 | 4,49               | 1,69                    | 0,06                 | 3,11                 | 0,09                 |
| 3         | 4                 | 8,61               | 3,78                    | 0,24                 | *****                | *****                |
| 4         | 1                 | 0,2                | 0,11                    | 0,002                | 0,01                 | 0,002                |
| 4         | 2                 | 0,95               | 0,357                   | 0,006                | 0                    | 0,02                 |
| 4         | 3                 | 3,89               | 1,356                   | 0,047                | 0                    | 0,07                 |
| 4         | 4                 | 7,34               | 2,28                    | 0,16                 | 0,01                 | 0,09                 |



#### Resultados en tiempos de procesamiento inicial

Con la información contenida en la Tabla 57 y en la Tabla 58 , en lo referente a los datos de tiempo de ejecución en los procesos iniciales, se obtienen los siguientes gráficos:

- Gráfico de obtención de datos y procesamiento interno (Figura 91).
- Gráfico de generación de marcadores (Figura 92).
- Gráfico de procesamiento de agrupaciones (Figura 93).
- Gráfico de estructuras de relaciones (Figura 94).
- Gráfico de procesamiento de relaciones anillo (Figura 95).
- Gráfico de tasa de refresco (Figura 96).
- Gráficos de uso de memoria (Figura 97 y Figura 98).

A continuación, se exponen y se detallan los procesos relacionados con dichos gráficos, así como el contenido de los mismos.

#### Obtención de datos y procesamiento interno.

En la Figura 91 se muestra el gráfico que indica para cada ordenador usado para realizar las pruebas (Ordenador 1, Ordenador 2, Ordenador 3 y Ordenador 4) el tiempo de ejecución invertido para la obtención de cada uno de los conjuntos de datos (1, 2, 3 y 4) desde el grafo de conocimiento y su procesamiento interno.

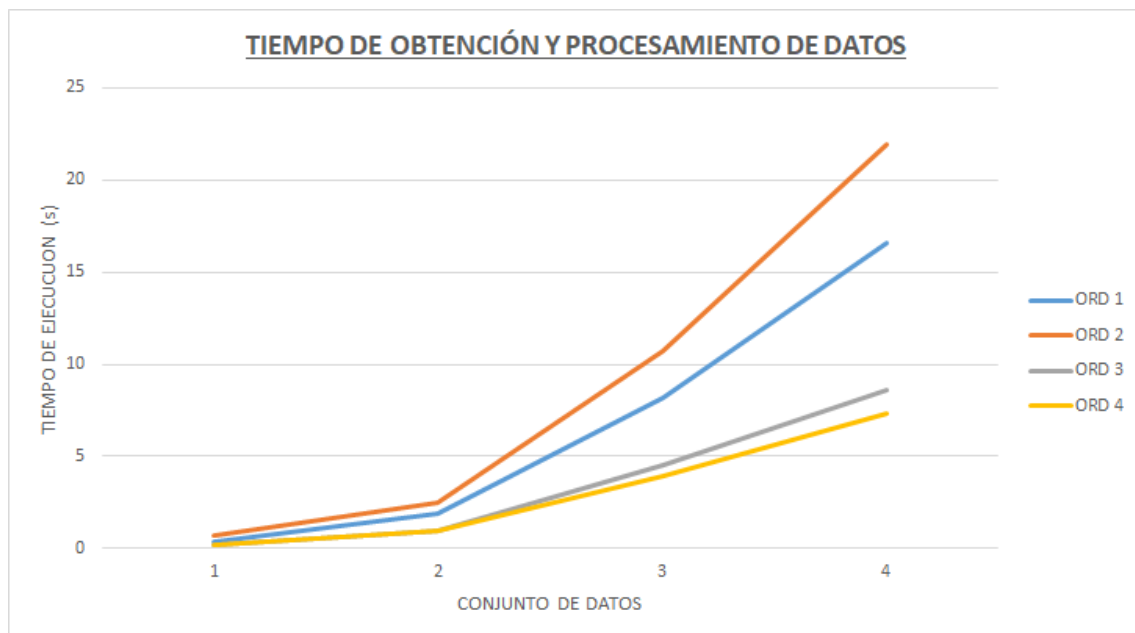


Figura 91 Gráfica con los tiempos de ejecución para la obtención y procesamiento de datos en las pruebas de carga.

Como era de esperar, los resultados de esta gráfica son peores conforme el hardware de procesamiento del ordenador es menos potente. Ordenador 2 es el equipo con peor capacidad de procesamiento, aunque tiene más memoria que el Ordenador 1, por eso los tiempos son peores para el Ordenador 2. Los ordenadores con mejor tiempo son los que tienen una capacidad de procesamiento más alta, el Ordenador 3 y el Ordenador 4.



Aunque esta es la tarea de mayor coste temporal, ya se observa desde el tercer conjunto de datos, incluso en el mejor de los casos, el tiempo es superior a los 2 segundos, que es el tiempo, a partir del cual el usuario comienza a empeorar su experiencia, debido al tiempo de espera. Esto teniendo en cuenta que aún quedan más procesos, indica que habrá que realizar algún tipo de acción para dar algún feedback al usuario de que el sistema está trabajando, además de algún tipo de información sobre los objetos, si es posible.

En el último conjunto de datos, el de mayor cantidad de información, la situación es más grave. El tiempo de espera es muy superior a 10 segundos en los dos ordenadores de peor capacidad de procesamiento (Ordenador 1 y Ordenador 2), tiempo de espera intolerable para el usuario.

#### Generación de marcadores.

En la Figura 92 se muestran los tiempos de ejecución para el proceso de generación de marcadores gráficos de los objetos. En este proceso se crean estructuras de procesamiento gráfico, con lo cual además de la potencia de procesamiento, también influye la presencia de componente hardware gráfico.

Esta información se ve perfectamente reflejada en el gráfico, donde los tiempos con los conjuntos de datos 3 y especialmente en el 4 ya están muy diferenciados. Siendo este tiempo menor conforme la capacidad de procesamiento, y la potencia de hardware gráfico adicional es más alta. Por ejemplo, en el conjunto de datos 4, el Ordenador 4 que tiene mejores prestaciones en estas áreas, el tiempo es sensiblemente inferior a los 2 segundos, mientras que el Ordenador 3 es de casi 4 segundos, cercano al doble. Por otra parte, el ordenador 2, el de peores prestaciones, es cercano a 7 segundos, el triple de tiempo que el Ordenador 4.

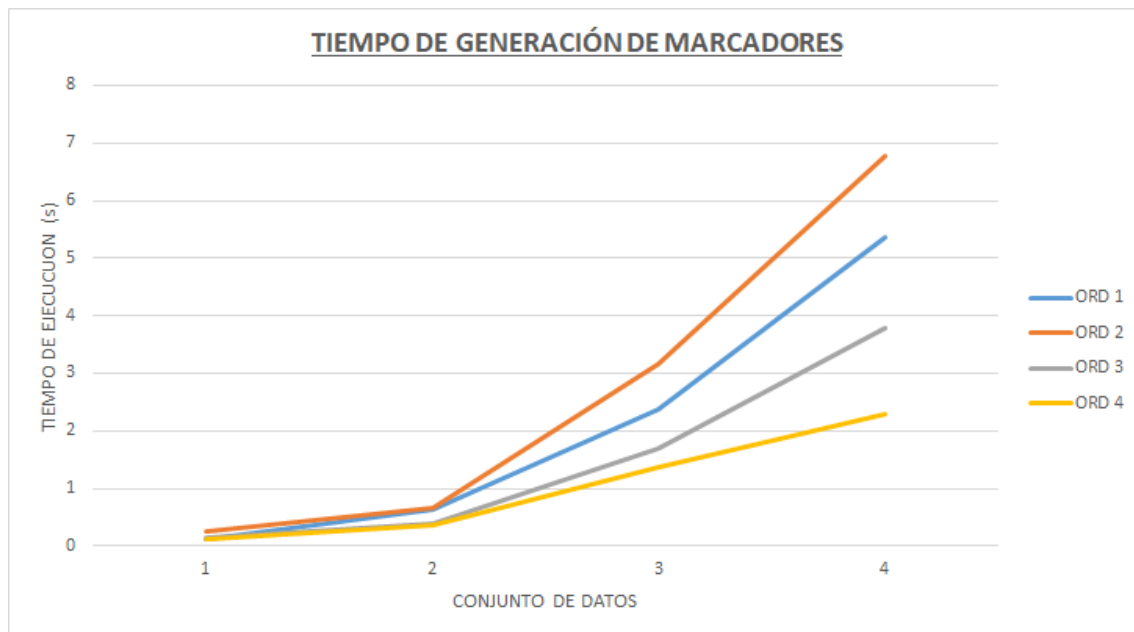


Figura 92 Gráfica con tiempos de ejecución del proceso de generación de marcadores gráficos en las pruebas de carga.

### Procesamiento de agrupaciones

El tiempo empleado para generar las agrupaciones durante las pruebas de carga no es muy elevado, llegando a mediar tiempos de 0,3 segundos en el peor de los casos, el ordenador con menos recursos hardware, con el conjunto de datos más grande.

En la Figura 93 se muestra una gráfica con estos tiempos, para cada ordenador y conjunto de datos. En ella se aprecia, al igual que en el resto de tiempos hasta ahora analizados, que este dato es principalmente función de la capacidad de procesamiento del ordenador. En el conjunto de 150001 datos todos los sistemas están por debajo de 0,1 segundos, pero en el mayor, sólo ordenador con más recursos queda cerca de 0,15 segundos y el resto obtiene tiempos de 0,25 segundos, o mayores.

Los tiempos obtenidos muestran que al algoritmo empleado, basado en el algoritmo de Marker Cluster (Leaver 2012), es muy eficiente. Si en su lugar se usa un algoritmo que analiza la posición de cada punto, respecto a todos los puntos, supondría un coste de orden cuadrático ( $O(n^2)$ ). Obteniéndose tiempos muy superiores a los actuales, ya que en este caso el algoritmo es de coste  $O(n) + k$ , como se observó en los primeros desarrollos. Además, no ha habido ningún problema para entender el funcionamiento de las agrupaciones en las pruebas de aceptación y usabilidad. De estos resultados se deduce que no sólo es rápido, sino que también da unos resultados aceptables y fáciles de comprender para los usuarios.

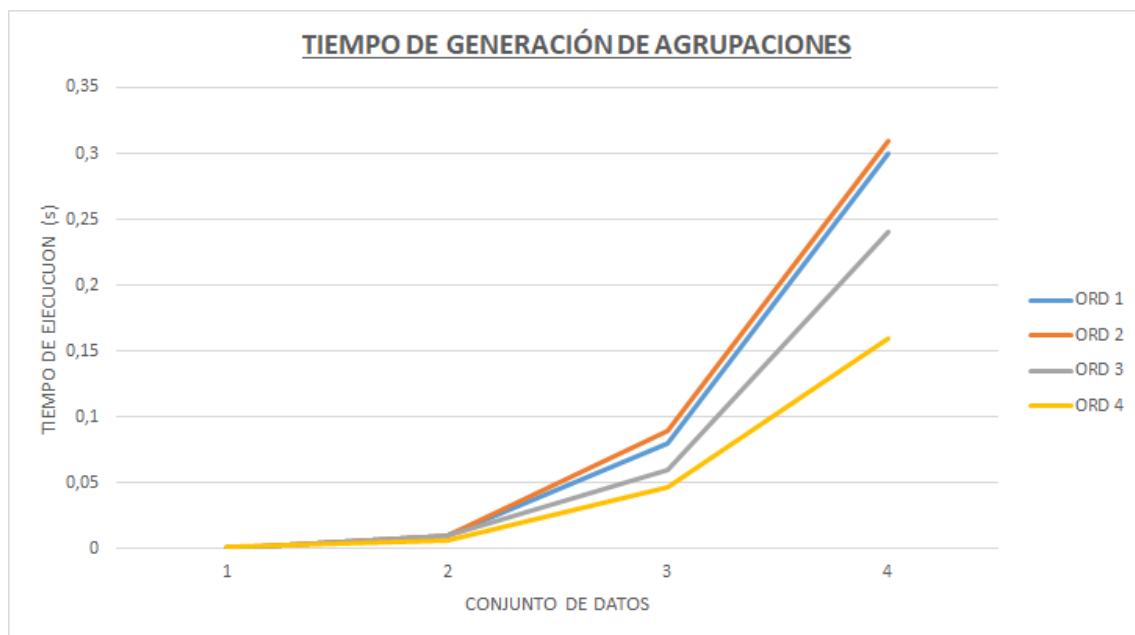


Figura 93 Gráfica con tiempos de ejecución del proceso de generación de agrupaciones.

### Generación de estructuras de relaciones.

Una vez procesados los datos de los objetos, se crean estructuras internas para optimizar el proceso de búsqueda de objetos que tengan el mismo valor en sus propiedades y así obtener resultados lo más rápidos posibles. La generación de estas estructuras se ha mejorado mucho en las últimas versiones de la herramienta de visualización y los tiempos son muy pequeños. El problema de la obtención de estos datos es debido a un problema con la gestión de memoria de WebGL, que se comentará en el análisis del parámetro de consumo de memoria. La ejecución de la herramienta se interrumpe en los ordenadores con menos memoria.

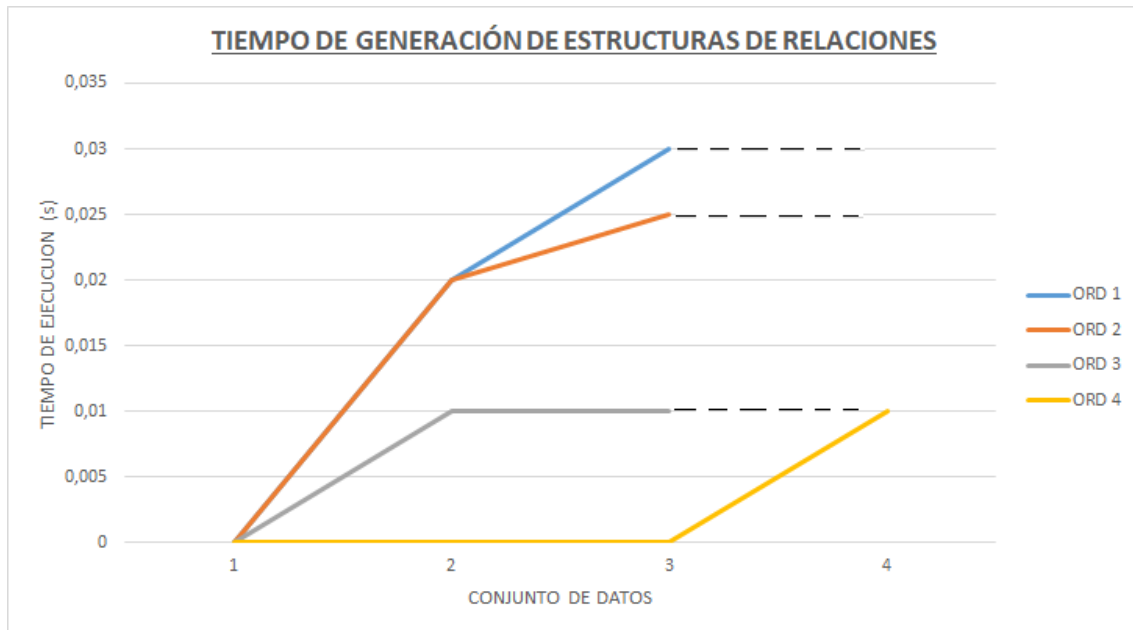


Figura 94 Gráfica con tiempos de ejecución para la generación de estructuras de relaciones durante las pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.

Por este motivo, en el conjunto de datos más grande, el 4, sólo existen datos de tiempo en el ordenador 4, que tiene más recursos de memoria, en el resto de ordenadores, la ejecución se interrumpe durante este proceso. En la Figura 94 se muestran los tiempos de procesos para este dato. Son tiempos que en ningún caso superan los 0.03 segundos.

Los tiempos son pequeños, pero se ve claramente su relación con la capacidad de procesamiento del ordenador y con el número de datos del conjunto.

#### Procesamiento de relaciones anillo.

En la Figura 95 se muestra una gráfica de los tiempos de ejecución para crear las estructuras que permitan representar las relaciones de anillo. En esta gráfica se aprecia claramente su relación con los recursos de procesamiento del ordenador y del tamaño del conjunto de datos.

Debido al problema de la memoria, al igual que en el parámetro anterior, la ejecución se interrumpe y sólo hay datos para el conjunto de datos 4 en el ordenador con más recursos de memoria.

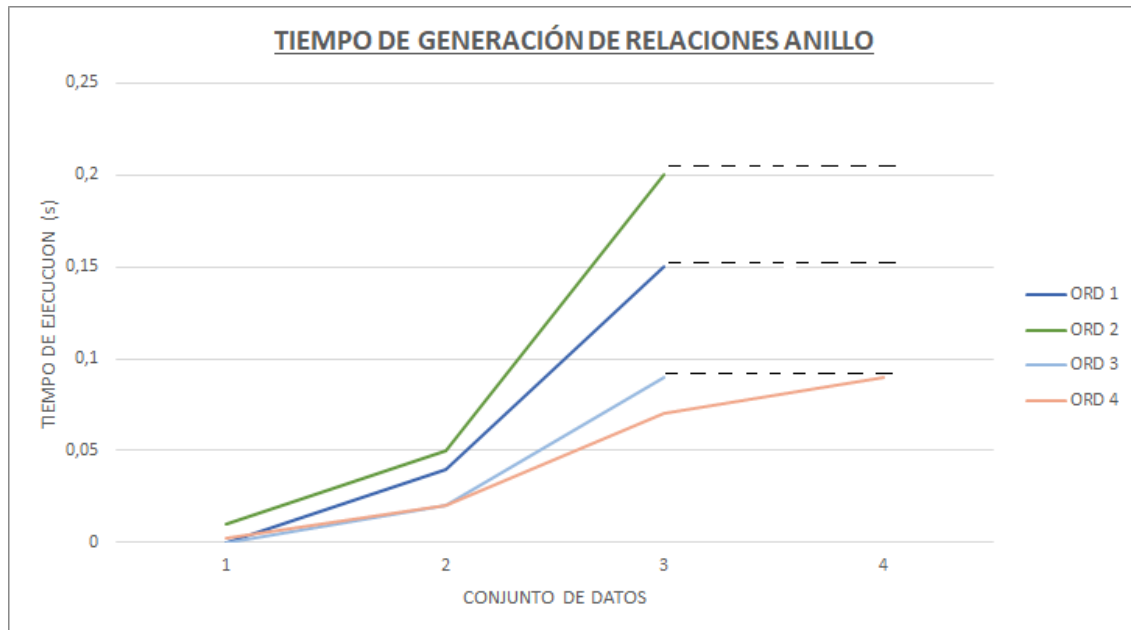


Figura 95 Gráfico con los tiempos de ejecución para generar relaciones anillo en pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.

#### Tasa de refresco (FPS)

La velocidad de refresco de la visualización realizada por la aplicación, es un dato muy importante para evaluar la aceptabilidad y usabilidad de una aplicación. Este dato está directamente relacionado con la fluidez de la visualización y por lo tanto con la experiencia del usuario.

