

Modelado específico de procedimientos en el dominio de la Administración Electrónica

Guillermo Infante Hernández ¹, Benjamín López Pérez ¹, Aquilino Adolfo Juan Fuente ¹

infante@innova.uniovi.es, benja@uniovi.es, aajuan@uniovi.es

¹ Departamento de Informática, Universidad de Oviedo, 33007, Oviedo, España.

DOI: 10.4304/risti.9.1-14

Resumen: Debido a la complejidad que reviste el dominio de la Administración Electrónica (AE), las desarrolladas deberán ser capaces de integrarse con la multitud de plataformas y sistemas que componen dicho dominio. El impacto de esta integración además mejorar la tramitación de procedimientos. En el presente trabajo se propone un lenguaje de modelado gráfico (DSL) para la tramitación de procedimientos en el dominio específico de la AE. Este lenguaje parte de la definición de un metamodelo de AE que identifica sus principales entidades y define sus relaciones. Además se construyó una herramienta gráfica de modelado la cual fue probada con dos casos de estudio reales. aplicaciones

Palabras clave: Administración Electrónica; Metamodelo; DSL

Abstract: Due to the complexity involved in E-Government domain applications, the developed ones should be able to interoperate with the growing number of emerging management platforms. The impact of this integration has among its main goals to improve the procedure management task, which suffers an important lack of technological proposals. This work proposes a graphical domain specific language (DSL) for procedure management in E-Government domain. This language came from the definition of a metamodel that states the principle E-Government elements and defines its relations. A further graphical modeling tool was developed to implement the DSL and was tested with two actual case studies.

Keywords: E-Government; Metamodel; DSL

1. Introducción

El desarrollo de la AE particularmente en España ha venido acompañado de la aprobación de una normativa que lo regula y garantiza su evolución. Dentro de esta normativa se destaca la Ley de Acceso Electrónico de los Ciudadanos a los Servicios Públicos (LAECSP, Ley 11/2007, de 22 de junio) que se ha visto reforzada por medidas aprobadas por la Unión Europea como la Directiva de Servicios (DS 2006/123/CE). Para la puesta en marcha de la Directiva de Servicios, se solicita la aplicación de

medidas concretas, como las ventanillas únicas para prestadores de servicios, la puesta en marcha de procedimientos por vía electrónica y la cooperación administrativa entre los estados miembros. En este sentido ha quedado regulada la forma en la que llevar la administración electrónica a los ciudadanos pero aún existe la necesidad de propuestas tecnológicas concretas de cómo hacerlo.

La principal idea que abarca este trabajo está relacionada precisamente con una propuesta tecnológica para el modelado de procedimientos administrativos en el contexto de la Administración Electrónica. Partiendo de esta idea se hace a continuación una relación de las principales contribuciones aportadas:

- Se definió un metamodelo que integra las entidades principales identificadas en el dominio de la AE, el cual se encuentra descrito en la Sección 2.1. Hasta el momento no se había encontrado un metamodelo que identificase estas entidades y definiese sus relaciones. Éste constituye un paso fundamental para lograr consenso en lo que a tramitación por medios electrónicos se refiere.
- Se construyó una herramienta gráfica de modelado basada en el metamodelo creado, la cual contribuye a la integración de plataformas distintas de tramitación en el contexto de la Administración Electrónica. Esta integración puede ser alcanzada mediante la traducción de los modelos creados a ficheros XMI de intercambio (Specification, A. A., & Group, O. M., 2003), los cuales pueden ser transformados en código fuente de cualquier lenguaje. La construcción de esta herramienta se describe entre las secciones 2.1 y 2.5.
- Se modelaron dos procedimientos reales con la herramienta creada como caso de estudio descrito en la Sección 3 y se generó un sitio web de forma automática a partir de la información recogida en el modelado de dichos procedimientos.

1.1. Deficiencias en la tramitación de procedimientos

Se han analizado varias aproximaciones de solución a la tramitación electrónica de procedimientos en el contexto de la AE, por ejemplo (Beynon-Davies, P. 2007) define un metamodelo para la AE desde el punto de vista socio-tecnológico. Propone además una serie de modelos de negocio que podrían ser utilizados para el desarrollo de la AE. Este trabajo aunque ve la necesidad de modelar el dominio de la AE no propone soluciones concretas de cómo hacerlo. Por otro lado no utiliza tecnologías de modelado de procesos y carece de una propuesta de solución tecnológica para afrontar este problema.

