

Asistencia Dirigida por Ontologías al Diseño Arquitectónico de Videojuegos

Facundo Gonzalez Gola¹, Alejandro Sánchez^{1,2}, Germán Montejano^{1,3}

¹Universidad Nacional de San Luis
Ejercito de Los Andes 950, San Luis, Argentina - Tel: +54 266 4520300
gonzalezgolafacundo@gmail.com, {asanchez,gmonte}@unsl.edu.ar

²Universidad Nacional de la Patagonia San Juan Bosco
Bvd Brown 2051 Puerto Madryn, Tel. +54-280-4883585

³Universidad Nacional de la Pampa
Av. Uruguay 151, (6300) Santa Rosa, La Pampa, Argentina, Tel.: +54-2954-245220 Int. 7125

Resumen

El diseño arquitectónico de videojuegos es complejo. Es normal que los videojuegos se proyecten para ser ejecutados sobre una amplia variedad de dispositivos, sistemas operativos, e interpretes. A su vez, siempre se utilizan componentes de software – llamados motores – que proveen un número de funcionalidades básicas, pero plantean limitaciones sobre la plataforma y el entorno de desarrollo. Esto se combina con requerimientos de performance, y en muchos casos de escalabilidad, naturales en este dominio. Por otro lado, los equipos de desarrollo son típicamente multidisciplinares, lo que dificulta la comunicación de decisiones de diseño.

Esta línea plantea la investigación y desarrollo de ontologías para dirigir el diseño arquitectónico de videojuegos. Estas ontologías explicitarán y alinearán las conceptualizaciones utilizadas por las disciplinas involucradas, facilitarán la integración de datos sobre los artefactos relacionados al diseño, y proveerán una teoría lógica para guiar parte del diseño

arquitectónico y explicar inconsistencias en el mismo.

Palabras clave: ontología, videojuegos, diseño arquitectónico.

Contexto

Esta línea de investigación, desarrollo e innovación se desprende del proyecto “Ingeniería de Software: conceptos, prácticas y herramientas para el desarrollo de software de calidad” de la facultad de Ciencias Físico, Matemáticas y Naturales de la Universidad Nacional de San Luis.

Introducción

El mercado mundial de videojuegos se encuentra en constante crecimiento. La ESA¹, asociación creadora del evento más importante a nivel mundial de video juegos (el Electronic Entertainment Expo o E3), reporta en (Essential facts about the computer and video game industry 2016) un gran crecimiento del mercado Norte Americano con respecto al año anterior. En 2016 el 63% de la población Norte Americana ha jugado al menos 3 horas o más por semana, significando un

¹ Entertainment Software Association

mercado de un total de 23,5 billones de dólares.

Sin embargo, muchos proyectos de desarrollo de videojuegos han fracasado y continúan haciéndolo en la actualidad. Considere por ejemplo el caso del juego ET, de la empresa Atari (Dingman 2014). El juego fue sacado de producción y llevó a la empresa a la bancarrota. Otro caso es el juego Duke Nukem Forever, que demoró 14 años en salir al mercado por problemas de diseño (Schiesel 2011). Peor aún es el caso del juego No Man's Sky, donde sus desarrolladores fallaron en cumplir con las características prometidas y debieron afrontar procesos legales (BBC-Tech 2016). Casos como estos se reportan frecuentemente en la actualidad, con mayor proporción en pequeñas y medianas empresas.

Es posible atribuir muchos de los fracasos en proyectos de desarrollo de videojuegos al diseño arquitectónico de los mismos. La complejidad en esta actividad viene dada principalmente por la gran variedad de plataformas sobre las que se pretenden liberar los juegos y sus limitaciones físicas, los componentes de software que se deben utilizar y sus condicionamientos sobre las plataformas, los típicos requerimientos de performance y escalabilidad, y la frecuente composición multidisciplinaria de los equipos de desarrollo.

El mercado demanda que los videojuegos sean liberados sobre una amplia gama de dispositivos que va desde consolas (computadoras específicamente diseñadas para juegos), pasando por computadoras hogareñas hasta teléfonos móviles. Estos dispositivos disponen de su software de base sobre el cual deben ejecutar los programas, tales como sistemas operativos y máquinas virtuales,

provistos bajo diferentes licencias, y disponibles en diferentes versiones. Tanto los dispositivos como su software evolucionan rápidamente.

Existen componentes de software – llamados motores de juegos – que pueden facilitar el desarrollo al proveer un número de funcionalidades tales como la renderización de imágenes, la emisión de sonidos, la captura de eventos generados por el usuario, la administración de memoria, o la ejecución de scripts bajo determinadas condiciones de disparo. Dichos motores son desarrollados para determinados dispositivos asumiendo la disponibilidad de ciertas capacidades y software de base.