Esta velocidad, medida en cuadros, o frames, por segundo (FPS) en la herramienta de visualización es siempre igual, o superior a los 60 FPS. Esta velocidad según estudios realizados es considerada por la mayoría de los usuarios como adecuada para obtener una buena experiencia de usuario, incluso aunque suponga reducir sensiblemente la resolución (Debattista, Bugeja et al. 2018).

En la Figura 96 se representa una gráfica con los datos de esta velocidad en FPS obtenida durante las pruebas de carga. Debido a la interrupción de la aplicación en el conjunto de datos mayor en los ordenadores 1, 2 y 3, no hay información para éstas pruebas. De la información que se dispone se puede concluir que hay como dos niveles de velocidad, uno sobre los 60 FPS en el caso de los ordenadores 1 y 2, que tienen menos capacidad procesamiento y hardware gráfico, y de 120 FPS en los ordenadores 3 y 4, con mayor capacidad de procesamiento y presencia de hardware gráfico de mayor potencia.



Figura 96 Gráfica con los resultados de velocidad de refresco de la aplicación en cuadros por segundo (FPS) obtenida durante las pruebas de carga. Las líneas punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.

#### Memoria usada.

Tal y como se ha visto en los resultados de las pruebas, el conjunto de datos 4 no ha podido terminado de ser cargado en 3 de los ordenadores, y por lo tanto no se tienen datos de todo el proceso. Esto es debido a un problema de memoria en los ordenadores con menos recursos en memoria. Este error genera una interrupción de la aplicación.

Esto es debido a que la memoria consumida durante el proceso de carga es mayor que la permitida por el sistema. Aunque la memoria ha sido debidamente deliberada a nivel de aplicación, en Unity, WebGL no libera la memoria hasta finalizar el proceso de dibujado. Como Unity genera un componente WebGL, la aplicación final tiene esta limitación de WebGL. En el proceso de carga, existen procesos que consumen memoria, y que su finalización es requerida para terminar el proceso de dibujado, su memoria no se libera y termina la aplicación. Este proceso se ejecuta antes de que finalice el proceso de dibujo y, por tanto, WebGL no puede liberar la memoria adecuadamente. Como consecuencia el proceso se bloquea antes de que el sistema de pruebas registre la memoria realmente utilizada por todo el proceso.

Debido a este problema, al igual que pasaba con datos anteriores, los datos del conjunto 4 no son registrados en los tres primeros ordenadores.

En la Figura 97 y en la Figura 98 se muestran gráficas de la memoria reservada para realizar la visualización y la memoria finalmente usada.

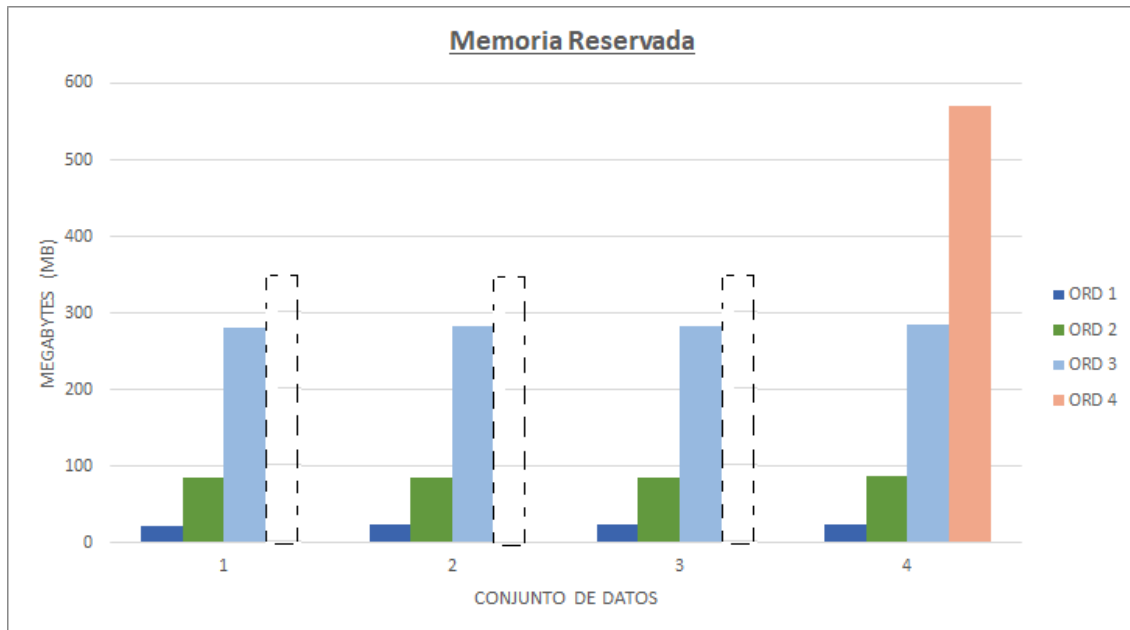


Figura 97 Memoria reservada por el componente de visualización durante las pruebas de carga. Las barras punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.

En la gráficas representadas en la Figura 97 y en la Figura 98 se aprecia el comportamiento esperable del sistema, a más datos a representar, más memoria se reserva y se utiliza. Esto es debido a que la memoria se usa tanto para guardar información de los datos, como de las estructuras gráficas a representar.

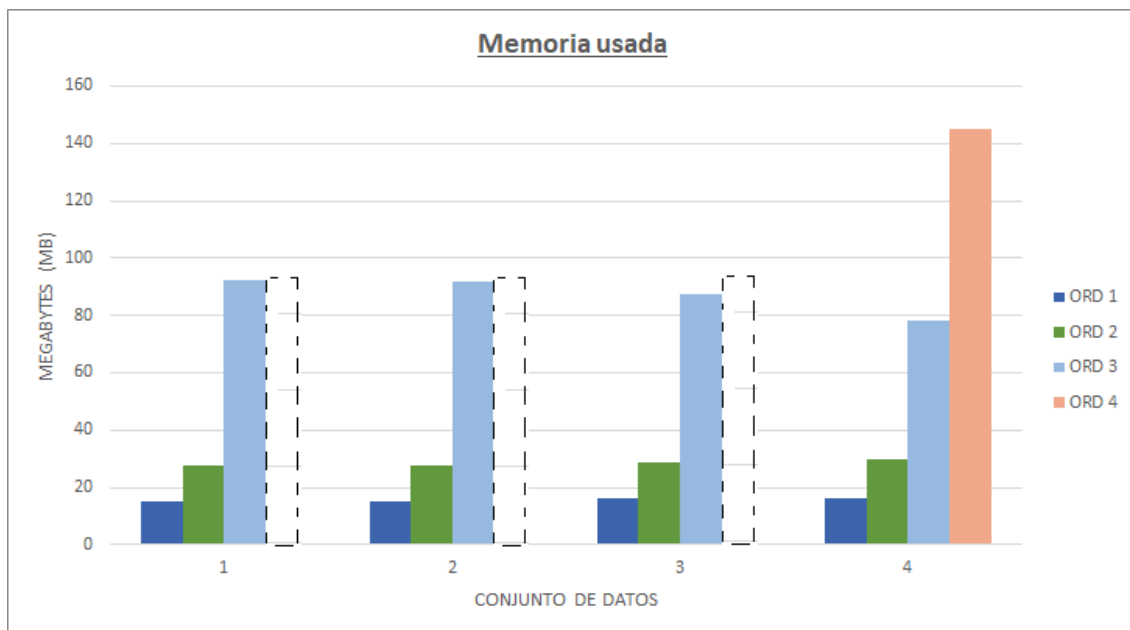


Figura 98 Memoria usada por el componente de visualización durante las pruebas de carga. Las barras punteadas indican que no hay datos para ese ordenador en ese conjunto de datos.

#### 10.3.5. Conclusiones de los resultados

Después de analizar los datos obtenidos durante las pruebas de carga, se observa que la mayoría de parámetros dependen principalmente de la capacidad de procesamiento del ordenador.

Algunos datos, como el FPS, la generación de marcadores, o generación de relaciones anillo también se ven afectados por la potencia de hardware gráfico presentes en los ordenadores.

Los principales problemas observados durante las pruebas de carga son:

- El tiempo de carga. Dado el tipo de proceso que es, seguramente el usuario puede tolerar más de dos segundos, que es el tiempo de carga en el que el usuario comienza a estar molesto con el rendimiento, pero desde luego es intolerable que llegue a 10 segundos, y en opinión del autor de esta investigación debería de ser como mucho 5, o 6 segundos en el peor de los casos. En la Figura 99 se muestra una gráfica con el tiempo total invertido en el proceso de carga, la suma de los tiempos del procesamiento inicial. En esta gráfica se observa que el tiempo invertido en los dos primeros conjuntos de datos, está por debajo del umbral de 6 segundos en todos los ordenadores. Sin embargo, en el tercer conjunto de datos, sólo el ordenador 3 y 4 están por debajo de ese umbral, y el 1 y 2 superan el tiempo intolerable para el usuario. En el conjunto de datos 4, el tiempo no es correcto para los tres primeros ordenadores, ya que hay algunos parámetros cuyos tiempos no se han sumado, pero debido a su pequeño tamaño no alteran considerablemente el total. Con esta aclaración, el tiempo de todos los ordenadores es superior al considerado aceptable por el autor de esta investigación. Ese conjunto de datos tiene un total de 32243 datos, si bien no es pequeño, no es ni mucho menos enorme para el conjunto de datos posibles que pueden manejarse en patrimonio cultural. Debido a esto es necesario reducir el tiempo de carga para que sea admisible en una aplicación web, por lo menos si se espera manejar conjuntos de datos mayores de 15.000 datos.
- La memoria usada. Tal y como se ha apreciado en los datos, con ordenadores de 8 Gb o menos de memoria RAM el conjunto de datos 4 no ha sido capaz de ejecutarse, por problemas de memoria derivados de la liberación de memoria de WebGL. La situación es análoga al problema anterior. Si el conjunto de datos es superior a 15.000 datos, es probable que el problema se reproduzca y se interrumpa la ejecución del componente.

Con toda esta información, el autor de esta investigación, recomienda usar la actual herramienta de visualización basada en STEVO con conjuntos de datos no superiores a 15.000 datos. Con este volumen de información, el sistema tendría tiempos aceptables y un buen funcionamiento, incluso en equipos de bajas prestaciones.

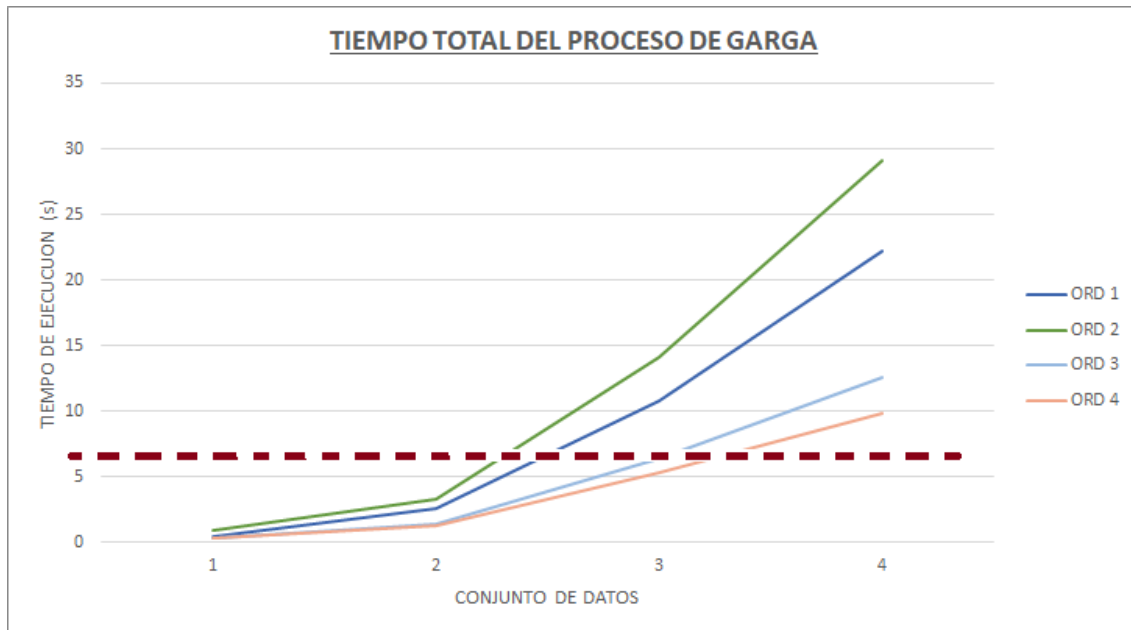


Figura 99 Gráfica con el tiempo total del proceso de carga del componente de visualización, durante las pruebas de carga. La línea roja punteada indica el umbral a partir del cual no es tolerable más tiempo, en opinión del creador de esta investigación.

#### 10.4. CONCLUSIONES

En este capítulo se han presentado los resultados de dos tipos de pruebas sobre el resultado del marco de trabajo diseñado e implementado en esta investigación. En concreto, el resultado es un mapa espacio temporal embebido en el portal del proyecto SILKNOW, que mantiene datos sobre tejidos de seda del último milenio.

La primera prueba consiste en una prueba de aceptación y usabilidad basada en la escala SUS, aplicada a colectivos de diferentes ámbitos profesionales y formación de diferentes países. Estas pruebas se aplicaron sobre la herramienta cuando aún estaba en estado prototipo. Esto fue así para buscar y obtener mejoras de la herramienta con tiempo de reacción en el proyecto, además de encontrar otros problemas.

A pesar del estado prototipo de la herramienta, el resultado está muy cercano a una herramienta que ofrece una buena experiencia de usuario, esto teniendo en cuenta que uno de los problemas que tenía el prototipo era un importante retraso en la carga de la visualización. Este hecho, junto con imperfecciones en la interfaz de usuario, son problemas que condicionan la opinión de los usuarios. Contando con estos inconvenientes los resultados fueron altos. En la versión actual, se ha reducido muchísimo el tiempo de carga y se ha corregido y mejorado la interfaz de usuario, de acuerdo con muchas de las propuestas de los usuarios. El autor de esta investigación cree que se puede asumir que el resultado sería mejor si se repitieran las pruebas en la versión actual, más avanzada.

La segunda prueba consiste en un conjunto de pruebas de carga consistentes en ejecutar la aplicación sobre equipos informáticos con diferentes capacidades hardware (memoria, procesamiento, hardware gráfico) y medir parámetros que permitan evaluar su funcionamiento. En concreto se tomaron los parámetros de memoria usada y reservada, tasa de refresco de la aplicación y tiempo de inicialización de varios procesos de arranque. Además, en estas pruebas los conjuntos de datos usados eran de diferentes tamaños, desde 300 objetos, a más de 30.000 objetos.



El resultado de estas pruebas de carga, como era de esperar, es que el sistema funciona con más rapidez y fluidez con sistemas de mejores prestaciones. Los principales problemas detectados fueron el tiempo de carga y problemas de ejecución relacionados con la memoria en el conjunto con mayor número de datos. A nivel de rendimiento gráfico el sistema mostró siempre velocidades de 60 FPS, o superior, para todos los ordenadores, llegando a 120 FPS en el caso de los ordenadores con mejores prestaciones y hardware gráfico.

En equipos de prestaciones medias y bajas, el tiempo de carga supera ligeramente los 10 segundos, que es el máximo tiempo de respuesta tolerable para un usuario medio, en lo referente a la respuesta de una aplicación informática con conjuntos de 15.000 datos.

Además, el conjunto con mayor número de datos sólo consiguió ejecutarse en el equipo de mejores prestaciones. El consumo de memoria era tan alto, que debido a la forma de liberar memoria que tiene WebGL, dependiente del final del proceso de dibujado, ocasionaba que se interrumpiera el proceso de ejecución en el resto de equipos.

Estas pruebas confirman lo detectado en el capítulo anterior de esta investigación. La herramienta, tal y como está implementada, consume demasiados recursos y debe de optimizarse tanto la carga, como la representación de información. Se delega demasiada carga en el módulo gráfico. Mucha de esa carga debe de realizarse en el componente de visualización desarrollado en Javascript, y solicitar datos bajo demanda a través de una API, ahora mismo demasiado reducida.

En cualquier caso, la herramienta en su actual versión, funciona perfectamente con conjuntos de datos no superiores a 15.000 datos en la mayoría de equipos informáticos, navegadores web y sistemas operativos.

Por lo tanto, se puede concluir que se ha conseguido desarrollar una herramienta de visualización multiplataforma, que es capaz de visualizar el contenido de un grafo de conocimiento con datos de patrimonio cultural, atendiendo a las especificaciones contenidas en una ontología basada en STEVO. El resultado genera una buena experiencia para los usuarios y se alcanza la finalidad buscada.

## SECCIÓN V. CONCLUSIONES, DISCUSIÓN Y TRABAJO FUTURO.

## CAPÍTULO 11. RESULTADOS Y DISCUSIÓN

### 11.1. RESUMEN

En este capítulo se presenta un recopilatorio de resultados, tanto a nivel de aplicaciones, entendidas como aportaciones a la generación de nuevo conocimiento, y también se citan otros resultados obtenidos desde el proceso de investigación realizado. A continuación, se describe una discusión con los diferentes resultados obtenidos y las aportaciones realizadas. Para finalizar, se termina con las conclusiones obtenidas.