Otro de los estudios revisados fue el de (Becker, J., Pfeiffer, D., & Räckers, M. ,2007). Este trabajo va un poco más lejos y ya introduce la propuesta de un método para el modelado de procesos de dominio específico en la AP. El enfoque de modelado de dominio específico se ha considerado en este trabajo como la solución al problema de la reorganización de las instituciones. Se ha creado un método que aplica el vocabulario del dominio de la AP para capturar de forma eficiente el mapa de procesos de una organización. Resulta interesante el enfoque de este trabajo ya que introduce el modelado de dominio específico y define un vocabulario para ese dominio. En este caso el objetivo del trabajo se centra en la reorganización administrativa y no en soluciones de tramitación electrónica.

Finalmente se analizó el Proyecto W@anda (2004). Este proyecto ha sido llevado a cabo por la Junta de Andalucía para la tramitación de expedientes de forma electrónica. Basa su solución en un tramitador electrónico y la definición de un dominio semántico. Para identificar las entidades principales se ha basado en los conceptos recogidos en Specification, W. M. (1999) y trata la tramitación de expedientes como una máquina de estados. Este trabajo aunque define un dominio común para la tramitación de expedientes, basa su solución en la aplicación de flujos de trabajo (workflows). Este enfoque formaliza la gestión de actividades y responsabilidades asignadas a cada tarea así como las transiciones entre las mismas, sin embargo restringe la tramitación a una tecnología concreta, lo cual dificulta la estandarización de este proceso.

No obstante, aunque la prestación de servicios a los ciudadanos con el uso de medios electrónicos ha avanzado notablemente, hecho que se ve reflejado en los enfoques analizados entre otros, aún quedan necesidades importantes por cubrir. Basándose en los estudios publicados por la Fundación Orange y La junta de Andalucía, eEspaña (2010) y el Proyecto W@nda (2004) respectivamente, se pueden identificar algunas de estas necesidades. Del total de requisitos posibles, en este trabajo se abarcan los siguientes:

- Debe existir una forma de tramitar común. Procedimientos similares se tramitarán de igual forma en diferentes organizaciones.
- Se debe simplificar la comunicación entre los entornos de tramitación existentes con la utilización de estándares y hacerlos independientes de los Sistemas de Información de los que proceden.

2. DSL para el modelado de procedimientos en AE

Analizados los requisitos e investigadas las propuestas tecnológicas de como satisfacerlos, se utiliza la ingeniería dirigida por modelos (MDE), la cual constituye una solución eficiente para esta tarea. En esta sección se profundiza en estas técnicas y especialmente en la propuesta de arquitectura dirigida por modelos MDA (Booch, G. et al, 2003) para el desarrollo de un lenguaje de dominio específico (DSL) para la tramitación de procedimientos electrónicos.

La elección del desarrollo de un DSL para el modelado de procedimientos en AE viene dado por la propia naturaleza de este dominio. El dominio de la AE está compuesto por un gran número de entidades relacionadas entre sí e interpretadas de forma distinta en diferentes escenarios. La manera en que se realiza la tramitación electrónica resulta una tarea compleja de implementar debido a la normativa que regula su funcionamiento y las variaciones que experimenta de una AP a otra. Es en este contexto surgen problemas de incompatibilidad e interoperabilidad entre las plataformas que componen la AE. A pesar de que la fase inicial del desarrollo de un DSL resulta un proceso costoso, a largo plazo reduce los costes del ciclo de vida de desarrollo debido al incremento de la productividad (Christensen, 2003), figura 1. Esto significa que si se contase con una solución que normalizara en alguna medida el desarrollo de aplicaciones para este dominio, la productividad en el desarrollo de las mismas crecería considerablemente.