### 11.2. RESULTADOS Y APORTACIONES

Tomando como base el estudio del estado del arte realizado en diversas disciplinas, así como el trabajo de investigación desarrollado en esta tesis doctoral, las principales contribuciones, consecuencia de este trabajo, son las siguientes:

- En el estado del arte de patrimonio cultural, se ha descrito una situación en la que las administraciones han detectado la necesidad de mantener el patrimonio cultural. Este hecho ha sido especialmente evidente en la última década, donde se han realizado multitud de inversiones en proyectos de digitalización, y soporte de información en grafos de conocimiento, así como todo tipo de herramientas relacionadas con este ámbito. Sin embargo, en la actualidad, aunque hay un claro modelo, que es CIDOC-CRM, para representar este tipo de información, aún existen varios modelos para hacerlo, y no todos ofrecen la posibilidad de un intercambio de datos sencillo. A pesar de que estos modelos llevan varios años actualizándose, hay temas clave en el patrimonio cultural que no están representados, como es la incerteza de los datos. Otras propiedades, como el tratamiento de los lugares y el tiempo, en los que ha estado relacionado un objeto, desde su producción hasta el momento actual, tampoco existe un consenso sobre su representación. Además, existen varios tesauros, o vocabularios, sin unificar. Estos vocabularios son clave, para facilitar el intercambio de información entre uno y otro modelo, y mientras exista tanta variedad va a ser complejo realizar este intercambio. Además, pese al esfuerzo de la administración, hay muchos objetos que no están disponibles públicamente, por varios motivos, y una gran cantidad que ni siquiera están digitalizados. Esta situación es grave, en un momento de incertidumbre económica, como es el actual, que está provocando el cierre de algunos museos importantes, y muchos otros pequeños, además de que el mantenimiento de este patrimonio en muchas ocasiones está en grave peligro. Por lo tanto, para resumir, podría decirse que este trabajo contribuye, indicando que la situación ha mejorado mucho en la última década, pero aún hay mucho trabajo por hacer, y es importante hacerlo cuanto antes, si se quiere preservar este patrimonio.
- Analizando el estado del arte sobre visualización de grafos de conocimiento, al menos desde el conocimiento del autor de este trabajo, tras realizar una exhaustiva búsqueda bibliográfica, además de aplicar sistemas de visualización de grafos, sólo ha encontrado dos trabajos con un mínimo de entidad, desarrollo y argumentación científica, sobre el diseño de una ontología de visualización. Por una parte, la ontología VISO, y por otra parte la ontología VUMO. En esta investigación se ha expuesto que las dos son trabajos muy válidos, pero su aplicación es compleja para el ámbito que se propone en este trabajo. La ontología de visualización VISO está orientada a implementar todo el proceso de visualización desde la ontología, incluso define un lenguaje RVL, que permite implementar como visualizar e

interactuar con la información. La ontología VUMO, especifica una serie de información sobre los datos a visualizar, los relaciona con la ontología de dominio, y representa que se tienen que ver con una escena determinada, u otra, sin muchos más aspectos. Desde el punto de vista del autor de esta tesis doctoral, se aporta concluyendo que la definición de un sistema de visualización complejo mediante la ontología VISO es muy costosa y depende del estado actual de la herramienta de visualización en RVL, que actualmente está detenida y queda mucho por desarrollar. Además, al estar la implementación en RVL vinculada a la ontología de dominio, supondría un coste importante la reutilización de esta implementación sobre otro dominio. También se indica que la ontología VUMO está muy orientada a la visualización de grafos de conocimiento sobre tráfico, y no hay un desarrollo final genérico, que permita adaptar este trabajo al ámbito de patrimonio cultural. Además, cualquier cambio en la representación gráfica implicaría realizar nuevos desarrollos. La principal aportación que se realiza es la de indicar la necesidad de una ontología de visualización que permita representar cómo se van a visualizar las particularidades gráficas de los datos de un modelo, pero sin que exista la necesidad de llegar al nivel de implementación. Este nivel será llevado a cabo por una herramienta de código abierto, que leerá el contenido de esta ontología, relacionada con el dominio de datos, para visualizar el contenido según las especificaciones indicadas en la ontología.

- En la revisión bibliográfica realizada por el autor de esta tesis doctoral no ha encontrado un modelo sobre cómo se han de representar los datos de patrimonio cultural, teniendo en cuenta las particularidades de este campo. Debido a ello se ha propuesto el diseño de un modelo con las necesidades de representación gráfica de este tipo de datos. El modelo de datos permite especificar una visualización basada en escenas, asociadas a un entorno temporal y espacial. En la visualización, se mostrarán conceptos y sus propiedades, que estarán relacionados con conceptos y propiedades del grafo de conocimiento del dominio de datos, especificando el entorno espacial y temporal de este dominio. En este modelo, se han tenido en cuenta aspectos inherentes a este campo, como son la incerteza de los datos. También se permite definir la visualización de comportamientos, especificando cómo se agrupan los datos, cómo se relacionan entre ellos, así como la fluctuación de los datos en el tiempo, tanto de forma lineal, como simultánea.
- Se ha diseñado e implementado ontología denominada STEVO, como una propuesta para representar el modelo anteriormente descrito. Esta ontología está implementada en OWL, y está disponible de forma pública a través del portal de repositorios Github. La especificación reutiliza conceptos gráficos definidos en la ontología VISO, así como otras ontologías utilizadas para representar el entorno espacial y temporal, como la ontología TIME de W3C, así como la ontología IFG del Institut National de l'information Géographique (. En la versión actual, esta ontología permite representar datos en dos tipos de escenas. Una de ellas sobre un mapa dinámico, y otra sobre los ejes cartesianos, tanto en dos, como en tres dimensiones.
- Se ha diseñado un marco de trabajo cuyo objetivo es procesar el contenido de la ontología STEVO desde una aplicación web, así como al grafo de conocimiento del dominio de datos. Este componente, una vez se inicia y accede a la información debe de ser capaz de representar gráficamente un conjunto de datos del dominio y permitir a un usuario interactuar y navegar por dicha representación, de acorde a los contenidos de la ontología STEVO. En el diseño se han descrito los módulos principales, como ha de integrarse en una aplicación web, así como la comunicación interna entre esos módulos y también con el

contenido de STEVO, el grafo de conocimiento y el resto componentes web en los que está integrada la aplicación.

- Tras analizar las librerías que permiten representación gráfica en dos y tres dimensiones sobre una aplicación web, se ha desarrollado una propuesta del marco de implementación diseñado en este trabajo. La plataforma elegida ha sido WebGL, por la potencia, rendimiento y su integración en la mayoría de navegadores y sistemas operativos actuales, lo que convierte al desarrollo en una solución multiplataforma. Los desarrollos podrían haberse realizado perfectamente, tanto en Unity 3D, como en Three.JS, pero el autor se decantó por Unity 3D, por motivos de experiencia y de mantenimiento, así como las herramientas y extensiones existentes en el mercado. Además, Unity3D es una herramienta generalista que permite la representación de entornos en dos y tres dimensiones de alta complejidad y escalabilidad, esto permitirá la inclusión de nuevos y más diversas escenas en un futuro con mayor facilidad. Esta propuesta ha sido desplegada sobre dos proyectos reales, con diferentes tipos de escenas de visualización. En uno de los dos proyectos se han realizado pruebas de aceptación y usabilidad, así como de carga, obteniendo resultados satisfactorios.

### 11.3. DISCUSIÓN

A continuación, se exponen dos temas de discusión. En el primer tema se analizan algunos aspectos de la evolución de la digitalización de los datos de patrimonio cultural, su representación mediante grafos de conocimiento, su visualización en la web, así como las ventajas, inconvenientes y trabajo que queda por desarrollar en esta área.

El segundo tema discute sobre las herramientas existentes para visualizar estos datos, el nivel de generalización que tienen, sus necesidades y el papel de la ontología STEVO y la aplicación de los resultados de esta tesis doctoral dentro de este ámbito.

#### 11.3.1. *Evolución de la digitalización de los datos de patrimonio cultural*

Tal y como se ha descrito en el estado del arte de esta investigación, y en parte se expone como parte de las aportaciones de la misma, en la última década ha proliferado de forma importante la digitalización de datos de patrimonio cultural, así como su mantenimiento en grafos de conocimiento.

Aunque muchos de estos trabajos están basados en el modelo CIDOC-CRM, este no es el único usado para representar datos de patrimonio cultural, si esto no fuera suficiente problema, existe el inconveniente adicional del uso de varios tesauros diferentes para referenciar datos de la misma tipología.

Este auge en la digitalización de datos de patrimonio cultural, ha provocado una proliferación del desarrollo de proyectos web, que publican y permiten a sus usuarios buscar e interactuar con estos datos. Dada la gran cantidad de información que se maneja en estos proyectos, se hace más evidente la necesidad de usar estrategias de visualización, mediante las cuales se puedan cumplir los objetivos a alcanzar por los usuarios de los diferentes ámbitos de estos proyectos: Investigación, Docencia, Turismo, etc., así como el objetivo general de las administraciones de difundir el patrimonio cultural.

### Investigación

Es muy importante poder buscar datos especificando valores de las diferentes propiedades de los objetos. Esta funcionalidad permite localizar de forma rápida aquellos objetos que satisfacen unos criterios determinados. Esto es posible gracias al actual proceso de digitalización en el ámbito del patrimonio cultural, sin embargo, para muchos estudios se requieren otras herramientas que no están disponibles de forma sencilla. Esto provoca que el investigador tenga que hacer un trabajo costoso, que en ocasiones es inabordable, dado el gran volumen de información que se maneja. Un ejemplo de tareas de este tipo sería la realización de análisis de datos, con el fin de descubrir patrones en la información, que permita inferir posibilidades. Hecho, que además se complica con la incerteza inherente a la información de patrimonio cultural.

Por ejemplo, un investigador que quiera saber si tejidos de seda que utilicen oro, producidos en Italia durante el siglo XV, tienen alguna relación con los tejidos que usan plata en el mismo siglo, pero procedentes de Turquía, tiene llevar a cabo los siguientes pasos:

- Analizar la información de las propiedades de los diferentes tejidos que cumplen las especificaciones del análisis: Lugar de procedencia, momento de producción, considerar tejidos que usen plata y oro entre sus materiales.
- Realizar un refinamiento de los datos, de acorde a la probabilidad de que realmente proceden de Italia, Turquía, e incluso que se hayan producido en el siglo XV. Todo esto teniendo en cuenta que la herramienta de publicación permita hacer un tratamiento adecuado de la granularidad espacial y temporal. Es decir, que permita saber que un tejido producido en Roma también ha sido producido en Italia, y lo mismo con el tiempo.
- Emparejar estos datos de acorde a las propiedades que tengan en común: materiales, técnicas de tejido, productor, etc.

Como se ha expuesto en el estado del arte, la mayoría de los proyectos web que permiten manejar esta información, permiten realizar filtrados, pero tienen muy poco, o nada desarrollada la posibilidad de obtener relaciones entre los datos, o comparar resultados entre diferentes búsquedas. Esto implica realizar varios filtrados por separado y si la herramienta tiene posibilidades de exportación, tratarlos con herramientas externas, como hojas de cálculo, paquetes estadísticos, etc.

Tener los datos digitalizados y accesibles en una aplicación web es un paso fundamental para los investigadores del patrimonio cultural, pero con el volumen de datos existente y la incerteza de los mismos, realizar investigaciones de alto nivel en este campo puede ser una labor compleja que conlleva un alto coste. Este proceso sería mucho más sencillo si se redujera al máximo el uso de herramientas externas y la propia aplicación web tratara de diferentes formas el problema de la incerteza y permitiera comparar los resultados de diferentes filtrados y visualizar las relaciones entre los datos del resultado de forma sencilla.

### Turismo

En la actualidad, el turista interesado en el patrimonio cultural tiene diferentes fuentes para encontrar los lugares de interés cultural de una zona, su ubicación, lo que puede encontrar allí, o cómo acceder y el coste que requiere. Sin embargo, junto a esta información, que desde luego es importante, pueden existir otros factores que hagan decidir al turista si finalmente realiza su viaje. Varios estudios (Menor-Campos, Pérez-Gálvez et al. 2020, Fernández, Pérez-Gálvez et al. 2017) indican que los que más atrae a este tipo de turismo son las relaciones de este patrimonio

con la cultura (festividades, indumentaria, costumbres, leyendas, hechos históricos derivados o relacionados), incluso aspectos tan humanos, como la gastronomía de un lugar, pueden ser fundamentales para que termine eligiendo un destino u otro. Estas personas, suelen tener un alto nivel educativo y suelen usar varias fuentes para terminar de decidir su destino. Los proyectos de visualización de datos de patrimonio cultural pueden jugar un papel importante en esta decisión, si explotan la posibilidad de enlazar con estos datos que el turista quiere conocer. Para que esto sea una realidad, no sólo debe explotarse al máximo la relación entre los datos de una aplicación, sino las existentes con otros dominios de datos.

Por ejemplo, un turista puede estar dudando entre visitar Valencia en octubre, o ir a otra ciudad española. Esta persona puede decidir ir a Valencia, si descubre que en esa fecha se celebra la festividad de Sant Dionís, en la que se regalan dulces, que según la tradición tiene forma de artefactos pirotécnicos, elaborados como protesta ante las leyes abusivas del rey Felipe V, que conquistó la región, y que además este rey tiene un cuadro colgado al revés en la ciudad de Xàtiva, porque incendió esta ciudad, que tiene uno de los castillos mejor conservados del país. En la actualidad hay varias herramientas web para localizar el Castillo de Xàtiva de forma sencilla, pero si estás aplicaciones, en función de la fecha actual, o si la resaltan debidamente, podrían informar que existen festividades en ese mes, relacionadas en cierta forma con la historia del Castillo, podrían ser un factor determinante para el turista.

Estos datos sólo pueden conseguirse compartiendo, enlazando y visualizando datos de diferentes fuentes de información.

### *Difusión del patrimonio cultural de forma correcta*

Con el desarrollo de proyectos de búsqueda y visualización de datos de patrimonio cultural a través de aplicaciones web, parece que se está alcanzando este objetivo de difusión. Cada vez es más fácil encontrar información sobre todo tipo de objetos de patrimonio cultural a través de la web, tanto en portales diseñados directamente con este campo, como de forma indirecta. Estos avances son realmente necesarios para la difusión, y aunque existen muchos objetos por digitalizar y/o enlazar, su incremento es innegable. Sin embargo, una vez más, la posibilidad de relacionar los datos y poder buscar por diferentes criterios, incluido el de buscar por imágenes similares es muy importante y aún queda mucho trabajo por hacer en esta área. Este tipo de búsquedas, y poder enlazar con los datos relacionados, tanto a nivel de objetos, como con otros elementos culturales e históricos afecta a otros temas sociales de gran importancia, además del patrimonio cultural y su preservación.

Por ejemplo, dado el aumento de digitalización de estos datos, es relativamente sencillo acceder a imágenes digitales y también a otras fuentes de datos. Este tipo de información, fuera del contexto y contando con el efecto multiplicador de las redes sociales, promueven en la sociedad actos tan deleznable como el racismo y la xenofobia (Guiliano, Heitman 2019).

Por estos motivos, es muy importante que un usuario pueda encontrar fácilmente datos relacionados con esa fotografía, o ese hecho, para poder encontrar el contexto que no ha sido incluido y poder tener datos para procesar de forma correcta la información y rebatirla si fuese necesario.

### *11.3.2. Herramientas para la visualización de datos de patrimonio cultural*

Como se ha expuesto varias veces en este trabajo, hay una cantidad ingente de datos de patrimonio cultural soportados por grafos de conocimiento, con varias relaciones entre ellos, y existe una necesidad importante para visualizarlos de forma gráfica.

Estos datos están representados por varios modelos y usan diferentes tesauros. Aunque este problema va reduciéndose y cada vez la mayoría de los proyectos se representan mediante un conjunto reducido de modelos, no es un tema que vaya a resolverse a corto plazo.

Las herramientas de visualización generalmente son realizadas para los fines del proyecto, y en muchas ocasiones los desarrollos no son abiertos.

Esto implica la necesidad de realizar herramientas de visualización que sean independientes de los modelos y del dominio de datos, además de que sean abiertas.

Un soporte ideal, dado que el dominio de datos está soportado por un grafo de conocimiento, que dicha visualización se representará mediante una ontología. En la revisión realizada en el estado del arte las ontologías encontradas que persiguen este objetivo son VUMO y VISO. Sin embargo, aunque las dos ideas son muy buenas y tienen asociados unos trabajos excelentes, presentan algunos problemas importantes para el objetivo planteado en esta investigación.

La ontología VISO, junto al lenguaje RVL, busca no sólo definir cómo se han de visualizar los datos, sino que también desciende a un nivel muy cercano a la implementación con el lenguaje RVL. Con la ontología VISO define un conjunto de representaciones gráficas, que son utilizadas por RVL que está enlazado con extensiones de las ontologías utilizadas en el grafo de conocimiento de dominio. Desde el punto de vista del autor tesinando puede ser muy interesante para visualizaciones complejas de un dominio de datos. Sin embargo, presenta problemas cuando se trabaja con grandes conjuntos de datos e interfaces complejos. La problemática derivada de la comunicación con aplicaciones externas, uso de servicios web, y una correcta gestión de una reserva de datos (cache), para optimizar búsquedas, combinada con la complejidad de una interfaz de usuario que proporcione una buena experiencia de uso. El lenguaje RVL puede ser adecuado, para representar un conjunto limitado de instancias y conceptos de un grafo de conocimiento, en función de las reglas y el valor de sus propiedades, pero actividades como gestionar agrupaciones de miles de datos, hacer uso de texturas de diferentes resoluciones, así como representar escenas complejas, como un hipercubo con capas con diferentes texturas, suponen un reto difícil de alcanzar. Además, existe el problema de la reutilización de código. RVL está muy vinculado al dominio de datos, cambiar este dominio supondría actualizar el código RVL, lo cual es un coste adicional y nada despreciable. La implementación de un visor RVL es una herramienta multiplataforma integrable en una aplicación web, pero está en etapas iniciales de desarrollo, queda mucho trabajo por desarrollar y además el proyecto lleva algunos años parado. Esto no es un problema grave, ya que el código es abierto, pero los desarrollos a realizar requerirían un coste importante para poder generar una versión que visualice datos de proyectos de patrimonio cultural, con una interfaz y funcionalidades amigables para un usuario medio.

La ontología VUMO, según el criterio de este investigador, no desciende al nivel de implementación, como hace VISO. Esta ontología tiene una herramienta que permite visualizar el contenido de las instancias especificadas en VUMO, que están enlazadas con el dominio de datos. Sin embargo, la principal función de esta ontología consiste en adaptar los datos del dominio a los conceptos de las escenas, muy orientadas a tráfico urbano, y usar las escenas de la herramienta. La ontología permite definir muy pocas propiedades de la visualización. No da opción a definir comportamientos como las agrupaciones, incerteza, relaciones. Tampoco relaciona los sistemas espaciales ni temporales del dominio de datos con los de la visualización. Y tiene limitaciones importantes a la hora de definir cómo se representan los datos.



Resumiendo, los trabajos realizados en esta área están muy polarizados, por una parte, VISO y RVL, es un resultado muy potente que permite especificar con mucho detalle como representar la información, pero además de que queda mucho por desarrollar a nivel de visualización, tiene limitaciones importantes para visualizar datos de patrimonio cultural. Por otra parte, VUMO no tendría problema para representar estos datos, siempre que se puedan vincular a las escenas de visualización y sus conceptos, centrados en proyectos de tráfico urbano, que además deja pocas posibilidades en la representación y el control del tiempo y la incerteza. Para el objetivo de esta visualización es necesario una solución intermedia, que no requiera la complejidad de RVL y VISO, pero que tenga una potencia elevada para definir cómo se visualizan los datos, y gestione la incerteza, el control del tiempo y cómo visualizar las relaciones entre los datos.

STEVO y los resultados propuestos en este trabajo de investigación, serían una posible solución intermedia. Ya que permite especificar los detalles de un modelo de representación complejo sobre este tipo de datos. Esta especificación comprende la problemática de la incerteza, las relaciones entre los objetos, el control del tiempo, así como la relación entre esta especificación y los conceptos del dominio de datos. Además, permite la definición de muchos aspectos de la interfaz de usuario. STEVO y los resultados de esta investigación, usan toda la potencia de especificación de VISO para definir como se visualizan los datos gráficamente, pero evita la complejidad de RVL, delegando esta tarea en la herramienta de visualización, que es una aplicación web multiplataforma, de código abierto.

Por lo tanto, con STEVO tendríamos una ontología con la que podríamos especificar qué datos visualizar, cómo visualizarlos con un alto nivel de detalle gráfico, y definir varias posibilidades de interacción y de visualización. Esto se consigue mediante:

- La definición de unos conceptos, que serán los elementos que se van a visualizar, y sus propiedades, y establecer un enlace con los conceptos, propiedades y relaciones del dominio de datos. Además, se puede definir qué propiedades de esos conceptos que se van a visualizar van a poder ser filtradas desde la herramienta y van a permitir relacionar objetos, y fijar las propiedades que definen espacio y tiempo, muy importantes en los datos de patrimonio cultural.
- Especificando el entorno temporal y espacial, tanto del entorno de visualización, como del dominio de datos.
- Definir varias posibilidades de visualización en torno al concepto de Escena, indicando en cada escena, sus posibilidades de interacción, y de navegación básicas, el tipo de escena, y comportamientos que establecen aspectos clave de visualización e interacción, como las agrupaciones, visualización de relaciones y control del tiempo.

### 11.4. CONCLUSIONES

Con el trabajo aportado mediante esta tesis doctoral queda claro que, aunque aún resta mucho trabajo por hacer, se está realizando un esfuerzo importante para digitalizar datos de patrimonio cultural y soportar estos datos mediante grafos de conocimiento, principalmente y cada vez más sobre el modelo de CIDOC-CRM.

Estos avances son motivados por las administraciones, con ánimo de preservar el patrimonio cultural, pero también porque organismos educativos, de turismo y empresas están comenzando a considerar los beneficios de explotar los resultados de difundir el patrimonio cultural.

Los últimos desarrollos realizados en este campo están beneficiando el impulso de tecnologías que permiten manipular y acceder a este tipo de estructuras, además de generar más publicaciones en la comunidad científica.

Una de las áreas que más trabajo queda por hacer, no sólo en patrimonio cultural, sino en todo el campo de grafos de conocimiento es el la de visualizar la información del grafo para ser integrada en aplicaciones y utilizada por sus usuarios con interfaces similares a otras aplicaciones de visualización.

Aunque hay proyectos que consiguen estos resultados, son proyectos ad hoc, dónde la visualización suele estar centrada en el dominio de datos y las necesidades del proyecto, muchos de los cuales no son de código abierto. De esta forma, cada visualización de datos, se diseña y desarrolla prácticamente de nuevo.

Una forma de generalizar este problema es definir un modelo de visualización que estuviese representado por una ontología, que pudiera procesarse con una herramienta integrable en una aplicación web y que generase la visualización y la interacción en base a lo definido en la ontología.

En esta tesis doctoral se ha definido un modelo de visualización de datos que estando diseñado y orientado a patrimonio cultural puede ser aplicado a otro tipo de ámbitos. Esto es posible porque las particularidades del patrimonio cultural consisten principalmente en la necesidad de asociar los datos con información espacio temporal, que además es probable que tengan incerteza. Estas características están presentes en muchos dominios de datos. De todas formas, el modelo contempla estas posibilidades, pero no supondría ningún problema si el dominio de datos no las tuviera.

De cualquier forma, en este trabajo, además del modelo de visualización, se ha diseñado e implementado la ontología STEVO, que representa el modelo citado.

También se ha diseñado un marco de trabajo, del cual se ha implementado una propuesta que permite representar dos tipos de escenas partiendo y contemplando varias opciones de visualización e interacción.

La ontología desarrollada está basada en otros resultados, como la ontología VUMO y la ontología VISO, que de hecho reusa, junto con otras como TIME y la de representación de datos geográficos del Instituto Geográfico Francés.

Los resultados de esta investigación han sido integrados en dos proyectos de patrimonio cultural, SILKNOW y Arxiu Valencià del Disseny, cuyos datos están soportados en grafos de conocimiento. Uno de ellos aún está en desarrollo, pero el proyecto SILKNOW terminó en 2021 y dentro del cual se ha realizado un test de aceptación y usabilidad de la propuesta desarrollada en esta tesis, con resultados satisfactorios.

También se han realizado pruebas de carga para ver las limitaciones de la herramienta en cuanto a requisitos hardware y tamaños de los conjuntos de datos a visualizar. Estas pruebas sirvieron para confirmar la necesidad de desarrollar mejoras para que la herramienta fuera capaz de soportar conjuntos de datos muy grandes. A pesar de esto, el sistema podría soportar las necesidades de muchos proyectos sin que sea necesario usar equipos con grandes prestaciones.

Estos resultados, junto al hecho de que la visualización de escenarios de elevada calidad con gráficos en dos y tres dimensiones es una realidad en la web, y existen librerías como WebGL que son de código abierto y multiplataforma, hacer ver una ampliación de este trabajo gracias a su utilización en otros proyectos con grafos de conocimiento.

## CAPÍTULO 12. TRABAJO FUTURO

### 12.1. TRABAJO FUTURO

A continuación, se describen mejoras y ampliaciones de este trabajo que crear nuevas líneas que podrían ser desarrolladas en futuros trabajos.

#### *12.1.1 Visualización de grandes conjuntos de datos*

Uno de los principales problemas que presenta la implementación realizada es la limitación del tamaño del conjunto de datos. En la versión actual, se delega demasiada responsabilidad en el componente de visualización WebGL y no se realiza gestión de paginación de datos, o petición de datos en función de la zona y el nivel de detalle actual. Esta situación genera los problemas de memoria detectados en las pruebas de carga. Este funcionamiento es tolerable y funcional con conjuntos de datos no muy grandes, pero con grandes volúmenes de datos el sistema puede no ser operativo. En las pruebas de carga, con la tecnología actual, presente en la mayoría de equipos puede soportar visualizar conjuntos de hasta 15.000 objetos en la escena de mapa espacio temporal sin utilizar grandes prestaciones hardware ni a nivel gráfico ni de otro tipo. Con un equipo de altas prestaciones, podría manejar perfectamente la carga de 30.000 objetos, pero el tiempo requerido para la visualización inicial estaría cercano al tiempo medio que un usuario no considerara una buena experiencia de uso. La tecnología mejorará, y podrán visualizarse conjuntos de datos más grandes, pero siempre existirá un número en el que el sistema presentará problemas. Considerando que los datos de patrimonio cultural son de un tamaño cada vez mayor y que existen proyectos nacionales, como Semap (U.Valencia 2021), que maneja cerca de 400.000 objetos, es uno de los problemas más prioritarios a resolver.

Existen varias vías para solucionar, o reducir considerablemente, este problema. Una de las más factibles es que el módulo gráfico en WebGL haga peticiones al componente de visualización, desarrollado en JavaScript, para solicitar información de los datos que va a visualizar. Esto requiere que el componente de visualización se encargue de gestionar una paginación de datos, además de pre procesamiento de datos para obtener los datos de las agrupaciones, y relaciones en relación a los diferentes niveles de detalle de la escena. Estos trabajos de calcular relaciones, agrupaciones y demás información de forma previa, son necesarios para una rápida respuesta, pero requieren que se actualicen con cierta frecuencia si el dominio de datos se actualiza. La actualización del dominio de datos no es un hecho preocupante en el ámbito de patrimonio cultural, ya que la información se puede actualizar, pero no de forma periódica. Sin embargo, en otro tipo de dominios de datos la actualización es muy frecuente, y esto puede suponer un problema importante.

#### *12.1.2. Ampliación de nuevas variables de tiempo y espacio sobre la escena*

La ontología STEVO permite definir sobre cada escena una propiedad visual temporal, y otra espacial, de referencia en la escena. Esto puede ser suficiente para muchos proyectos, pero en otros es una restricción importante. Por ejemplo, dentro del ámbito cultural, un objeto puede haberse producido en un lugar concreto del espacio y en un momento concreto, lo mismo sucede con su ubicación actual, y las diferentes acciones, o eventos relacionados con dicho objeto (cambio de propiedad, uso en eventos históricos, etc.) Esta limitación es importante, ya que el usuario del sistema puede desear ver dónde y cuándo se han producido los datos, pero también podría estar en saber dónde están ahora, y desde cuando están en dicha ubicación. Por este motivo, es necesario que sobre una escena de STEVO puedan definirse varias propiedades visuales, relacionadas con tiempo y con espacio, asociadas a los diferentes comportamientos. Además, la implementación y el diseño del componente de visualización deben de contemplar esta información, para mostrar al usuario una interfaz adecuada con la que poder definir cuál activar.

### 12.1.3. Definición de nuevas escenas y transiciones entre las mismas

En la actualidad, la ontología STEVO permite definir dos tipos principales de escena: Una escena que muestra los datos en un mapa interactivo y dinámico, y otra en ejes cartesianos en dos y tres dimensiones. La ontología permite definir muchos aspectos de las posibilidades de interacción, aspectos gráficos de los objetos y sus propiedades, y todo un abanico de comportamientos. Sin embargo, por una parte, puede ser interesante realizar otro tipo de escenas para representar la información, y por otra que exista la posibilidad de que en una visualización coexistan varias escenas de visualización y sea posible definir la transición entre ellas. Un ejemplo de esta necesidad se presenta en el proyecto Semap, cuando hay grandes agrupaciones de objetos en un mismo lugar (museo, colecciones privadas, etc.) y se desea visualizar el contenido. En esta situación, al seleccionar la agrupación se muestra el contenido a través de estructuras tridimensionales dispuestas de forma helicoidal, o esférica, permitiendo al usuario navegar por esta escena y visualizar una imagen asociada a cada objeto. En la Figura 100 se muestra un captura de pantalla del proyecto Semap. En esta se observa la visualización, en disposición helicoidal, de una agrupación con 263 objetos. En este caso se representa un mapa dinámico de fondo y de forma superpuesta una visualización tridimensional con los datos de dicha agrupación.

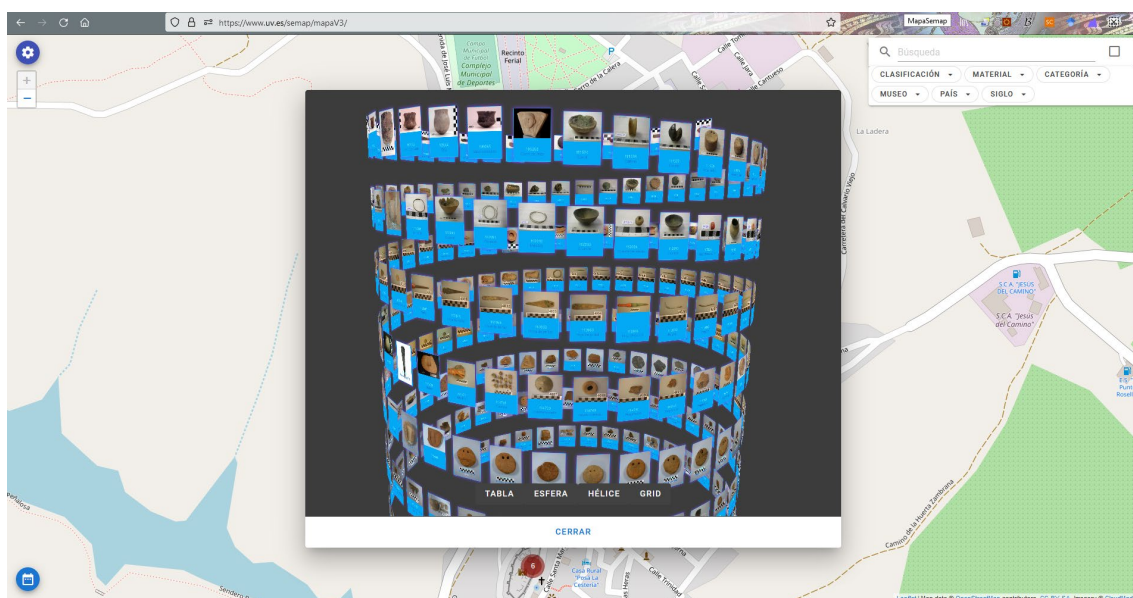


Figura 100 Captura de pantalla de los resultados del proyecto SEMAP. Vista helicoidal del contenido de una agrupación con 263 objetos de un yacimiento de Jaén. Fuente: Semap (U.Valencia 2021)

Otro tipo de escenas, aplicables a educación, o con otro tipo de fines, podría ser la generación automática de galerías virtuales, o libros donde el usuario puede pasar las páginas con realismo. En la Figura 101 se muestran dos capturas de pantalla de visualizaciones de este tipo realizadas con Javascript. A la izquierda un libro, donde el contenido de cada página se genera mediante HTML y Turn.JS (Garcia 2013), una librería Javascript, el usuario puede cambiar de página con realismo. A la derecha, una galería virtual generada automáticamente partiendo de imágenes que son agrupadas (Hindle 2017). El sistema genera de forma automatizada las galerías y los niveles de detalle de forma adecuada. En este último caso la librería usada sería Three.JS. Las dos opciones son de código abierto y permiten su integración en cualquier aplicación web sin ningún tipo de problema. Este tipo de soluciones, podrían ser integradas de forma sencilla con el sistema propuesto, pero su especificación debería de poder definirse en STEVO. Además, también es importante definir la transición, que aparezca de golpe, o no, sustituyendo a la anterior escena, o visualizarse de forma simultánea serían algunas de las opciones que habría que definir.



Figura 101 Ejemplo de posibles escenas integrables con la herramienta propuesta. A la izquierda un libro desarrollado con Turn.JS (García 2013), y a la derecha una galería 3D (Hindle 2017) generada a partir de imágenes, con Three.JS.

#### 12.1.4. Nuevos elementos de visualización

Las posibilidades gráficas que pueden definirse con STEVO distan mucho de estar terminadas. La visualización de datos es un campo en el que hay un gran trabajo realizado y muchos de estos resultados deberían de poderse especificar con STEVO y poder ser implementados. Ejemplos de estas extensiones sería la posibilidad de definir las agrupaciones de datos con tamaños, o colores, que fueran función del número de objetos que contienen. El resultado de esta característica puede verse en el proyecto Semap, donde las agrupaciones representan el tamaño de su contenido mediante colores, pero también puede hacerse modificando el tamaño de las mismas. Las relaciones en corona circular implementadas con STEVO están limitadas a un número limitado de propiedades, a partir de 4 ya cuesta identificar y comprender el porcentaje de relaciones, sería interesante modificar durante la visualización que propiedades aparecen en el anillo, o incluso hacer modificaciones del control. Elementos más avanzados de la representación gráfica, como fuerzas, dinámica, sobre los objetos, e incluso el uso de avatares virtuales, serían elementos que también podrían incluirse, pero también requieren un trabajo importante en el diseño de la ontología y del modelo.

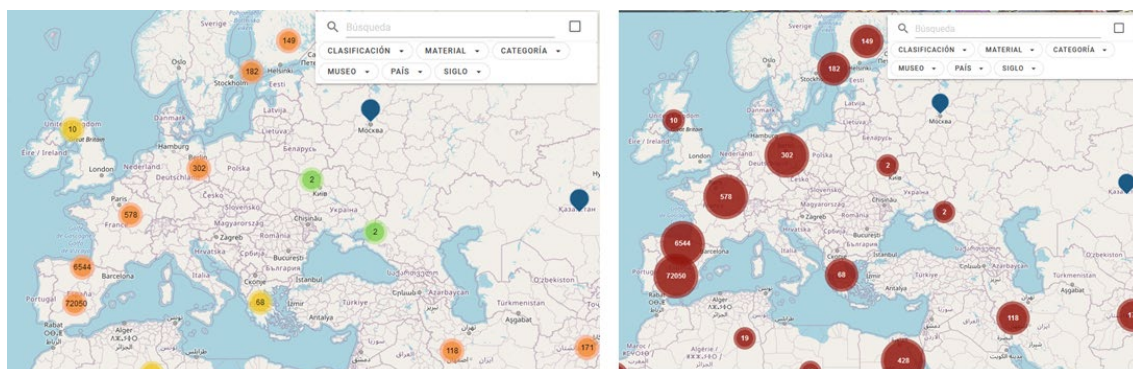


Figura 102 Representación del tamaño de las agrupaciones mediante colores (izquierda), o tamaño de los símbolos (derecha). Fuente: Semap (U.Valencia 2021).

#### 12.1.5. Funcionalidades de exportación

Al estar los datos soportados por un grafo de conocimiento y en una plataforma con un API, que permite ejecutar consultas sobre el grafo, es fácil compartir los datos con otras aplicaciones. Sin embargo, no es tan sencillo compartir los datos de visualización. En la actualidad muchas herramientas de visualización recogen información de imágenes, y datos en formato GeoJSON (Butler, H., Daly, M., Doyle, A., Gillies, S., Hagen, S., and T. Schaub 2016), para representar la visualización. Por este motivo, sería interesante que la herramienta de visualización, permitiera



la exportación de la información en diferentes formatos y que esta posibilidad estuviera contemplada en STEVO.

#### *12.1.6. Difusión*

Los resultados de STEVO, el diseño propuesto, la propuesta de implementación y este trabajo de investigación estarán dispuestos de forma abierta. La investigación y sus resultados han sido de gran utilidad para el autor de este trabajo. Uno de sus objetivos es que este trabajo sea útil para la comunidad científica. Si este objetivo se consigue y estos resultados son usados en otros proyectos, probablemente la comunidad científica participará en ampliar la herramienta desarrollada, incluso aportar nuevas propuestas de desarrollo en otras librerías como ThreeJS. Para que esto sea una realidad se tiene que hacer una importante labor de difusión en redes sociales, grupos, publicación y participación en congresos.

#### *12.1.7. Desarrollo de sistema de recomendación de visualización de datos*

Hasta ahora, todas las líneas abiertas de trabajo futuro estaban dirigidas a mejorar el rendimiento y las funcionalidades de los resultados obtenidos en esta tesis doctoral. Sin embargo, otra línea de investigación muy interesante sería explotar los resultados para poder evaluar la mejor forma de visualizar la información, en función del dominio de datos y de las características del usuario. Uno de los motivos de definir la visualización de datos mediante un modelo y representarlo mediante una ontología, era el de poder definir cómo se ha de visualizar la información de forma computacional. Al definir la visualización mediante una ontología, es muy sencillo modificar la visualización, o acceder a las diferentes formas de visualización posibles. Si se puede disponer de información sobre las preferencias de un usuario en concreto, o de usuarios con un perfil similar al que va a visualizar la información, se pueden escoger las mejores opciones en función de estas preferencias. Esta información sobre las necesidades, o posibilidades de interacción que más le atraen a un usuario pueden obtenerse bien por que él lo especifica, o mediante la realización de experimentos. Con esta forma de proceder, se podrían definir en un sistema perfiles de usuarios atendiendo a la finalidad de la visualización (investigación, educativo, turismo, etc.) y otros factores como nivel cultural, edad, etc., y representar la información de forma que sea más útil para esta persona de forma automatizada.