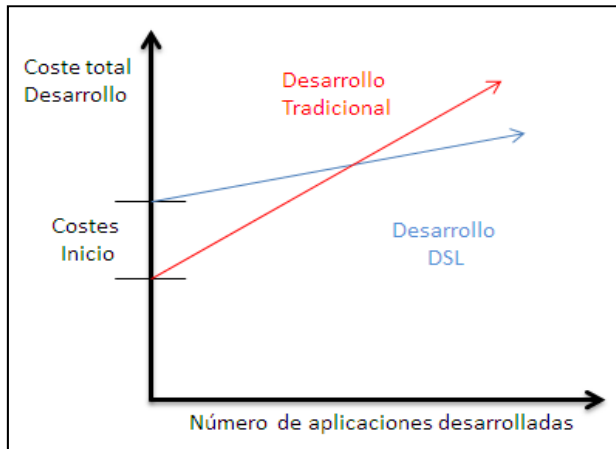


Figura 1 – Rentabilidad del Desarrollo DSL (Christensen, 2003)

El dominio de la AE como se ha mencionado está compuesto por un gran número de entidades. Para sacar verdadero provecho de un DSL, este debe ser diseñado como un lenguaje limitado y estrechamente enfocado a un problema en particular (Fowler, 2006). Por esta razón se han escogido las entidades más significativas que forman parte del proceso de tramitación electrónica para establecer un alcance razonable en el desarrollo de esta propuesta. En esta implementación se escogieron los entornos de GMF y EMF (Gronback, R. C., 2009) para la creación de una herramienta de modelado de procedimientos basada en un DSL. Esta elección está basada en la comparativa de (Pelechano et al, 2006) donde establece un análisis basado en criterios como: *el metamodelado, repositorio, transformaciones entre modelos y validaciones*. La tabla 1 muestra las entidades que forman el núcleo de la tramitación de procedimientos.

Se decidió que esta herramienta debería estar implementada en un entorno gráfico para permitir tanto a personal administrativo como técnico hacer sus desarrollos o modelados sin necesidad de un conocimiento técnico previo.

GMF establece un proceso muy específico para construir esta herramienta. Éste se compone de los siguientes subprocesos: **1.** definición del metamodelo, **2.** generación de código del modelo, **3.** definición de la metáfora gráfica, **4.** definición de las herramientas del modelo, **5.** especificación de la correspondencia entre los elementos del modelo y la metáfora gráfica, y **6.** generación del código de la herramienta. Estos subprocesos se detallan a continuación y se muestran en la figura 2.

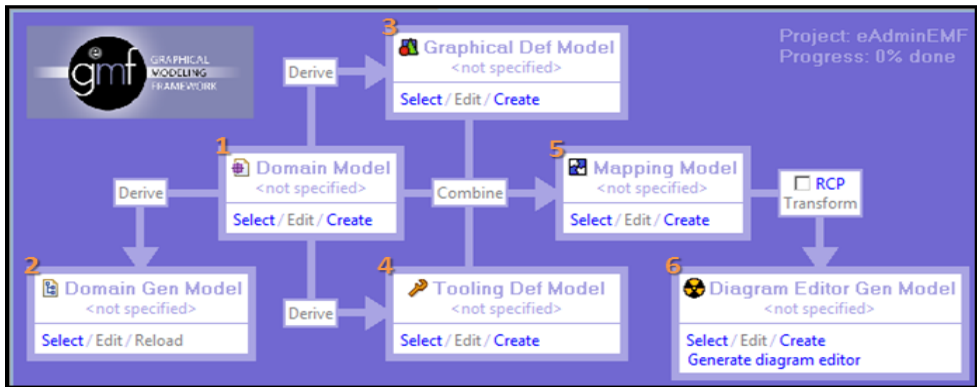


Figura 2 – Construcción de la herramienta DSL con EMF.

2.1. Definición del metamodelo

El metamodelo creado se basa en la definición del dominio específico de la Administración Electrónica definido en la tabla 1.

Tabla 1. Descripción de elementos principales del modelo.

Entidad	Descripción de Dominio
<i>Servicio</i>	"Cualquier actividad realizada por la Administración Pública dirigida a los ciudadanos para satisfacer sus necesidades, derechos u obligaciones".
<i>Procedimiento</i>	"Proceso de toma de decisiones en la Administración Pública".
<i>Trámite</i>	"Secuencia ordenada de tareas o actuaciones que representa la unidad básica de gestión dentro de un procedimiento.
<i>Interesado</i>	"Usuario participante en el procedimiento en el momento que forme parte activa en la tramitación del mismo".
<i>Expediente</i>	"Conjunto de documentos que integran un procedimiento administrativo".

GMF precisa de un metamodelo y para ello se sirve de EMF, el cual constituye un marco de modelado que soporta y genera documentos XMI con la especificación del dominio. El área de trabajo o modelado de EMF muestra parte del metamodelo creado en la figura 3.