## GLOSARIO DE TÉRMINOS

### A

#### AAT

Art & Architecture Thesaurus, 188

#### API

Application Program Interface, 47, 52, 56, 57, 59, 61, 63, 64, 66, 67, 106, 162, 163, 170, 171, 172, 176, 179

Application Programmin Interface, 47

#### ARIADNE

Artificial Intelligence Aided D-band Network, 5

#### ARTEC

Advanced Research and Technological Expansion in Computer Graphics, 3, 13

#### AVD

Arxiu Valencià de Disseny, 4, 212, 213

### Ch

#### CHARM

Cultural Heritage Abstract Reference Model, 32

### C

#### CIDOC-CRM

Cultural Heritage Abstract Reference Model, 2, 5, 6, 8, 32, 33, 34, 54, 61, 95, 137, 187, 193, 195, 197, 199, 200, 211, 235, 237, 241, 274

### D

#### DAML

DARPA Agent Markup Language, 19

#### DAV

Distributed Authoring and Versioning, 63

#### DTD

Document Type Definition, 18

### E

#### EAMENA

Endangered Archaeology in the Middle East & North Africa, 53

#### EDM

EUROPEANA Data Model, 32

#### eRQL

easy RDF Query Language, 22

#### ETRS

European Terrestrial Reference System, 116

### F

#### FISH

Forum on Information Standards in Heritage, 30

#### FPS

Frames Per Second, 231

### G

GeoJSON  
Geographical Javascript Object Notation, 69

GPU  
Unidad de Procesamiento Gráfico, 65

### H

HERACLES  
HEritage Resilience Against CLimate Events on Site, 5

HTML  
HyperText Markup Language, 65

### I

IFG  
Institut National de l'information Géographique, 236

IGF  
Institut National de l'information Géographique, 117, 118  
Institut National de l'Information Géographique, 117

ISO  
International Organization for Standardization, 30

### J

JDBC  
Java Database Connectivity, 63

JS  
JavaScript, 160, 161

JSON  
JavaScript Object Notation, 165, 171, 174, 175

### M

METHONTOLOGY  
Methodology for Ontology construction, 104

### N

NELL  
Never-Ending Language Learner, 17

NEMO  
Network of European Museum Organisations, 28

NNDB  
Notable Names Database, 16

### O

ODBC  
Open DataBase Connectivity, 63

OIL  
Ontology Interchange Language, 19

ONG  
Organización No Gubernamental, 58

OpenGL  
Open Graphic Language, 66

OWL  
Ontology Web Language, 20, 21, 49, 63, 64, 151, 236



### R

- RAF
  - Royal Air Force, 55
- RAM
  - Random Access Memory, 231
- RDF**
  - Resource Description Framework, 19, 21, 22, 62, 63, 64, 160
- RDF4J
  - RDF For Java, 63
- RDFS
  - Resource Description Framework Schema, 19
- RDQL*
  - RDF Data Query Language, 23
- RQL**
  - RDF Query Language, 22
- RVL
  - RDFS/OWL Visualization Language, 48, 107, 109
- RVL
  - RDFS/OWL Visualization Language, 110

### S

- SeRQL**
  - Sesame RDF Query Language, 22
- SHACL
  - Shapes Constraint Language, 21
- SHOE**
  - Simple HTML Ontology Extension, 18
- SILKNOW
  - Silk heritage in the Knowledge Society, 3
- SKOS
  - Simple Knowledge Organization System, 30
- SOAP
  - Simple Object Access Protocol, 63
- SPARQL
  - SPARQL Protocol and RDF Query Language, 22, 96
- SQL
  - Structured Query Language, 64
- STEVO
  - Spatio Temporal Visualization Ontology, 104
- SUS
  - Sistema de Escala de Usabilidad, 215
- SVG
  - Scalable Vector Graphics, 67

### U

- UNESCO
  - United Nations Educational, Scientific and Cultural Organization, 27
- URI
  - Uniform Resource Identifier, 19, 96
- UTM
  - Universal Transverse Mercator, 116

### V

- VISO
  - Visualization Ontology, 13

### VRML

Virtual Reality Modeling Language, 66

### VUMO

Visualization Urban Mobility Ontology, 47

## W

### WebGL

Web Graphic Language, 65

### WGS

World Geodetic System, 116

### WS

Web Service, 63

## X

### X3D

Extensible 3D, 47

### XML

Extensible Markup Language, 18

### XOL

XML-Based Ontology Exchange Language, 18

### XSD

XML Schema Definition Language, 145

XML Schema Document, 18

## Y

### YAGO

Yet Another Great Ontology, 17

## REFERENCIAS

- ABADIE, N. and ATEMEZING, A., May 05, 2015-last update, Ontology of coordinates reference systems. Available: <http://data.ign.fr/def/ignf/20150505.en.htm#classes> [Nov 15, 2021].
- ACKE, L., DE VIS, K., VERWULGEN, S. and VERLINDEN, J., 2021. Survey and literature study to provide insights on the application of 3D technologies in objects conservation and restoration. *Journal of Cultural Heritage*, **49**, pp. 272-288.
- AGA KHAN DOCUMENTATION CENTER AT, MIT, 2019-last update, LayerCake: New data visualization tool. Available: <https://layercake.mit.edu/> [08/18/, 2021].
- AGAFONKIN, V., 2020-last update, Leaflet. Available: <https://leafletjs.com/> [08/22/, 2021].
- AGAFONKIN, V., 2022-last update, Leaflet. Available: <https://leafletjs.com/> [Jan 31, 2022].
- AGAFONKIN, V., 03/31/, 2016-last update, Clustering millions of points on a map with Supercluster. Available: <https://blog.mapbox.com/clustering-millions-of-points-on-a-map-with-supercluster-272046ec5c97> [08/16/, 2021].
- AKIHIKO KUSANAGI, 2022-last update, Mini Tokyo 3D. Available: <https://minitokyo3d.com> [Sep 30, 2022].
- ALI, W., SALEEM, M., YAO, B., HOGAN, A. and NGOMO, A.N., 2020. Storage, Indexing, Query Processing, and Benchmarking in Centralized and Distributed RDF Engines: A Survey. *Preprints*, .
- ANDRIENKO, G., ANDRIENKO, N., BAK, P., KEIM, D., WROBEL, S., 2013. *Visual Analytics of Movement*. Springer.
- ANDRIENKO, N., ANDRIENKO, G. and GATALSKY, P., 2000Visualization of spatio-temporal information in the Internet, 2000, pp. 577-585.
- ANDRIENKO, N., ANDRIENKO, G. and GATALSKY, P., 2003. Exploratory spatio-temporal visualization: an analytical review. *Journal of Visual Languages & Computing; Visual Data Mining*, **14**(6), pp. 503-541.
- ANDROID, 2022-last update, Map Items (Android SDK). Available: [https://developer.android.com/documentation/android-sdk-navigate/4.7.2.0/dev\\_guide/topics/map-items.html](https://developer.android.com/documentation/android-sdk-navigate/4.7.2.0/dev_guide/topics/map-items.html) [Sep 30, 2021].
- ANSARI, M.Y., AHMAD, A., KHAN, S.S., BHUSHAN, G. and MAINUDDIN, 2020. Spatiotemporal clustering: a review. *Artificial Intelligence Review*, **53**(4), pp. 2381-2423.
- APACHE FOUNDATION, 2021a-last update, Fuseki. Available: <https://jena.apache.org/documentation/fuseki2/> [Nov 12, 2021].
- APACHE FOUNDATION, 2021b-last update, Jena. Available: <https://jena.apache.org/> [Nov 12, 2021].

ARCADIA FUND, EAMENA PROJECT, BRITISH COUNCIL, 2018-last update, EAMENA Database Search Web Page. Available: <https://database.eamena.org/en> [08/22/, 2021].

ATEMEZING, G.A. and AMARDEILH, F., 2018 Benchmarking Commercial RDF Stores with Publications Office Dataset, A. GANGEMI, A.L. GENTILE, A.G. NUZZOLESE, S. RUDOLPH, M. MALESHKOVA, H. PAULHEIM, J.Z. PAN and M. ALAM, eds. In: *The Semantic Web: ESWC 2018*, 8 Satellite Events 2018, Springer International Publishing, pp. 379-394.

AUMANN, J., 2021-last update, Github. API proyecto Chronas. Available: <https://github.com/Chronasorg/chronas-api> [08/24/, 2021].

AZAVEA, 2021-last update, OpenTreeMap. Available: <https://www.opentreemap.org/> [08/30/, 2021].

BACH, B., DRAGICEVIC, P., ARCHAMBAULT, D., HURTER, C. and CARPENDALE, S., 2017. A Descriptive Framework for Temporal Data Visualizations Based on Generalized Space-Time Cubes. *Computer Graphics Forum*, **36**(6), pp. 36-61.

BACH, B., DRAGICEVIC, P., ARCHAMBAULT, D., HURTER, C. and CARPENDALE, S., Jun 11, 2014 A Review of Temporal Data Visualizations Based on Space-Time Cube Operations, Jun 11, 2014.

BACLAWSKI, K., BENNETT, M., BERG-CROSS, G., CASANAVE, C., FRITZSCHE, D., LUCIANO, J., SCHNEIDER, T., SHARMA, R., SINGER, J., SOWA, J., SRIRAM, R.D., WESTERINEN, A. and WHITTEN, D., 2018. Ontology Summit 2018 Communiqu  : Contexts in context. *Applied Ontology*, **13**, pp. 181-200.

BAYRAKDAR, S., YUCEDAG, I., SIMSEK, M. and DOGRU, I.A., 2020. Semantic analysis on social networks: A survey. *International Journal of Communication Systems*, **33**(11), pp. e4424.

BEKIARI, C., BRUSEKER, G., DOERR, M., ORE, C., STEAD, S., VELIOS, A., 2021. *Definition of the CIDOC Conceptual Reference Model*  
. CIDOC-CRM.

BEKIARI, C., BRUSEKER, G., DOERR, M., ORE, C., STEAD, S., VELIOS, A., *Definition of the CIDOC Conceptual Reference Model. Version 7.1.*  
.

BENNETT, M. and BACLAWSKI, K., 2017. The role of ontologies in Linked Data, Big Data and Semantic Web applications. *Applied Ontology*, **12**, pp. 189-194.

BHUSHAN, S., 2016. *SPARQL-THE QUERY LANGUAGE OF SMANTIC WEB*.

BIGMEDILYTICS, 2021-last update, Bigmedilytics Blue Print. Available: <https://www.bigmedilytics.eu/blueprint/> [Sep 30, 2021].

BLUMAUER, A., 04/12/, 2016-last update, From Taxonomies over Ontologies to Knowlwdge Graphs  
. Available: <https://2021-eu.semantics.cc/taxonomies-over-ontologies-knowledge-graphs> [08/05/, 2021].

BOGUCA, E.P. and JAHNKE, M., 2018. *Feasibility of the Space–Time Cube in Temporal Cultural Landscape Visualization*.

BOLÍVAR, H., GONZÁLEZ, R., SANJUÁN, O., 2009. Ontology of a scene based on Java 3D architecture. *International Journal of Interactive Multimedia and Artificial Intelligence*, 1(2), pp. 14-19.

BOLLACKER, K., EVANS, C., PARITOSH, P., STURGE, T. and TAYLOR, J., 2008. Freebase: a collaboratively created graph database for structuring human knowledge. *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. New York, NY, USA: Association for Computing Machinery, pp. 1247-1250.

BONATIVA, L., BOSTOCK, M., 2018-last update, TopoJSON Format. Available: <https://github.com/topojson/topojson-specification> [08/28/, 2021].

BONEVA, I., LABRA-GAYO, J.E., HYM, S., PRUD'HOMMEAU, E.G., SOLBRIG, H.R., STAWORKO, S., 2014. Validating RDF with shape expressions. *CoRR*, .

BOOKING, 2021-last update, Booking Web Page. . Available: <https://www.booking.com> [08/28/, 2021].

BOSTOCK, M., OGIEVETSKY, V., HEER, J., 2011. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, , pp. 2309.

BOSTOCK, M., OGIEVETSKY, V., HEER, J., 2011. D3: Data-Driven Documents. *IEEE Trans. Visualization & Comp. Graphics*, .

BOULOS, M.N.K., WARREN, J., GONG, J. and YUE, P., 2010. Web GIS in practice VIII: HTML5 and the canvas element for interactive online mapping. *International Journal of Health Geographics*, 9(1), pp. 14.

BRAVO, M., HOYOS REYES, L.F. and REYES ORTIZ, J.A., 2019. Methodology for ontology design and construction. *Contaduría y administración*, 64(4),.

BRITISH MUSEUM., , Museum of the World. Available: <https://britishmuseum.withgoogle.com/> [05/06/, 2021].

BRITISH MUSEUM and GOOGLE CULTURAL INSTITUTE, 2015-last update, The Museum of the World. Available: <https://britishmuseum.withgoogle.com/> [08/22/, 2021].

BROOKE, J., 2013. SUS: a retrospective. *Journal of Usability Studies*, 8(2), pp. 29-40.

BUTLER, H., DALY, M., DOYLE, A., GILLIES, S., HAGEN, S., AND T. SCHAUB, 2016-last update, The GeoJSON Format. Available: <https://www.rfc-editor.org/info/rfc7946> [08/28/, 2021].

BUTLER, H., et al., 2016. The GeoJSON Format. *Internet Engineering Task Force (IETF)*, .

CALVO PALACIOS, J.L. and PUEYO CAMPOS, A., 1989. Mapas coropléticos e isopléticos y cartografía de potenciales de población. *Geographicalia*, .

- CANTONE, D. AND CRISTOFARO, S. AND NICOLOSI-ASMUNDO, M. AND PRADO, F. AND SANTAMARIA, D.F. AND SPAMPINATO, D., 2019. An EpiDoc ontological perspective: the epigraphs of the Castello Ursino Civic Museum of Catania via CIDOC CRM. *Archeologia e Calcolatori*, , pp. 139-157.
- CAR, N.J. and HOMBURG, T., 2021. *GeoSPARQL 1.1: an almost decadal update to the most important geospatial LOD standard*. CEUR-WS.
- CARRIERO, V., GANGEMI, A., MANCINELLI, M., MARINUCCI, L., NUZZOLESE, A., PRESUTTI, V. and VENINATA, C., 2019. *ArCo: the Italian Cultural Heritage Knowledge Graph*.
- CESIUM GS. INC., 2022-last update, Cesium JS. Available: <https://cesium.com/cesiumjs/> [Jan 31, 2022].
- CHAUDHRI, V.K., FARQUHAR, A., FIKES, R., KARP, P.D. and RICE, J.P., 1998. *Open knowledge base connectivity 2.0*.
- CHEN, P., LU, Y., ZHENG, V.W., CHEN, X. and YANG, B., 2018. KnowEdu: A System to Construct Knowledge Graph for Education. *IEEE Access*, **6**, pp. 31553-31563.
- CHEN, X., JIA, S. and XIANG, Y., 2020. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, **141**, pp. 112948.
- CIAMPAGLIA, G.L., SHIRALKAR, P., ROCHA, L.M., BOLLEN, J., MENCZER, F. and FLAMMINI, A., 2015. Computational Fact Checking from Knowledge Networks. *PloS One*, **10**(6), pp. e0128193.
- CIMMINO, A., FERNÁNDEZ-IZQUIERDO, A. and GARCÍA-CASTRO, R., 2020Astrea: Automatic Generation of SHACL Shapes from Ontologies, A. HARTH, S. KIRrane, A. NGONGA NGOMO, H. PAULHEIM, A. RULA, A.L. GENTILE, P. HAASE and M. COCHEZ, eds. In: *The Semantic, Web 2020*, Springer International Publishing, pp. 497-513.
- CLARAMUNT, C., 2020. Ontologies for geospatial information: Progress and challenges ahead. *Journal of Spatial Information Science*, **20**, pp. 35-41.
- CLAUS O. WILKE, 2019. *Fundamentals of Data Visualization*. 1 edn. O'Really Media, Inc.
- CLOONAN, M.V., 2014. *Preserving our heritage: Perspectives from Antiquity to the Digital Age*. Chicago (USA): American Library Association.
- CONTRERAS, J., BENJAMINS, V.R., BLÁZQUEZ, M., LOSADA, S., SALLA, R., SEVILLA, J., NAVARRO, D., CASILLAS, J., MOMPÓ, A., PATÓN, D., CORCHO, O., TENA, P. and MARTOS, I., 2004A Semantic Portal for the International Affairs Sector, E. MOTTA, N.R. SHADBOLT, A. STUTT and N. GIBBINS, eds. In: *Engineering Knowledge in the Age of the Semantic, Web 2004*, Springer Berlin Heidelberg, pp. 203-215.
- CORCHO, O., FERNÁNDEZ-LÓPEZ, M., GÓMEZ-PÉREZ, A. and LÓPEZ-CIMA, A., 2005. Building Legal Ontologies with METHONTOLOGY and WebODE. In: V.R. BENJAMINS, P. CASANOVAS, J. BREUKER and A. GANGEMI, eds, *Law and the Semantic Web: Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 142-157.

- CORDIS, 2005-last update, ESPERONTO SERVICES PROJECT. Available: <https://cordis.europa.eu/project/id/IST-2001-34373> [Jun 10, 2021].
- COUNCIL OF EUROPE, 2021-last update, Cultural Routes Database. Available: <https://www.coe.int/en/web/cultural-routes/cultural-routes-database-main-page> [Sep 30, 2021].
- COX, S.J.D. and RICHARD, S.M., 2015. A geologic timescale ontology and service. *Earth Science Informatics*, **8**(1), pp. 5-19.
- CRONOBOOK, 2021-last update, Cronobook. Available: <https://cronobook.com/> [08/22/, 2021].
- CYARK, 2021-last update, Cyark Cultural Heritage Explorer. Available: <https://www.cyark.org/explore/> [08/25/, 2021].
- DAAT HAMAKOM, 2017-last update, Jewish Cultures Mapped . Available: <http://www.jewish-cultures-mapped.org> [08/24/, 2021].
- DATA FOR HISTORY CONSORTIUM, 2018-last update, OntoME (Ontology Management Environment). Available: <https://ontome.dataforhistory.org/>.
- DCS. UNIV. IDAHO, 2019-last update, Palontology Information Explorer Pilot. Available: <http://www2.cs.uidaho.edu/%7Emax/gts/> [08/25/, 2021].
- DEBATTISTA, K., BUGEJA, K., SPINA, S., BASHFORD-ROGERS, T. and HULUSIC, V., 2018. Frame Rate vs Resolution: A Subjective Evaluation of Spatiotemporal Perceived Quality Under Varying Computational Budgets. *Computer Graphics Forum*, **37**(1), pp. 363-374.
- DEVERIA, A. and LENNART, S., 2022. *WebGL - 3D Canvas graphics*.
- DIETMAR, A., 2021a-last update, Chronas development history. Available: <https://www.kickstarter.com/projects/1152044848/chronas-interactive-history-map-application/description> [08/23/, 2021].
- DIETMAR, A., 2021b-last update, Chronas Project. Available: <https://chronas.org> [08/24/, 2021].
- DIETMAR, A., 2021c-last update, Github. Proyecto Chronas. Available: <https://github.com/Chronasorg/chronas> [08/24/, 2021].
- DIJKSHOORN, C., AROYO, L., VAN OSSENBRUGGEN, J. and SCHREIBER, G., 2018. Modeling cultural heritage data for online publication. *Applied Ontology*, **13**, pp. 255-271.
- DOERR, M., GRADMANN, S., HENNICKE, S., ISAAC, A., MEGHINI, C., VAN DE SOMPEL, E., 2021. The europeana data model (EDM) . *World Library and Information Congress: 76th IFLA general conference and assembly*, **10**.
- DONG, X.L., GABRILOVICH, E., HEITZ, G., HORN, W., MURPHY, K., SUN, S. and ZHANG, W., 2014. From data fusion to knowledge fusion, 2014, pp. 881-892.

DONG, X., GABRILOVICH, E., HEITZ, G., HORN, W., LAO, N., MURPHY, K., STROHMANN, T., SUN, S. and ZHANG, W., 2014. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. *KDD '14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: Association for Computing Machinery, pp. 601-610.