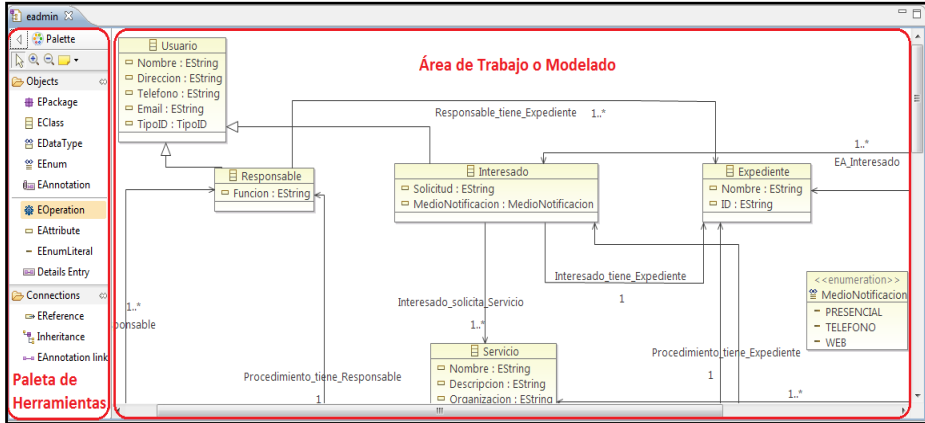


Figura 3 – Área de trabajo y paleta de herramientas EMF.

El árbol resultante con la totalidad de las entidades modeladas se muestra en la figura 4.

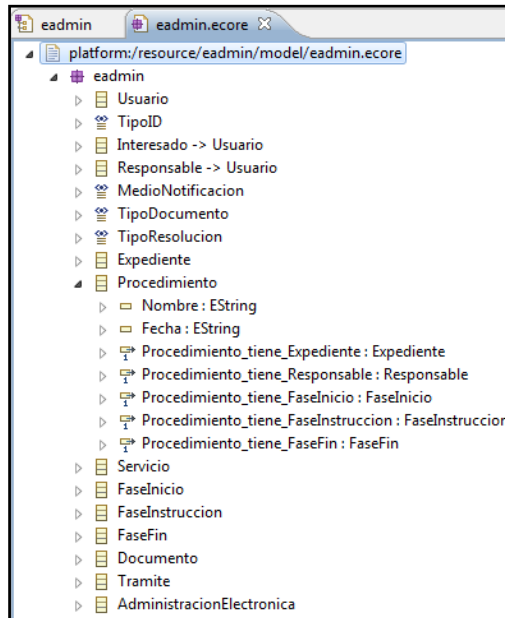


Figura 4 – Árbol del Metamodelo de Administración Electrónica.

2.2. Generación del código del modelo

El código del modelo se genera aplicando patrones de transformación. El resultado de la generación es un conjunto de clases e interfaces Java, que serán utilizadas más adelante en el proceso de creación de la herramienta de modelado específico de dominio. El objetivo de esta transformación es que todos los elementos que constituyen la herramienta se comporten tal y como el metamodelo creado lo establece. El código generado se muestra en la figura 5.

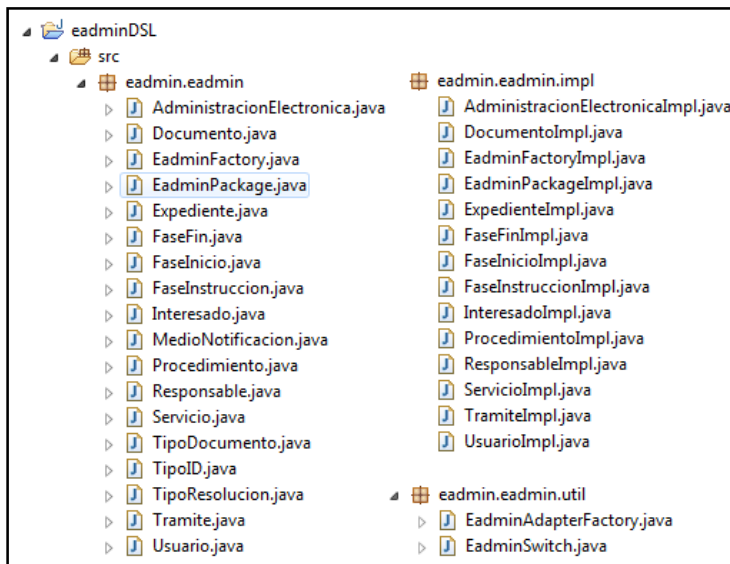


Figura 5 – Código Generado para el metamodelo de AE.