DOU, D., WANG, H. and LIU, H., 2015 Semantic data mining: A survey of ontology-based approaches, 2015, pp. 244-251.

DUDÁŠ, M., LOHMANN, S., SVÁTEK, V. and PAVLOV, D., 2018. Ontology visualization methods and tools: a survey of the state of the art. *Knowledge Engineering Review*, **33**.

ECLIPSE FOUNDATION, Jan 23, 2022-last update, RDF4J. Available: <https://rdf4j.org/> [Jan 28, 2022].

EHRLINGER, L., & WÖSS, W., 2016. Towards a Definition of Knowledge Graphs. *SEMANTICS*.

ELKAN, C. and GREINER, R., 1993. Building large knowledge-based systems: Representation and inference in the cyc project: D.B. Lenat and R.V. Guha. *Artificial Intelligence*, **61**(1), pp. 41-52.

ELNAGDY, S., QIU, M. and GAI, K., 2016. Understanding Taxonomy of Cyber Risks for Cybersecurity Insurance of Financial Industry in Cloud Computing. *undefined*, .

ERLING, O., 2012. Virtuoso, a Hybrid RDBMS/Graph Column Store. *IEEE Data Eng.Bull.*, **35**(1), pp. 3-8.

ESCOBAR, P., ROLDÁN-GARCÍA, M.D., PERAL, J., CANDELA, G. and GARCÍA-NIETO, J., 2020. *An Ontology-Based Framework for Publishing and Exploiting Linked Open Data: A Use Case on Water Resources Management*.

ESPINOZA-ARIAS, P., GARCÍA-DELGADO, M., CORCHO, O., VIVAS-WHITE, P. and POTTI-MANJAVACAS, H., 2019. A sustainable process and toolbox for geographical linked data generation and publication: a case study with BTN100. *Open Geospatial Data, Software and Standards*, **4**.

ESTER M, KRIEGEL HP, SANDER J, XU X, 1996 A density-based algorithm for discovering clusters in large spatial databases with noise, *Second International Conference on Knowledge Discovery and Data Mining* 1996, pp. 226-231.

EUROPEAN COMMISSION, 2017. *Mapping of Cultural Heritage action in European Union policies, programmes and activities*. European Commission.

EUROPEAN COMMISSION, 2021. *Commission Staff Working Document: Evaluation of the Commission Recommendation of 27 October 2011 on the digitisation and online accessibility of cultural material and digital preservation*.

EUROPEAN COMMISSION, 2019-last update, INSPIRE Knowledge Base. Infrastructure for spatial information in Europe. Available: <https://inspire.ec.europa.eu/> [04/04/, 2021].

EUROPEANA, 04/04/, 2021-last update, EUROPEANA. Available: <https://pro.europeana.eu/>.



EVANS, A., ROMEO, M., BAHREHMAND, A., AGENJO, J. and BLAT, J., 2014. 3D graphics on the web: A survey. *Computers & Graphics*, **41**, pp. 43-61.

FALCONER, S.M., BULL, R.I., GRAMMEL, L. and STOREY, M., 2009. Creating Visualizations through Ontology Mapping, 2009, pp. 688-693.

FARQUHAR, A., FIKES, R. and RICE, J., 1997. The Ontolingua Server: a tool for collaborative ontology construction. *International Journal of Human-Computer Studies*, **46**(6), pp. 707-727.

FELLBAUM, C.(.), 1998. *WordNet An Electronic Lexical Database*. Cambridge: MA: MIT Press.

FENSEL, D., ŞİMŞEK, U., ANGELE, K., HUAMAN, E., KÄRLE, E., PANASIUK, O., TOMA, I., UMBRICH, J. and WAHLER, A., 2020. Why We Need Knowledge Graphs: Applications. In: D. FENSEL, U. ŞİMŞEK, K. ANGELE, E. HUAMAN, E. KÄRLE, O. PANASIUK, I. TOMA, J. UMBRICH and A. WAHLER, eds, *Knowledge Graphs: Methodology, Tools and Selected Use Cases*. Cham: Springer International Publishing, pp. 95-112.

FERNÁNDEZ, G.A.M., PÉREZ-GÁLVEZ, J.C. and LÓPEZ-GUZMÁN, T., 2017. Tourist motivations in a heritage destination in Spain. *European Journal of Tourism, Hospitality and Recreation*, **7**(3), pp. 226-235.

FERNANDEZ-LOPEZ, M., GOMEZ-PEREZ, A., JURISTO, N., March, 1997. METHONTOLOGY: from Ontological Art towards Ontological Engineering | BibSonomy, *AAAI97 Spring Symposium*, March, 1997 March, 1997, pp. 33-40.

FILHO, F.T. and VIEIRA, M. D. F. Q., 2014. An ontology-driven framework to support scenario representation in a 3D operator training simulator, 2014, pp. 298-303.

FISH, 2003-last update, Vocabularies of FISH. Available: <http://www.heritage-standards.org.uk/fish-vocabularies/> [08/12/, 2021].

FISH, 1998-last update, Forum on Information Standards in Heritage (FISH). Available: <https://www.heritagedata.org> [08/12/, 2021].

GARCIA MORENO, A., HERNÁNDEZ-PÉREZ, T., 2019. Patrimonio cultural europeo digitalizado: Europeaana. *PragMATIZES - Revista Latino-Americana de Estudos em Cultura* **16**, .

GARCIA, E., 2013-last update, Turm.JS. Available: <http://www.turnjs.com/> [Jan 15, 2022].

GARCÍA-MARCO, F.-., 2016. THE EVOLUTION OF THESAURI AND THE HISTORY OF KNOWLEDGE ORGANIZATION: BETWEEN THE SWORD OF MAPPING KNOWLEDGE AND THE WALL OF KEEPING IT SIMPLE. *undefined*, .

GARROTE, A., 2016-last update, rdfstore-js Github. Available: <https://github.com/antoniojarrote/rdfstore-js> [Feb 20, 2022].

GETTY CONSERVATION INSTITUTE and WMF, 2021-last update, Arches Project. Available: <https://www.archesproject.org/> [08/22/, 2021].

GETTY, R.I., 2005-last update, The Cultural Objects Name Authority Thesaurus of the Getty Research Institute. Available: <https://www.getty.edu/research/tools/vocabularies/cona> [08/12/, 2021].

GETTY, R.I., 1997a-last update, The Art & Architecture Thesaurus of the Getty Research Institute. Available: <https://www.getty.edu/research/tools/vocabularies/aat/> [08/12/, 2021].

GETTY, R.I., 1997b-last update, The Geographic Names Thesaurus of Getty Research Institute. . Available: <https://www.getty.edu/research/tools/vocabularies/tgn> [08/12/, 2021].

GETTY, R.I., 1997c-last update, The Union List of Artist Names Thesaurus of Getty Research Institute. . Available: <https://www.getty.edu/research/tools/vocabularies/ulan> [08/12/, 2021].

GLIMM, B., HORROCKS, I., MOTIK, B., STOILLOS, G. and WANG, Z., 2014. Hermit: An OWL 2 Reasoner. *Journal of Automated Reasoning*, **53**(3), pp. 245-269.

GOLDEN, P. and SHAW, R., 2016. Nanopublication beyond the sciences: the PeriodO period gazetteer. *PeerJ Computer Science*, **2**.

GOMEZ-PEREZ, A. and CORCHO, O., 2002. Ontology languages for the Semantic Web. *IEEE Intelligent Systems*, **17**(1), pp. 54-60.

GONZÁLEZ-PÉREZ, C., 2018. An Ontology for Cultural Heritage. In: SPRINGER, ed, *Information Modelling for Archaeology and Anthropology*. Springer International Publishing, pp. 195-215.

GONZALEZ-PEREZ, C., MARTÍN-RODILLA, P., PARCERO-OUBIÑA, C., FÁBREGA-ÁLVAREZ, P. and GÜIMIL-FARIÑA, A., 2012 Extending an Abstract Reference Model for Transdisciplinary Work in Cultural Heritage, *6th Metadata and Semantics Research Conference (MTSR 2012)*, 11/28/2021 2012.

GOOGLE, 2021a-last update, Google Arts & Culture . Available: <https://artsandculture.google.com/> [04/04/21, .

GOOGLE, 2021b-last update, Google Arts & Culture. Available: <https://artsandculture.google.com> [08/22/, 2021].

GOOGLE, 2021c-last update, Google Maps API. Available: <https://developers.google.com/maps/> [08/25/, 2021].

GOOGLE., 2021-last update, Service worker. Chrome. Available: <https://developers.google.com/web/fundamentals/primers/service-workers> [Jan 18, 2022].

GREEN, C. G., AND VILLA, I., 2019. *Taking City Visualization into the Third Dimension with Point Clouds, 3D Tiles, and deck.gl*.

GRUNINGER , M., OBRST, L., KUTZ, O., LANGE , C., LEVENCHUK, A., QUATTRI, F., SCHNEIDER, T., SPERO, S., THESEN, A., VEGETTI, M., VIZEDOM, A., 2021. *Ontology Summit 2014 Communiqué Semantic Web and Big Data Meets Applied Ontology*.

GUARINO, N., 1998. *Formal Ontologies and Information Systems*.

GUILIANO, J. and HEITMAN, C., 2019. Difficult Heritage and the Complexities of Indigenous Data. *Journal of Cultural Analytics*, **4**(1), pp. 11041.

GUO, D., 2009. Flow Mapping and Multivariate Visualization of Large Spatial Interaction Data. *IEEE Transactions on Visualization and Computer Graphics*, **15**(6), pp. 1041-1048.

HAARSLEV, V. and MÖLLER, R., 2003. Racer: An OWL Reasoning Agent for the Semantic Web. *undefined*, .

HAASE, P., BROEKSTRA, J., EBERHART, A. and VOLZ, R., 2004A Comparison of RDF Query Languages, S.A. MCILRAITH, D. PLEXOUSAKIS and F. VAN HARMELEN, eds. In: *The Semantic Web – ISWC 20, 04 2004*, Springer Berlin Heidelberg, pp. 502-517.

HELLMUND, T., HERTWECK, P., HILBRING, D., MOSSGRABER, J., ALEXANDRAKIS, G., POULI, P., SIATOU, A. and PADELETTI, G., 2018. *Introducing the HERACLES Ontology—Semantics for Cultural Heritage Management*.

HELSINGIN KAUPUNGINMUSEO, 2015-last update, Helsinki ennen. Available: <http://karttalehtinen.fi/helsinki/> [05/06/, 2021].

HEWAGAMAGE, K.P., HIRAKAWA, M. and ICHIKAWA, T., 1999Interactive visualization of spatiotemporal patterns using spirals on a geographical map, 1999, pp. 296-303.

HIEBEL, G., MARTIN, D., OVIIND, E. and THEODORIDOU, M., Sep, 2015-last update, CRMgeo Spatiotemporal model. Available: <https://www.cidoc-crm.org/crmgeo/home-5> [Nov 15, 2021].

HINDLE, A., Jun 23, 2017-last update, How to create a virtual 3D gallery using IIIF and Three.js. Available: <https://blog.cogapp.com/how-to-create-a-virtual-3d-gallery-using-iiif-and-three-js-ad6766e45d3f> [Jan 15, 2021].

HOLLISTER, S., 2020. *Unity's IPO filing shows how big a threat it poses to Epic and the Unreal Engine*. The Verge.

HORROCKS, I., FENSEL, D., BROEKSTRA, J., DECKER, S., ERDMANN, M., GOBLE, C., HARMELEN, F.V., KLEIN, M., STAAB, S., STUDER, R. and MOTTA, E., 2000. The Ontology Inference Layer OIL. *undefined*, .

ICCROM, 2021-last update, ICCROM. Available: <https://www.iccrom.org/> [20/05/, 2021].

IGN, 2007. *El sistema ETRS89 y la nueva cartografía*. IGN.

IMATEX, 2021-last update, Textiloteca del Museo Textil de Tarrassa. . Available: [http://imatex.cdm.t.cat/\\_cat/fitxa\\_fitxa.aspx?m=n&num\\_id=8238](http://imatex.cdm.t.cat/_cat/fitxa_fitxa.aspx?m=n&num_id=8238) [04/04/, 2021].

INE, 2022-last update, Callejero censo electoral. Available: [https://www.ine.es/ss/Satellite?L=es\\_ES&c=Page&cid=1259952026632&p=1259952026632&p\\_agename=ProductosYServicios%2FPYSLayout](https://www.ine.es/ss/Satellite?L=es_ES&c=Page&cid=1259952026632&p=1259952026632&p_agename=ProductosYServicios%2FPYSLayout) [Feb 20, 2022].

INTERREG. and STONIA-LATVIA, 2018-last update, Industrial Heritage Tourism. Available: <https://industrialheritage.travel/map> [Sep 30, 2021].

ISAAC, A., ZINN, C., MATTHEZING, H., MEIJ, H.V.D., SCHLOBACH, S. and WANG, S., 2007. The value of usage scenarios for thesaurus alignment in cultural heritage context. *undefined*, .

ISO, 2021-last update, ISO 19108:2002 Geographic information - Temporal schema. Available: <https://www.iso.org/standard/26013.html> [Nov 15, 2021].

ISO, 2019-last update, ISO 19111:2007  
Geographic information — Spatial referencing by coordinates. Available: <https://www.iso.org/standard/41126.html> [Nov 15, 2021].

ISO, 2014a. *ISO 21127:2014*  
*Information and documentation — A reference ontology for the interchange of cultural heritage information.*

ISO, 2014b-last update, ISO 9186-1:2014. Available: <https://www.iso.org/standard/59226.html> [Apr 04, 2022].

ISO, 2013. *ISO 25964-2:2013*  
*Information and documentation — Thesauri and interoperability with other vocabularies — Part 2: Interoperability with other vocabularies.* International Organization for Standardization (ISO).

ISO, 2011. *ISO 25964-1:2011*  
*Information and documentation — Thesauri and interoperability with other vocabularies — Part 1: Thesauri for information retrieval.* International Organization for Standardization (ISO).

ISO, 2010-last update, ISO 81714-1:2010. Available: <https://www.iso.org/obp/ui/es/#iso:std:iso:81714:-1:ed-2:v1:en> [Apr 04, 2022].

ISO, 2007-last update, ISO 22727:2007. Available: <https://www.iso.org/obp/ui/#iso:std:iso:22727:ed-1:v1:en> [Apr 04, 2022].

ISO, 1986. *ISO 2788:1986*  
*Documentation — Guidelines for the establishment and development of monolingual thesauri.* International Organization for Standardization (ISO).

JAFFALI, S., JAMOOUSSI, S., KHELIFI, N. and HAMADOU, A.B., 2020Survey on Social Networks Data Analysis, S.S. RAUTARAY, G. EICHLER, C. ERFURTH and G. FAHRNBERGER, eds. In: *Innovations for Community, Services 2020*, Springer International Publishing, pp. 100-119.

JAN NAUTA, G., VAN DEN HEUVEL, W., 2015. *ENUMERATE Survey Report on Digitisation in European Cultural Heritage Institutions 2015*. Europeana/ENUMERATE.

JÄNICKE, S., HEINE, C. and SCHEUERMANN, G., 2013GeoTemCo: Comparative Visualization of Geospatial-Temporal Data with Clutter Removal Based on Dynamic Delaunay Triangulations, G. CSURKA, M. KRAUS, R.S. LARAMEE, P. RICHARD and J. BRAZ, eds. In: *Computer Vision, Imaging and Computer Graphics. Theory and, Application 2013*, Springer Berlin Heidelberg, pp. 160-175.

JEEN BROEKSTRA AND ARJOHN KAMPMAN., 2004. *SeRQL: An RDF Query and Transformation Language DRAFT*.

JI, S., PAN, S., CAMBRIA, E., MARTTINEN, P. and YU, P.S., 2021. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE Transactions on Neural Networks and Learning Systems*, , pp. 1-21.

JOHN HOPKINS. U. MEDICINE., 2021-last update, COVID-19 Dashboard. Available: <https://coronavirus.jhu.edu/map.html> [08/28/, 2021].

JOSHI, A., KALE, S., CHANDEL, S. and PAL, D.K., 2015. Likert Scale: Explored and Explained. *Current Journal of Applied Science and Technology*, , pp. 396-403.

JOVANOVIK, M. and SPASIĆ, M., 2018Benchmarking Virtuoso 8 at the Mighty Storage Challenge 2018: Challenge Results, D. BUSCALDI, A. GANGEMI and D. REFORGIATO RECUPERO, eds. In: *Semantic Web, Challenges 2018*, Springer International Publishing, pp. 24-35.

KALIBATIENE, D. and VASILECAS, O., 2011Survey on Ontology Languages, J. GRABIS and M. KIRIKOVA, eds. In: *Perspectives in Business Informatics, Research 2011*, Springer Berlin Heidelberg, pp. 124-141.

KARP, P., CHAUDHRI, V. and THOMÉRE, J., 2000. XOL: An XML-Based Ontology Exchange Language. *undefined*, .

KARVOUNARAKIS, G., ALEXAKI, S., CHRISTOPHIDES, V., PLEXOUSAKIS, D. and SCHOLL, M., 2002. *RQL: A Declarative Query Language for RDF*. Association for Computing Machinery.

KHRONOS GROUP, 2022-last update, OpenGL ES. Available: <https://www.khronos.org/opengles/2022>].

KÖBBEN, B.J. and KRAAK, M., 2020. Mapping, Web. In: A. KOBAYASHI, ed, *International Encyclopedia of Human Geography (Second Edition)*. Oxford: Elsevier, pp. 333-337.

KOCH, I., RIBEIRO, C. and TEIXEIRA LOPES, C., 2020ArchOnto, a CIDOC-CRM-Based Linked Data Model for the Portuguese Archives, M. HALL, T. MERČUN, T. RISSE and F. DUCHATEAU, eds. In: *Digital Libraries for Open, Knowledge 2020*, Springer International Publishing, pp. 133-146.

KONYS, A., 2017Ontology-Based Approaches to Big Data Analytics, S. KOBAYASHI, A. PIEGAT, J. PEJAŚ, I. EL FRAY and J. KACPRZYK, eds. In: *Hard and Soft Computing for Artificial Intelligence, Multimedia and, Security 2017*, Springer International Publishing, pp. 355-365.

KRAAK, M.J., 2005. *Timelines, temporal resolution, temporal zoom and time geography*. International Cartographic Association.

KU, W., LIAW, Y., HUANG, J., NFOR, O.N., HSU, S., KO, P., LEE, W. and CHEN, C., 2016. An Online Atlas for Exploring Spatio-Temporal Patterns of Cancer Mortality (1972–2011) and Incidence (1995–2008) in Taiwan. *Medicine*, **95**(21),.

KULAWIAK, M., DAWIDOWICZ, A. and PACHOLCZYK, M.E., 2019. Analysis of server-side and client-side Web-GIS data processing methods on the example of JTS and JSTS using open data from OSM and geoportal. *Computers & Geosciences*, **129**, pp. 26-37.