2.3. Definición de la metáfora gráfica y herramientas

La definición gráfica consiste en decidir qué primitivas de modelado harán la función de nodos (elementos de la herramienta de modelado que se desea construir), cuáles serán conectores (enlaces entre los nodos de la herramienta de modelado) y cuáles etiquetas (propiedades de los nodos y enlaces de la herramienta de modelado). En este paso además se define el panel de herramientas y el aspecto que tendrán todos los elementos gráficos del modelo tales como la iconografía de la paleta donde se representa cada nodo, enlace y propiedad.

2.4. Correspondencia entre los elementos del modelo y la metáfora gráfica

En este escenario todo lo creado anteriormente cobra sentido ya que se relacionan y asocian todos y cada uno de los elementos creados con anterioridad. El objetivo fundamental de este paso es el de asegurar que todas las anteriores especificaciones se han realizado correctamente y mantienen coherencia.

2.5. Generación del código de la herramienta

Una vez terminado el subproceso de mapeo, se completa el proceso de definición de la herramienta de modelado gráfico de procedimientos. El último paso para poder ejecutar la herramienta es la generación de su código, el cual se hace de forma totalmente automática. Una vez generado el código es posible ejecutar la herramienta como un plug-in de Eclipse¹. La figura 6 muestra la interfaz de la herramienta creada.

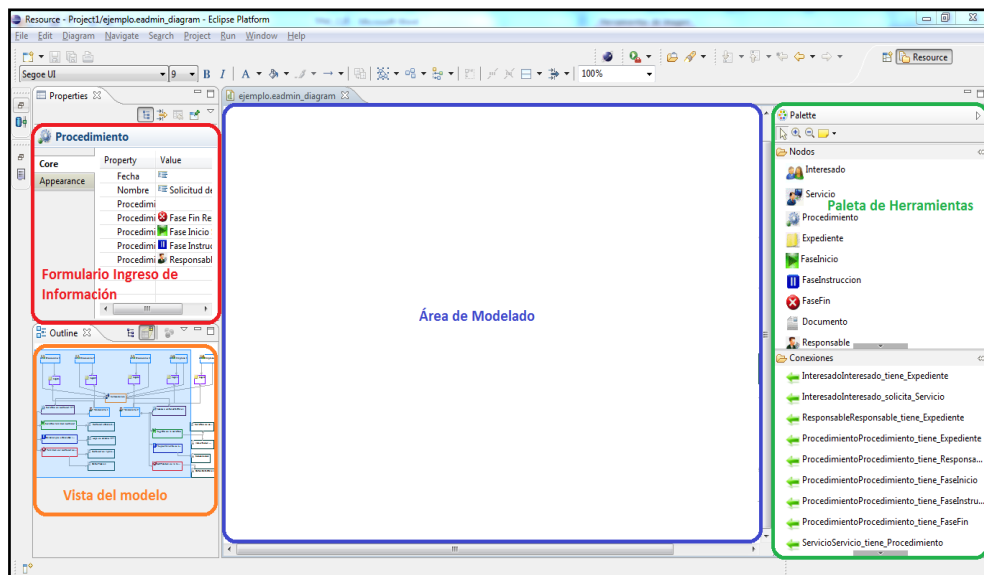


Figura 6 – Herramienta de modelado gráfico.

3. Caso de Estudio

En este caso en particular se modelaron dos procedimientos ofrecidos a través de los servicios del Ministerio de Hacienda (MEH²). La implementación de estos procedimientos no va más allá de los propósitos demostrativos que se persiguen para validar la herramienta con casos reales. El primero de ellos es el procedimiento de Solicitud de borrador de IRPF, el cual persigue el objetivo de facilitar la confección y presentación de la Declaración del Impuesto sobre la Renta de las Personas Físicas y cuyo órgano responsable es la Agencia Estatal de Administración Tributaria. El segundo de los procedimientos modelados fue el de Acceso a datos catastrales cuyo objetivo es facilitar la consulta y certificación de datos catastrales y la obtención de copias de documentos, su órgano responsable es la Dirección General del Catastro.