LABRA-GAYO, J., GARCÍA-GONZÁLEZ, H., FERNÁNDEZ-ALVAREZ, D. and PRUD'HOMMEAUX, E., 2019. Challenges in RDF Validation. In: G. ALOR-HERNÁNDEZ, J.L. SÁNCHEZ-CERVANTES, A.

RODRÍGUEZ-GONZÁLEZ and R. VALENCIA-GARCÍA, eds, *Current Trends in Semantic Web Technologies: Theory and Practice*. Cham: Springer International Publishing, pp. 121-151.

LAMARCA, M.J., 2018. *HIPERTEXTO: EL NUEVO CONCEPTO DE DOCUMENTO EN LA CULTURA DE LA IMAGEN*, Universidad Complutense de Madrid.

LASH, S., 2017. Review of The Museum of the World [website]. *Internet Archaeology*, .

LAVALLE, S., LESSER, E., SHOCKLEY, R., HOPKINS, M.S., KRUSCHWITZ, N., 2011. Big data, analytics and the path from insights to value. *MIT Sloan Manag*, , pp. 21-32.

LEAVER, D., 2012-last update, Leaflet.markercluster. Available: <https://github.com/Leaflet/Leaflet.markercluster> [08/16/, 2021].

LEE, C., DEVILLERS, R. and HOEBER, O., 2014. Navigating spatio-temporal data with temporal zoom and pan in a multi-touch environment. *International Journal of Geographical Information Science*, **28**(5), pp. 1128-1148.

LEHMANN, J., ISELE, R., JAKOB, M., JENTZSCH, A., KONTOKOSTAS, D., MENDES, P.N., HELLMANN, S., MORSEY, M., VAN KLEEF, P., AUER, S. and BIZER, C., 2015. DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, **6**, pp. 167-195.

LEÓN, A., GAITÁN, M., SEBASTIAN, J., ALBA, E. and INSA, I., 2020SILKNOW. Designing a thesaurus about historical silk for small and medium-sized textile museums, *4th International Congress Science and Technology for the Conservation of Cultural Heritage (TechnoHeritage 2019)* 2020, pp. 187-190.

LINDBERG, A., 2020. *Performance Evaluation of JavaScript Rendering Frameworks*.

LIU, J., 2018. *Combining Enterprise Knowledge Graph and News Sentiment Analysis for Stock Price Volatility Prediction*.

LIU, S., CUI, W., WU, Y. and LIU, M., 2014. A survey on information visualization: recent advances and challenges. *The Visual Computer*, **30**(12), pp. 1373-1393.

LOHMANN, S., LINK, V., MARBACH, E. and NEGRU, S., 2015WebVOWL: Web-based Visualization of Ontologies, P. LAMBRIX, E. HYVÖNEN, E. BLOMQVIST, V. PRESUTTI, G. QI, U. SATTLER, Y. DING and C. GHIDINI, eds. In: *Knowledge Engineering and Knowledge Management 2015*, Springer International Publishing, pp. 154-158.

LOUVRE, 2021-last update, Louvre Collections. Available: <https://collections.louvre.fr/> [04/04/, 2021].

LUCERNE, U., 2021-last update, Hochschule Luzern (Lucerne University). Available: <https://www.hslu.ch> [21/05/, 2021].

LUKE, S. and HEFLIN, J., 2000-last update, SHOE Project. Available: <http://www.cs.umd.edu/projects/plus/SHOE/spec1.01.html> [09/01/, 2021].

MACQUEEN, J., 1967 Some methods for classification and analysis of multivariate observations, *Berkeley Symposium on Mathematical Statistics and Probability*, 1967 1967, pp. 281-297.

MALAVOLTA, I., PROCACCIANTI, G., NOORLAND, P. and VUKMIROVIC, P., 2017 Assessing the Impact of Service Workers on the Energy Efficiency of Progressive Web Apps, 2017, pp. 35-45.

MANIRAJ, V. and SIVAKUMAR, R., 2010. Ontology Languages – A Review. *International Journal of Computer Theory and Engineering*, **2**, pp. 887-891.

MAPBOX, 2021-last update, Mapbox. Available: <https://www.mapbox.com/> [08/22/, 2021].

MAPTIVE, 2022-last update, Maptive Heat Mapping Tool. Available: <https://answers.maptive.com/article/219-how-to-use-the-heat-mapping-tool> [Feb 20, 2022].

MARTÍNEZ-GONZÁLEZ, M.M. and ALVITE-DÍEZ, M., 2020. A semantic web methodological framework to evaluate the support of integrity in thesaurus tools. *Journal of Information Science*, **46**(3), pp. 378-391.

MARTIN-RODILLA, P. & GONZALEZ-PEREZ, C., October, 2018 Representing Imprecise and Uncertain Knowledge in Digital Humanities: A Theoretical Framework and ConML Implementation with a Real Case Study, *Technological Ecosystems for Enhancing Multiculturality* October, 2018, Association for Computing Machinery, pp. 863–871.

MATUSZEK, C., CABRAL, J., WITBROCK, M., DEOLIVEIRA, J., 2006 An Introduction to the Syntax and Content of Cyc, *AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, 2006 2006, Association for the Advancement of Artificial Intelligence (AAAI).

MCGUINNESS, D.L., FIKES, R., HENDLER, J. and STEIN, L.A., 2002. DAML+OIL: an ontology language for the Semantic Web. *IEEE Intelligent Systems*, **17**(5), pp. 72-80.

MCKEAGUE, P., VAN'T VEER, R., HUVILA, I., MOREAU, A., VERHAGEN, P., BERNARD, L., COOPER, A., GREEN, C. AND VAN MANEN, N., 2019. Mapping Our Heritage: Towards a Sustainable Future for Digital Spatial Information and Technologies in European Archaeological Heritage Management. *Journal of Computer Applications in Archaeology*, **2**, pp. 89-104.

MEGHINI, C., NICCOLUCCI, F., FELICETTI, A., RONZINO, P., NURRA, F., PAPATHEODOROU, C., GAVRILIS, D., THEODORIDOU, M., DOERR, M., TUDHOPE, D., BINDING, C., SCOPIGNO, R., VLACHIDIS, A., RICHARDS, J., WRIGHT, H., GESER, G., CUY, S., FIHN, J., FANINI, B. and HOLLANDER, H., 2017. ARIADNE: A research infrastructure for archaeology. *Journal on Computing and Cultural Heritage*, **10**, pp. 1-27.

MENOR-CAMPOS, A., PÉREZ-GÁLVEZ, J.C., HIDALGO-FERNÁNDEZ, A. and LÓPEZ-GUZMÁN, T., 2020. *Foreign Tourists in World Heritage Sites: A Motivation-Based Segmentation*.

MEROÑO-PENUELA, A. and HOEKSTRA, R., 2016 grlc Makes GitHub Taste Like Linked Data APIs, H. SACK, G. RIZZO, N. STEINMETZ, D. MLADENIĆ, S. AUER and C. LANGE, eds. In: *The Semantic Web 2016*, Springer International Publishing, pp. 342-353.



META PLATFORMS, 2022-last update, React Project. Available: <https://es.reactjs.org> [Jan 20, 2022].

MILES, A. and BECHHOFFER, S., 2009. *SKOS Simple Knowledge Organization System : W3C Recommendation*. W3C.

MILLER, L., SEABORNE, A. and REGGIORI, A., 2002 Three Implementations of SquishQL, a Simple RDF Query Language, I. HORROCKS and J. HENDLER, eds. In: *The Semantic Web — ISWC 20, 02 2002*, Springer Berlin Heidelberg, pp. 423-435.

MINGEI, 2020-last update, Mingei Project. Available: <https://www.mingei-project.eu> [20/05/2020], .

MINISTERIO DE CULTURA Y DEPORTE., 2021-last update, CERES. Red Digital de Colecciones de Museos de España. Available: <http://ceres.mcu.es> [04/04/, 2021].

MITCHELL, T., COHEN, W., HRUSCHKA, E., TALUKDAR, P., YANG, B., BETTERIDGE, J., CARLSON, A., DALVI, B., GARDNER, M., KISIEL, B., KRISHNAMURTHY, J., LAO, N., MAZAITIS, K., MOHAMED, T., NAKASHOLE, N., PLATANIOS, E., RITTER, A., SAMADI, M., SETTLES, B., WANG, R., WIJAYA, D., GUPTA, A., CHEN, X., SAPAROV, A., GREAVES, M. and WELLING, J., 2018. Never-ending learning. *Communications of the ACM*, **61**(5), pp. 103-115.

MRDOOB, 2022-last update, Three.JS Github. Available: <https://github.com/mrdoob/three.js> [Jan 31, 2022].

MUSEO DEL PRADO., 2021-last update, Museo del Prado Colección. . Available: <https://www.museodelprado.es/coleccion> [04/04/, 2021].

MUSEU PICASSO, May 26, 2020-last update, Ontología Museo Picasso. Available: <http://museupicassobcn.org/ontologia/> [Nov 10, 2021].

NAFIS, F., YAHYAOUY, A. and AGHOUTANE, B., 2019. *Ontologies for the classification of cultural heritage data*.

NAH, F.F., 2004. A study on tolerable waiting time: how long are Web users willing to wait? *null*, **23**(3), pp. 153-163.

NATALYA F. NOY AND DEBORAH L. MCGUINNESS., 2001. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880.

NATIONAL CULTURAL FUND OF HUNGARY, , Hungaricana Cultural Heritage Portal. Available: <https://hungaricana.hu/en/> [05/06/, 2021].

NAZEMI, K., BURKHARDT, D., GINTERS, E. and KOHLHAMMER, J., 2015. Semantics Visualization – Definition, Approaches and Challenges. *Procedia Computer Science; 2015 International Conference Virtual and Augmented Reality in Education*, **75**, pp. 75-83.

NEMO WORKING GROUP DIGITALISATION AND IPR, 2020. *Digitisation and IPR in European Museums*. . Network of European Museum Organisations.



NGUYEN ,H., MAU QUOC, MARTIN SERRANO, HAN NGUYEN MAU, JOHN G. BRESLIN, DANH LE PHUOC, 2019. A Performance Study of RDF Stores for Linked Sensor Data. *Vixra*, .

NIEKE, S., Mar 21, 2021-last update, D3-3d. Available: <https://github.com/Nieked3-3d> [Jan 18, 2022].

NIELSEN, J., 1994. *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

NISHANBAEV, I., CHAMPION, E. and MCMEEKIN, D.A., 2019. *A Survey of Geospatial Semantic Web for Cultural Heritage*.

NOY, N.F. and MCGUINNESS, D.L., 2001. Ontology development 101: A guide to creating your first ontology.

OGC, 2018-last update, 3D Tiles Specification. Available: <https://www.ogc.org/standards/3DTiles> [Jan 14, 2022].

OLIVEIRA, D., Jan 13, 2020-last update, Not Science Fiction - Kurt Cagle's View on Semantic Technologies. Available: <https://2020-us.semantics.cc/not-science-fiction-kurt-cagles-view-semantic-technologies> [08/30/, 2021].

ONG, E., XIANG, Z., ZHAO, B., LIU, Y., LIN, Y., ZHENG, J., MUNGALL, C., COURTOT, M., RUTTENBERG, A. and HE, Y., 2017. Ontobee: A linked ontology data server to support ontology term dereferencing, linkage, query and integration. *Nucleic acids research*, **45**, pp. D347-D352.

OPEN GEOSPATIAL CONSORTIUM, 2011-last update, OGC GeoSPARQL - A Geographic Query Language for RDF Data. Available: <http://www.opengeospatial.org/standards/requests/80>.

OPEN, J.F., 2022-last update, Node.js Project. Available: <https://nodejs.org> [Jan 22, 2022].

OPENJS, F., 2022-last update, jQuery Project. Available: <https://jquery.com/> [Jan 18, 2022].

OPENLAYERS, 2021-last update, OpenLayers. Available: <https://openlayers.org> [08/29/, 2021].

OPENLINKSW, 2021-last update, Virtuoso OpenLinkSw . Available: <https://virtuoso.openlinksw.com/> [08/30/, 2021].

ORACLE, 2008-last update, Behavior Class Reference. Available: <https://download.java.net/media/java3d/javadoc/1.3.2/javafx/media/j3d/Behavior.html> [09/20/, 2021].

OSGF, 2021-last update, OpenLayers. Available: <https://openlayers.org/> [Nov 18, 2021].

OSMF, 2021-last update, OpenStreetMap. Available: <https://www.openstreetmap.org> [08/22/, 2021].

PADFIELD, J., KONTIZA, K., BIKAKIS, A. and VLACHIDIS, A., 2019. *Semantic Representation and Location Provenance of Cultural Heritage Information: the National Gallery Collection in London*.

- PAGÁN, E.A., SALVATELLA, M.D., PITARCH, M.D., MUÑOZ, A.L., TOLEDO, M.D., RUIZ, J.M., VITELLA, M., LO CICERO, G., ROTTENSTEINER, F., CLERMONT, D., DOROZYNSKI, M., WITTICH, D., VERNUS, P. and PUREN, M., 2020. *From Silk to Digital Technologies: A Gateway to New Opportunities for Creative Industries, Traditional Crafts and Designers. The SILKNOW Case.*
- PANTOJA, J., DOCAMPO, J., MARTÍN, A., MATURANA, R.A., NAVALÓN, C., 6/04/2016The new Prado Museum website: A (semantic) challenge, *The annual conference of Museums and the Web*, 06/04/2016 6/04/2016.
- PAULHEIM, H., 2017. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, **8**(3), pp. 489-508.
- PELLISSIER TANON, T., VRANDEČIĆ, D., SCHAFFERT, S., STEINER, T. and PINTSCHER, L., 2016. From Freebase to Wikidata: The Great Migration. *WWW '16: Proceedings of the 25th International Conference on World Wide Web*. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, pp. 1419-1428.
- PÉREZ-URBINA, H., SIRIN, E. and CLARK, K., 2021. *Validating RDF with OWL Integrity Constraints*. Stardof.
- PERRY, M., HERRING, J., CAR, N.J., HOMBURG, T. and J.D. COX, S., *OGC GeoSPARQL - A geographic query language for RDF data: GeoSPARQL 1.1 draft.*
- PEUQUET, D.J., 1994. It's About Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. *null*, **84**(3), pp. 441-461.
- PLA-VIVAS, V., 2021. THE AGE OF ICONICAL PROLIFERATION: POSTMODERNIST WORKS IN THE VALENCIAN ARCHIVE OF DESIGN. *JOURNAL OF MODERNISM AND POSTMODERNISM STUDIES (JOMOPS)*, **2**(2), pp. 355-384.
- POLIKOFF, I., May 23, 2021-last update, Why I Don't Use OWL Anymore. Available: <https://www.topquadrant.com/owl-blog/> [08/30/, 2021].
- POLOWINSKI, J., 2015-last update, Semantics & Visualization [08/21/, 2021].
- POLOWINSKI, J., Jun 12, 2013Towards RVL, Jun 12, 2013, ACM, pp. 1-11.
- POLOWINSKI, J. and VOIGT, M., 2013. VISO: a shared, formal knowledge base as a foundation for semi-automatic infovis systems. *CHI EA '13: CHI '13 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, pp. 1791-1796.
- PORTALÉS, C., PÉREZ, M., CASANOVA-SALAS, P. and GIMENO, J., 2021. Virtual Loom: a tool for the interactive 3D representation of historical fabrics. *Multimedia Tools and Applications*, **80**(9), pp. 13735-13760.
- PORTALÉS, C., SEBASTIÁN, J., ALBA, E., SEVILLA, J., GAITÁN, M., RUIZ, P. and FERNÁNDEZ, M., 2018. *Interactive Tools for the Preservation, Dissemination and Study of Silk Heritage—An Introduction to the SILKNOW Project.*

PULIDO, J.R.G., RUIZ, M.A.G., HERRERA, R., CABELLO, E., LEGRAND, S. and ELLIMAN, D., 2006. Ontology languages for the semantic web: A never completely updated review. *Knowledge-Based Systems; Creative Systems*, **19**(7), pp. 489-497.

QUIAN, S., 07/31/, 2020a-last update, Github. Saborn Map Navigator Open Source. Available: <https://github.com/selenaqian/sanborn-maps-navigator> [08/24/, 2021].

QUIAN, S., 2020b-last update, Sanborn Map Navigator. Available: <https://selenaqian.github.io/sanborn-maps-navigator/> [08/24/, 2021].

RAMIS FERRER, B., MOHAMMED, W.M., AHMAD, M., IAROVYI, S., ZHANG, J., HARRISON, R. and MARTINEZ LASTRA, J.L., 2021. Comparing ontologies and databases: a critical review of lifecycle engineering models in manufacturing. *Knowledge and Information Systems*, **63**(6), pp. 1271-1304.

RICHENS, R.H., 1956. Preprogramming for mechanical translation . *Mechanical Translation*, **3**, pp. 20-25.

RIZUN, M., 2019. Knowledge Graph Application in Education: a Literature Review. *Acta Universitatis Lodzensis.Folia Oeconomica*, **3**(342), pp. 7-19.

ROTMENSCH, M., HALPERN, Y., TLIMAT, A., HORNG, S. and SONTAG, D., 2017. Learning a Health Knowledge Graph from Electronic Medical Records. *Scientific Reports*, **7**(1), pp. 5994.

RUDNIK, C., EHRHART, T., FERRET, O., TEYSSOU, D., TRONCY, R. and TANNIER, X., 2019. *Searching News Articles Using an Event Knowledge Graph Leveraged by Wikidata*.

SALEHPOUR, M. and DAVIS, J.G., 2020. A Comparative Analysis of Knowledge Graph Query Performance. *undefined*, .

SAMAROUDI, M., ECHAVARRIA, K.R. and PERRY, L., 2020. Heritage in lockdown: digital provision of memory institutions in the UK and US of America during the COVID-19 pandemic. *null*, **35**(4), pp. 337-361.

SAMPER ZAPATER, J.J., 2005. *ONTOLOGÍAS PARA SERVICIOS WEB SEMÁNTICOS DE INFORMACIÓN DE TRÁFICO: DESCRIPCIÓN Y HERRAMIENTAS DE EXPLOTACIÓN*, Universitat de València.

SAWSAA, A. and LU, J., 2014. Visualizing Information Science Knowledge by Modelling Domain Ontology (OIS). *International Journal of Information Retrieval Research (IJIRR)*, **4**(1), pp. 1-28.

SCHEEPENS, R., WILLEMS, N., VAN DE WETERING, H. and VAN WIJK, J.J., 2011Interactive visualization of multivariate trajectory data with density maps, 2011, pp. 147-154.

SEVILLA, J., Mar 15, 2022-last update, Repositorio Github Ontologia STEVO. Available: <https://github.com/javier-sevilla/STEVO> [Mar 15, 2022].

SEVILLA, J., CASANOVA, P., Jun 12, 2021-last update, Evaluación stress mapa espacio temporal. Available: [https://silknow.eu/silknow/STMAPS\\_Evaluation](https://silknow.eu/silknow/STMAPS_Evaluation) [Feb 12, 2022].

- SEVILLA, J., SAMPER, J.J., HERRERA, G. and FERNÁNDEZ, M., 2018. SMART-ASD, model and ontology definition: a technology recommendation system for people with autism and/or intellectual disabilities. *Int.J.Metadata Semant.Ontologies*, **13**(2), pp. 166-178.
- SHAW, R., RABINOWITZ, A., GOLDEN, P. and KANSA, E., 2016. A sharing-oriented design strategy for networked knowledge organization systems. *International Journal on Digital Libraries*, **17**(1), pp. 49-61.
- SHI, L., LI, S., YANG, X., QI, J., PAN, G. and ZHOU, B., 2017. Semantic Health Knowledge Graph: Semantic Integration of Heterogeneous Medical Knowledge and Services. *BioMed Research International*, **2017**, pp. 2858423.
- SHI, Z. and PUN-CHENG, L., 2019. *Spatiotemporal Data Clustering: A Survey of Methods*.
- SILKNOW CONSORTIUM, 2022-last update, ADASILK. Available: <https://ada.silknow.org/> [Mar 15, 2022].
- SILVA, I.C.S., SANTUCCI, G. and FREITAS, CARLA MARIA DAL SASSO, 2019. Visualization and analysis of schema and instances of ontologies for improving user tasks and knowledge discovery. *Journal of Computer Languages*, **51**, pp. 28-47.
- SIMSEK, U., ANGELE, K., KÄRLE, E., PANASIUK, O. and FENSEL, D., 2019. A formal approach for customization of schema.org based on SHACL. *undefined*, .
- SINGHAL, A., May, 2012-last update, Introducing the knowledge graph: things, not strings. . Available: <https://blog.google/products/search/introducing-knowledge-graph-things-not/> [04/08/, 2021].
- SIROHI, P., AGARWAL, A. and MAHESHWARI, P., 2020A survey on Augmented Virtual Reality: Applications and Future Directions, 2020, pp. 99-106.
- SLATER J.A., M.S., 1998. WGS 84 — Past, Present and Future. . *Advances in Positioning and Reference Frames*, **118**.
- SOBRAL, T., GALVÃO, T. and BORGES, J., 2016VUMO: Towards an ontology of urban mobility events for supporting semi-automatic visualization tools, 2016, pp. 1700-1705.
- SOBRAL, T., GALVÃO, T. and BORGES, J., 2020. An Ontology-based approach to Knowledge-assisted Integration and Visualization of Urban Mobility Data. *Expert Systems with Applications*, **150**, pp. 113260.
- SPAHIU, BLERINA AND MAURINO, ANDREA AND PALMONARI, MATTEO, 2018Towards Improving the Quality of Knowledge Graphs with Data-driven Ontology Patterns and SHACL, *ISWC 2018*, 2018 2018, pp. 103,117.
- STADLER, J., 2021-last update, Australia Cultural Atlas. Available: <http://australian-cultural-atlas.info/CAA/search.php> [08/25/, 2021].
- STANFORD UNIVERSITY, 2022-last update, Protégé Tool. Available: <https://protege.stanford.edu/> [Apr 10, 2022].

- STARDOG UNION, 2014-last update, StarDog Web. Available: <http://www.stardog.com> [Nov 12, 2021].
- SUCHANEK, F.M., KASNECI, G. and WEIKUM, G., 2007Yago: A core of semantic knowledge, 2007, pp. 697-706.
- SUN MICROSYSTEMS, 2000. *The Java 3D API Specification*. Oracle.
- SURE, Y. and STUDER, R., 2002. *A Methodology for Ontology-Based Knowledge Management*.
- TABER, K.S., 2018. The Use of Cronbach's Alpha When Developing and Reporting Research Instruments in Science Education. *Research in Science Education*, **48**(6), pp. 1273-1296.
- TCHOUANGOUEM, J.F., PAUWELS, P., FONBEYIN, H.A., MAGNIONT, C., KARRAY, M. and KAMSU-FOGUEM, B., 2019. Integration of environmental data in BIM tool & linked building data. *undefined*, .
- THE KHRONOS GROUP INC., 2021-last update, WebGL Home Page. Available: <https://www.khronos.org/webgl/> [08/28/, 2021].
- THE METROPOLITAN, 2020-last update, The Met Collection. Available: <https://www.metmuseum.org/art/collection> [08/09/, 2021].
- THE TRUSTEES OF THE BRITISH MUSEUM, 2021-last update, The British Museum. Available: <https://www.britishmuseum.org/> [08/22/, 2021].
- TOMISLAV HENGL, PIERRE ROUDIER, DYLAN BEAUDETTE and EDZER PEBESMA, 2015. plotKML: Scientific Visualization of Spatio-Temporal Data. *Journal of Statistical Software*, **63**(1), pp. 1-25.
- TONTI, G., BRADSHAW, J.M., JEFFERS, R., MONTANARI, R., SURI, N. and USZOK, A., 2003Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder, D. FENSEL, K. SYCARA and J. MYLOPOULOS, eds. In: *The Semantic Web - ISWC 20, 03 2003*, Springer Berlin Heidelberg, pp. 419-437.
- TRENT, R. and LIBRARY OF CONGRESS, 07/23/, 2020-last update, Github. LC Congress Python API. Available: <https://github.com/LibraryOfCongress/data-exploration/tree/master/maps> [08/24/, 2021].
- TRIPADVISOR, 2021-last update, Tripadvisor. Available: <https://www.tripadvisor.es/> [08/28/, 2021].
- TUDHOPE, D., BINDING, C., 2013-last update, Senaschal project. Available: <https://www.heritagedata.org/blog/about-heritage-data/senaschal/> [08/12/, 2021].
- U. VALÈNCIA, 2022-last update, Arxiu Valencia de Disseny. Available: <https://www.uv.es/archivo-valenciano-dise%C3%B1o/es/archivo-valenciano-del-diseno.html> [Jan 10, 2022].
- U.VALENCIA, 2021-last update, Semap Project. Available: <https://www.uv.es/semap/> [08/30/, 2021].

UNESCO, 2014. Heritage sustainability. In: UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION, ed, *UNESCO. CULTURE FOR DEVELOPMENT INDICATORS*. Paris: UNESCO, pp. 131.

UNESCO, 1977-last update, The UNESCO Thesaurus  
. Available: <http://vocabularies.unesco.org> [08/12/, 2021].

UNITY TECH., 2022a-last update, Asset Store. Available: <https://assetstore.unity.com/> [Jan 31, 2022].

UNITY TECH., 2022b-last update, Unity 3D. Available: <https://unity3d.com/> [Jan 31, 2022].

UNITY TECHNOLOGIES., 2022-last update, WebGL:Interacting with browser scripting. (Unity Documentation). Available: <https://docs.unity3d.com/Manual/webgl-interactingwithbrowserscripting.html> [Feb 18, 2022].

USCHOLD, M., KING, M., MORALEE, S. and ZORGIO, Y., 1998. The Enterprise Ontology. *The Knowledge Engineering Review*, **13**(1), pp. 31-89.

VAN RUYMBEKE, M., HALLOT, P., NYS, G. and BILLEN, R., 2018. Implementation of multiple interpretation data model concepts in CIDOC CRM and compatible models. *Virtual Archaeology Review*, **ol 9, No 19** (2018).

VECCO, M., 2010. A definition of cultural heritage: From the tangible to the intangible. *Journal of cultural heritage*, **16**(3), pp. 239.

VERHODUBS, O., 2017. Realization of Ontology Web Search Engine. *arXiv preprint arXiv:1702.06934*, .

VICTORIAN ALBERT MUSEUM, 2021-last update, V&A Collections  
. Available: <https://www.vam.ac.uk/collections> [04/04/, 2021].

W3C, Mar 26, 2020-last update, Time Ontology in OWL. Available: <https://www.w3.org/TR/owl-time/> [Nov 15, 2021].

W3C, 2019. *Comparison of RDFJS Libraries*.

W3C, Jul 20, 2017-last update, Shapes Constraint Language (SHACL). Available: <https://www.w3.org/TR/shacl/> [08/30/, 2021].

W3C, 2015-last update, W3C Wiki StarDog. Available: <https://www.w3.org/2001/sw/wiki/Stardog> [Nov, 10, 2021].

W3C, Apr 5, 2012-last update, W3C XML Schema Definition Language (XSD) 1.1. Available: <https://www.w3.org/TR/xmlschema11-1/> [08/31/, 2021].

W3C, 2008-last update, SPARQL Query Language for RDF. Available: <https://www.w3.org/TR/rdf-sparql-query/> [Nov 10, 2021].

W3C, Aug 16, 2006-last update, Extensible Markup Language (XML) 1.1 (Second Edition). Available: <https://www.w3.org/TR/2006/REC-xml11-20060816/> [08/31/, 2021].

- W3C, Feb 10, 2004a-last update, OWL Web Ontology Language. Available: <https://www.w3.org/TR/2004/REC-owl-features-20040210/> [08/30/, 2021].
- W3C, Feb 10, 2004b-last update, RDF Vocabulary Description Language 1.0: RDF Schema. Available: <https://www.w3.org/TR/2004/REC-rdf-schema-20040210/> [09/02/, 2021].
- W3C, Feb 10, 2004c-last update, RDF/XML Syntax Specification (Revised). Available: <https://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/> [09/02/, 2021].
- W3C, Feb 10, 2004d-last update, Resource Description Framework (RDF): Concepts and Abstract Syntax. Available: <https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/> [09/02/, 2021].
- W3C, 2003-last update, RDQL - A Query Language for RDF. Available: <https://www.w3.org/Submission/RDQL/> [Nov 10, 2021].
- W3C, Mar 27, 2009-last update, OWL 2 Web Ontology Language. Available: <https://www.w3.org/TR/2009/WD-owl2-overview-20090327/> [08/30/, 2021].
- W3C, 1998-last update, W3C XML Specification DTD 2.0. Available: <https://www.w3.org/XML/1998/06/xmlspec-report-v20.htm> [08/31/, 2021].
- WANG, C., MA, X. and CHEN, J., 2018a. Ontology-driven data integration and visualization for exploring regional geologic time and paleontological information. *Computers & Geosciences*, **115**, pp. 12-19.
- WANG, C., MA, X. and CHEN, J., 2018b. Ontology-driven data integration and visualization for exploring regional geologic time and paleontological information. *Computers and Geosciences*, **115**, pp. 12-19.
- WANG, D., GUO, D. and ZHANG, H., 2017. Spatial Temporal Data Visualization In Emergency Management: A view from data-driven decision. *EM-GIS'17: Proceedings of the 3rd ACM SIGSPATIAL Workshop on Emergency Management using*. New York, NY, USA: Association for Computing Machinery, pp. 1-7.
- WANG, Q., MAO, Z., WANG, B. and GUO, L., 2017. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering*, **29**(12), pp. 2724-2743.
- WEB3D CONSORTIUM, 2013-last update, X3D Specifications - ISO/IEC-19775-1. Available: <https://www.web3d.org/standards/number/19775-1> [09/20/, 2021].
- WENG, J., QIU, J. and GAO, Y., 2020. Ontology Language XOL Used for Cross-Application Communication. *Ontological Analyses in Science, Technology and Informatics*. IntechOpen, .
- WICK, M., 2021-last update, Geonames. Available: <https://www.geonames.org/> [30/08/, 2021].
- WILIAMS, J., 2012. *Learning HTML5 game programming : a hands-on guide to building online games using Canvas, SVG, and WebGL*. Upper Saddle River, NJ : Addison-Wesley.



- WINDHAGER F., MAYR E., SCHREDER, G., SMUC M., FEDERICO, P. AND MIKSCH, S., 2016 Reframing Cultural Heritage Collections in a Visualization Framework of Space-Time Cubes, *The 3rd Histoinformatics Workshop*, 07/11/2016 2016, CEUR Workshop Proceedings ISSN: 1616-0073.
- WINDHAGER, F. AND FILIPOV, V. A. AND SALISU, S. AND MAYR, E., June 2018 Visualizing Uncertainty in Cultural Heritage Collections, *EuroVis Workshop on Reproducibility, Verification, and Validation in Visualization* June 2018, Eurographics Association, pp. 7-11.
- WINDHAGER, F., FEDERICO, P., SCHREDER, G., GLINKA, K., DÖRK, M., MIKSCH, S. and MAYR, E., 2019. Visualization of Cultural Heritage Collection Data: State of the Art and Future Challenges. *IEEE Transactions on Visualization and Computer Graphics*, **25**(6), pp. 2311-2330.
- WINDHAGER, F., FILIPOV, V.A., SALISU, S. and MAYR, E., 2018. *Visualizing Uncertainty in Cultural Heritage Collections*. The Eurographics Association.
- WLEKLINSKI., F. AND TOLLE, K., 2005-last update, easy RDF Query Language . Available: <https://www.yumpu.com/en/document/read/34305500/erql-tutorial-goethe-universitat2021> [2021].
- WYLOT, M., HAUSWIRTH, M., CUDRÉ-MAUROUX, P. and SAKR, S., 2018. RDF Data Storage and Query Processing Schemes: A Survey. *ACM Computing Surveys*, **51**(4), pp. 1-36.
- XIAOANG, M., 2020-last update, Github. Paleontology Information Explorer Open Source. Available: <https://github.com/xgmachina/geotimeNam> [08/25/, 2021].
- XIAOGANG, M., 2021-last update, Github. Paleontology Information Explorer Ontology. Available: <https://github.com/xgmachina/geotimeversion> [08/25/, 2021].
- YOU, E., 2022-last update, Vue.js Project. Available: <https://vuejs.org> [Jan 18, 2022].
- ZERBINI, A., 2018. Developing a Heritage Database for the Middle East and North Africa. *null*, **43**, pp. S9-S18.
- ZERDOUMI, S., SABRI, A.Q.M., KAMSIN, A., HASHEM, I.A.T., GANI, A., HAKAK, S., AL-GARADI, M. and CHANG, V., 2018. Image pattern recognition in big data: taxonomy and open challenges: survey. *Multimedia Tools and Applications*, **77**(8), pp. 10091-10121.
- ZHAO, Y. and FAN, B., 2018. Exploring open government data capacity of government agency: Based on the resource-based theory. *Government Information Quarterly; Internet Plus Government: Advancement of Networking Technology and Evolution of the Public Sector*, **35**(1), pp. 1-12.
- ZHENG, K., GU, D., FANG, F., WANG, Y., LIU, H., ZHAO, W., ZHANG, M., AND LI, Q., 2017. VISUALIZATION OF SPATIO-TEMPORAL RELATIONS IN MOVEMENT EVENT USING MULTI-VIEW. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, **XLII-2/W7**, , pp. 1469-1476.
- ZHONG, C., WANG, T., ZENG, W. and MÜLLER, S., 2012. *Spatiotemporal Visualisation: A Survey and Outlook*. Springer Berlin Heidelberg.



ZHOU, C., WANG, H., WANG, C., HOU, Z., ZHENG, Z., SHEN, S., CHENG, Q., FENG, Z., WANG, X., LV, H., FAN, J., HU, X., HOU, M. and ZHU, Y., 2021. Geoscience knowledge graph in the big data era. *Science China Earth Sciences*, **64**(7), pp. 1105-1114.

ZOU, X., 2020. A Survey on Application of Knowledge Graph. *Journal of Physics: Conference Series*, **1487**, pp. 012016.

ZUNINO, A., VELÁZQUEZ, G., CELEMÍN, J.P., MATEOS, C., HIRSCH, M. and RODRIGUEZ, J.M., 2020. *Evaluating the Performance of Three Popular Web Mapping Libraries: A Case Study Using Argentina's Life Quality Index*.

## PUBLICACIONES

### DERIVADAS DE LA TESIS

- En este artículo se expone la herramienta de visualización utilizada en el proyecto SILKNOW. También describe el hecho de que la visualización está especificada mediante una ontología. Sin embargo, está especialmente enfocado en describir los resultados de la herramienta y compararlo con otras herramientas similares que se han expuesto en esta tesis doctoral.

\* Sevilla, J.; Casanova-Salas, P.; Casas-Yrurzum, S.; Portalés, C. Multi-Purpose Ontology-Based Visualization of Spatio-Temporal Data: A Case Study on Silk Heritage. *Appl. Sci.* 2021, 11, 1636. <https://doi.org/10.3390/app11041636>

- Este artículo describe varios aspectos del estado del arte de visualización de datos espacio temporales. Expone que quedan varios retos por cubrir. Y con el fin de poder avanzar en estos aspectos se expone los aspectos principales del diseño del marco de trabajo diseñado y las bases de una propuesta de implementación. También se detallan varios aspectos de visualización que se han cubierto en el desarrollo de este trabajo, como es la visualización simultánea de datos en tiempo, y la visualización de relaciones con los objetos que se están representando en el mapa.

\* Sevilla, J., Portalés, C., Gimeno, J., Sebastián, J. (2019). SILKNOWViz: Spatio-Temporal Data Ontology Viewer". In: , *et al.* Computational Science – ICCS 2019. ICCS 2019. Lecture Notes in Computer Science(), vol 11540. Springer, Cham. [https://doi.org/10.1007/978-3-030-22750-0\\_8](https://doi.org/10.1007/978-3-030-22750-0_8)

- Los dos artículos siguientes describen con detalle los aspectos de visualización de un mapa espacio temporal usado en el proyecto Semap. Este proyecto permite visualizar y realizar búsquedas sobre más de 300.000 objetos de patrimonio cultural cuyos datos están soportados en un grafo de conocimiento basado en CIDOC-CRM. Aunque en este trabajo no se usaron los resultados software de esta tesis doctoral, debido al volumen de datos, si que se usaron las técnicas de visualización especificadas por el modelo, que también es un resultado de este trabajo. Algunos ejemplos son el uso de agrupaciones, líneas de tiempo y visualización de agrupaciones en escenas superpuestas.

El primer artículo está más centrado en la adaptación de los datos del dominio a y su adaptación a CIDOC-CRM. Con, así como las transformaciones necesarias y la definición del grafo de conocimiento.

\* Sevilla, J., Casanova, P., Samper, J. Javier, Portalés, C. SeMap: A project on semantisation and visualisation of cultural heritage data. *EATIS 2022 Proceedings*. Aveiro (Portugal) June 2022.

El Segundo artículo está más orientado a la representación gráfica de los datos, así como su evaluación en un entorno educativo.

\* Portalés C, Casanova-Salas P, Sevilla J, Sebastián J, León A, Samper JJ. Increasing Access to Cultural Heritage Objects from Multiple Museums through Semantically-Aware Maps. *ISPRS International Journal of Geo-Information*. 2022; 11(4):266. <https://doi.org/10.3390/ijgi11040266>

RELACIONADAS

Gaitán M, Portalés C, Sevilla J, Alba E. Applying Axial Symmetries to Historical Silk Fabrics: SILKNOW's Virtual Loom. *Symmetry*. 2020; 12(5):742. <https://doi.org/10.3390/sym12050742>

Gaitán M, Alba E, León A, Pérez M, Sevilla J, Portalés C. Towards the Preservation and Dissemination of Historical Silk Weaving Techniques in the Digital Era. *Heritage*. 2019; 2(3):1892-1911. <https://doi.org/10.3390/heritage2030115>

Portalés C, Sebastián J, Alba E, Sevilla J, Gaitán M, Ruiz P, Fernández M. Interactive Tools for the Preservation, Dissemination and Study of Silk Heritage—An Introduction to the SILKNOW Project. *Multimodal Technologies and Interaction*. 2018; 2(2):28. <https://doi.org/10.3390/mti2020028>