¹ Entorno de Desarrollo (IDE). <http://www.eclipse.org/>

² Ministerio de Economía y Hacienda. <http://www.meh.es/>

La descripción detallada de ambos procedimientos puede consultarse en el MEH. El modelado definido para ambos procedimientos con la herramienta creada puede visualizarse con más detalle en la figura 7.

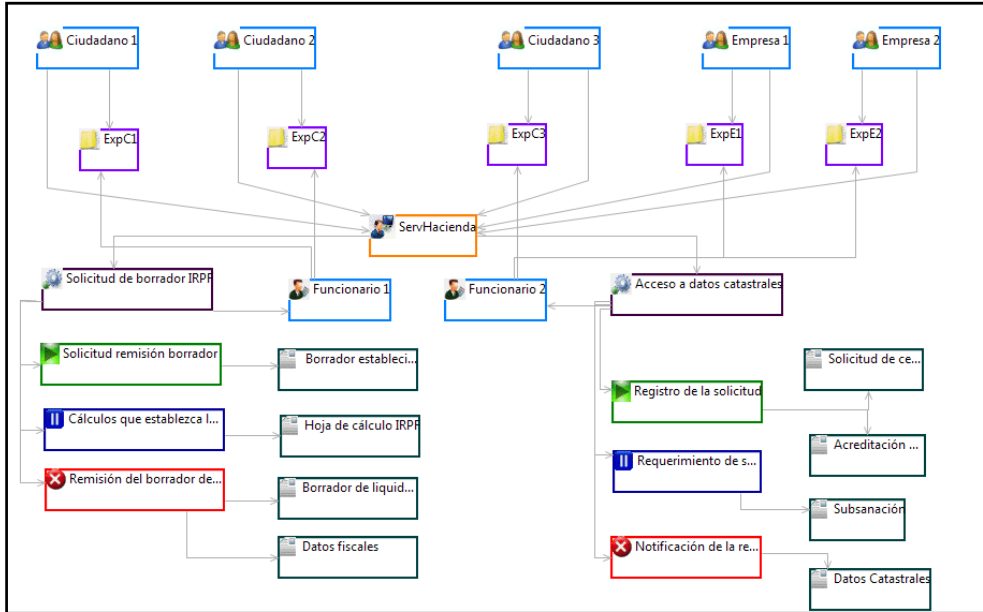


Figura 7 – Caso de estudio modelado con la herramienta creada.

El modelo creado es recogido en el formato XMI. Esta información recoge íntegramente el modelo creado con la herramienta y permite su posterior transformación a texto aplicando plantillas en lenguaje MOFScript (Oldevik et al., 2005), mediante las cuales es posible generar código a cualquier lenguaje de propósito general. Una vez transformado este fichero XMI al lenguaje que se desee, la aplicación resultante podrá ser desplegada en la plataforma escogida para su utilización.

Como se había planteado, este formato recoge toda la información del modelo creado: sus nodos, conexiones y datos introducidos. En el fragmento de la figura 8 se muestra el **Nodo** Procedimiento y los distintos componentes del modelo recogidos en XMI. En este caso el **Dato** que se muestra es el nombre del procedimiento en cuestión; las **Conexiones** representan los enlaces de este nodo a otros nodos o entidades previamente definidos en el metamodelo y las **Referencias** indican hacia donde apunta cada conexión, o sea el nodo destino de la conexión.

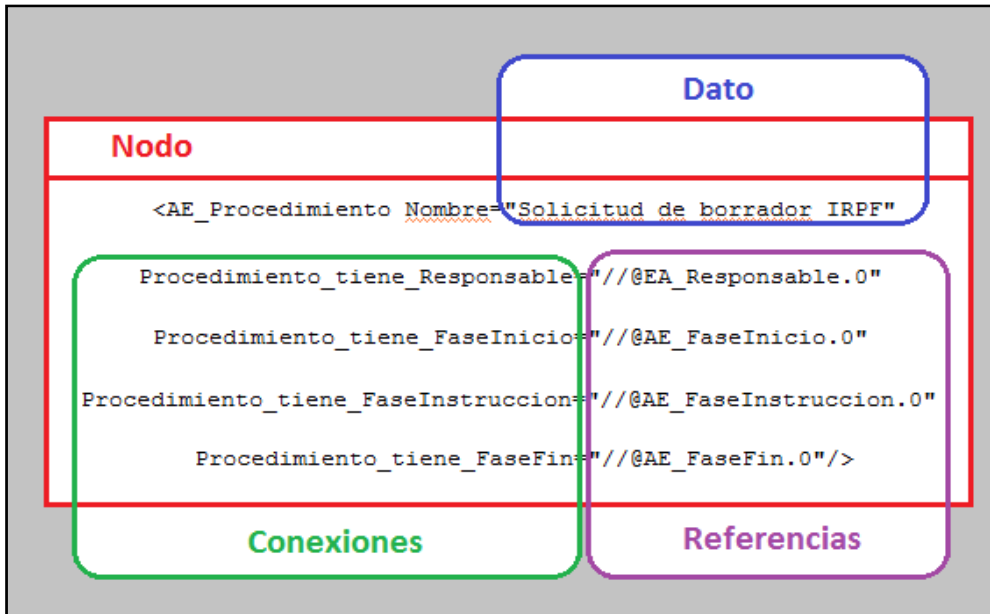


Figura 8 – Detalle XMI del modelo del caso de estudio.

4.1. Transformación y despliegue del modelo específico creado.

Una vez completado el modelado específico es posible transformar éste a código fuente de una aplicación web que muestre la información recogida en el modelo así como cualquier lógica que se desee implementar con el código generado. Esta transformación es llevada a cabo mediante el uso del lenguaje MOFScript. Se transforma el fichero XMI generado por la herramienta gráfica, el cual contiene un esquema de todos los datos contenidos en el modelo creado. La figura 9 muestra un fragmento del código creado con MOFScript. Se recorrieron todos los nodos del modelo creado de forma similar a un grafo dirigido y se fue extrayendo la información contenida en cada nodo.

```
EadminTransformation.m2t
18 /*****Interesado*****/
19 //Se crea la pagina de interesado
20 file fInteresado ("interesado.html");
21 fInteresado.println(header());
22 fInteresado.println(interésadoMiddle());
23
24 //Se verifica si existe algun nodo Interesado
25 if (self.EA_Interesado != null){
26
27 //Recorre todos los nodos que estan relacionados con interesado
28 self.EA_Interesado->forEach(nodo:mAE.Interesado){
29     fInteresado.println('<tr>');
30     fInteresado.print('<td>');
31     fInteresado.println(nodo.Nombre);
32     fInteresado.print('</td>');
33     fInteresado.print('<td>');
34     nodo.Interesado_solicita_Servicio->forEach(serv:mAE.Servicio){
35         fInteresado.println(serv.Nombre+'<br/>');
36     }
37     fInteresado.print('</td>');
38     fInteresado.print('<td>');
39     fInteresado.print(nodo.Interesado_tiene_Expediente.Nombre);
40     fInteresado.print('</td>');
41     fInteresado.println('</tr>');
42 }
43 }
```

Figura 9 – Fragmento de la plantilla MOFScript creada.

Se ha generado posteriormente un sitio web automáticamente a partir de la plantilla MOFScript creada. Todo el código HTML así como las hojas de estilo fueron generados por la herramienta. En esta etapa del trabajo se ha decidido implementar una representación tabulada de los datos obtenidos del modelo con el objetivo de demostrar la funcionalidad de la herramienta gráfica creada y el cierre del ciclo de vida de desarrollo del software mediante la aplicación de MDA. La figura 10 muestra una de las páginas del sitio generado.



Figura 10 – Página "Procedimientos" del sitio web generado.

5. Conclusiones

Se ha propuesto la integración de todos los conceptos identificados en la literatura para dar una visión más clara de cómo se relacionan los mismos en el contexto de la Administración Electrónica. Esto ha facilitado la creación de un metamodelo a partir de la definición del dominio de la misma.

A partir del metamodelo, se desarrolló un lenguaje de dominio específico para la Administración Electrónica donde se definió la semántica de dicho lenguaje en función del dominio y de las metáforas gráficas para su posterior implementación.

Se construyó una herramienta gráfica de modelado específico de procedimientos en el dominio de la Administración Electrónica.

Se modelaron dos casos de estudio reales generando a partir de los mismos un sitio web de forma automática con la información recogida en ambos modelos.

6. Trabajo Futuro

Toda investigación genera inquietudes y nuevas líneas por donde encaminar el trabajo futuro. A través del estudio de los conocimientos actuales en este campo y el análisis de los resultados obtenidos se pueden identificar algunas líneas futuras vinculadas con este tema. Una relación de estas líneas o posibles tareas a tener en cuenta para próximas intervenciones se propone a continuación.

- Realizar un estudio de arquitecturas existentes en la Administración Electrónica para identificar sus elementos comunes. Esto permitiría estandarizar los desarrollos y generar artefactos software con una ampliación de la herramienta creada para cada plataforma específica.
- Tomar una muestra de procedimientos de distintas administraciones y realizar su representación en la herramienta, contrastando posteriormente si se ha podido recoger toda la información de los distintos procedimientos en el modelo o si se detecta que existen carencias.

Referencias bibliográficas

- Beynon-Davies, P. (2007). Models for e-government. *Transforming Government People Process and Policy*, 1(1), 7-28. Retrieved from <http://www.emeraldinsight.com/10.1108/17506160710733670>
- Becker, J., Pfeiffer, D., & Räckers, M. (2007). Domain Specific Process Modelling in Public Administrations – The PICTURE-Approach. (M. A. Wimmer, J. Scholl, & A. Gronlund, Eds.) 6th International Conference EGOV 2007. Springer Berlin/Heidelberg. Obtenido de <http://www.springerlink.com/content/umhvo013w7581p49/?p=e5a056cd4bc14422912773fca8776eec&pi=6>.
- BOE. (2007). Ley 11/2007, de 22 de junio, para el acceso electrónico de los ciudadanos a los Servicios Públicos, Obtenido de http://www.boe.es/g/es/bases_datos/doc.php?coleccion=iberlex&id=2007/12352
- Booch, G., Brown, A. W., Iyengar, S., Rumbaugh, J., & Selic, B. (2004). An MDA Manifesto. *Business Process TrendsMDA Journal*. Obtenido de <http://www.citeulike.org/group/2440/article/1271715>
- Christensen, N.H. (2003). Domain-Specific Languages in Software Development and the relation to partial evaluation, PhD thesis, DIKU, Dept. of Computer Science, University of Copenhagen, Denmark.
- eEspaña (2010). Madrid: Fundación Orange France Telecom. Obtenido de http://fundacionorange.es/fundacionorange/analisis/eespana/e_espana10_10.html
- Fowler M. (2006). DSL Boundary, Obtenido de <http://martinfowler.com/bliki/DslBoundary.html>
- Gronback, R. C. (2009). Eclipse Modeling Project: A Domain-Specific Language (DSL) Toolkit. (E. Gamma, L. Nackman, & John Wiegand, Eds.) Eclipse Series (p. 736). Addison-Wesley Professional. Obtenido de <http://www.amazon.com/Eclipse-Modeling-Project-Domain-Specific-Language/dp/0321534077>
- Oldevik, J., Neple, T., Grønmo, R., Agedal, J., & Berre, A.-J. (2005). Toward Standardised Model to Text Transformations. In A. Hartman & D. Kreische (Eds.), *European Conference on Modeldriven Architecture Foundation and Application ECMDAFA*. Springer Berlin/Heidelberg. doi:10.1007/11581741_18

Pelechano V., Albert M., Muñoz J., and Cetina C., Building Tools for Model Driven Development Comparing Microsoft DSL Tools and Eclipse Modeling Plug-ins, In Proceedings of the 11th Conference on Software Engineering and Database (JISBD'06), Barcelona, España, 2006.

Specification, A. A., & Group, O. M. (2003). XML Metadata Interchange (XMI) Specification. Management, 01(May). Obtenido de <http://www.omg.org/spec/XMI/>

Specification, W. M. (1999). Workflow Management Coalition Terminology & Glossary (Document No. WFMC-TC-1011). Workflow Management Coalition Specification. Obtenido de <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/B002DJ1BEK>.

W@nda (2004). Proyecto W@nda: WORKFLOW EN LA ADMINISTRACIÓN ANDALUZA. Dominio Semántico. ISBN: 84-688-7845-6 Obtenido de <https://wso24.juntadeandalucia.es/pluton/adminelec/ArTec/wanda.jsp>.