A su vez, el desarrollo de videojuegos típicamente involucra expertos de una variedad de dominios, como ser artes gráficas, musicalización, y narración. Estos expertos cuentan con una formación muy diferente a los expertos IT y en consecuencia una conceptualización distinta para realizar el trabajo. La comunicación puede resultar poco eficaz y sea difícil inconsistencias en requerimientos y decisiones de diseño arquitectónicas.

Fallar en la detección de inconsistencias en el diseño arquitectónico de un videojuego pueden condenar al fracaso el proyecto. Por ejemplo, el juego requiere una cantidad de memoria mínima que algunas de las plataformas sobre las que se liberará no poseen. Otro ejemplo es utilizar para un juego de música una máquina virtual que permite la portabilidad del mismo, pero impide la ejecución simultánea de sonidos. Si el equipo no dispone de un arquitecto con suficiente conocimiento de la máquina virtual, este problema no se

detectaría hasta que el primer prototipo fuera probado.

Los casos como los mencionados pueden ser prevenidos si se dispusiera de un servicio que respondiera preguntas como las que siguen debajo.

- A. ¿Es consistente un conjunto dado de decisiones arquitectónicas?
- B. ¿Cuáles son limitaciones de memoria y procesamiento impuestas por las plataformas elegidas?
- C. ¿Qué componentes/software de base se ajusta a los requerimientos del video juego proyectado?

Existen antecedentes del uso de ontologías (Gruber 1993) para dirigir aplicaciones que respondan a preguntas como estas en el contexto del desarrollo de software. Una ontología es una representación formal y compartida de una conceptualización. Este enfoque resulta más flexible que el de una aplicación que responda directamente, ya que una ontología es una representación explícita de las teorías subyacentes que puede ser modificada para ajustarse a cambios, como por ejemplo, los originados por nuevas tecnologías en consolas o dispositivos móviles.

En (Pan et al. 2012), la expresión “desarrollo de software dirigido por ontologías” o ODS (por las siglas de su nombre en inglés: ontology-driven software development) es usado para referirse a la aplicación de tecnologías de ontologías (Baader et al. 2010) para facilitar los procesos de desarrollo de software, y específicamente, la práctica de desarrollo de software dirigido por modelos o MDS (de las siglas de model-driven software development). Los autores

exploran la aplicación al contexto del desarrollo de software de las capacidades de las ontologías para chequear consistencia y proveer guía.

Estos enfoques, en general, implican ontologías de distintos tipos (Guarino 1998). Se suele requerir el desarrollo/reutilización de ontologías de dominio, y el desarrollo de ontologías de aplicación. En ambos casos, las ontologías pueden incorporar entidades de ontologías de alto nivel. Las relaciones entre estas ontologías se desarrollan conforme a la disciplina de correspondencia entre ontologías (Euzenat and Shvaiko 2013).

A su vez, existen metodologías de desarrollo que permiten la participación, supervisión, y evaluación de los productos por parte de expertos del dominio. Esto es particularmente importante de cara al objetivo de que se consideren las conceptualizaciones de expertos de una variedad de dominios. Entre dichas metodologías podemos citar la propuesta en (Grüninger and Fox 1995), o en (Uchold and King 1995), o Kactus (Bernaras, Laresgoiti, and Corera 1996), o Methontology (Gómez-Pérez, Fernández-López, and De Vicente 1996), Ontology (Staab et al. 2001), Terminae (Corcho, Fernández-López, and Gómez-Pérez 2003), o REFSENO (Tautz and Wangenheim 1998),

El concepto de ontología y sus fundamentos y herramientas se han aplicado en el desarrollo de videojuegos. Por ejemplo, en (Llansó García 2014; Llansó et al. 2011) se propone una metodología para definir de manera formal y ágil el dominio del juego utilizando análisis formal de conceptos o FCA (por su nombre en inglés formal concept analysis). En (Leon Z. and

Sanchez 2010) utilizan UML y OCL para bosquejar una ontología para videojuegos para dispositivos móviles. El trabajo de (Trindade 2015) se enfoca en el desarrollo de una ontología para identificar requerimientos técnicos en el diseño de juegos.

Líneas de investigación, desarrollo e innovación

Esta línea de I-D-I investiga y desarrolla ontologías para dirigir el diseño arquitectónico de videojuegos. La línea se encuentra en etapa inicial. Se investigará la conceptualización para la toma de decisiones de diseño arquitectónico, se desarrollarán ontologías que formalicen este conocimiento y permitan luego el desarrollo de aplicaciones que puedan guiar al diseñador y detectar inconsistencias en especificaciones arquitecturas.

Objetivos y resultados esperados

El objetivo es especificar de manera explícita y formal el conocimiento necesario para decisiones de diseño arquitectónico en el desarrollo de videojuegos. Las ontologías abordarán los aspectos fundamentales del dominio de videojuegos y las teorías para responder a preguntas como las planteadas en los ítems A, B y C de la sección Introducción. Se busca proyectar aplicaciones dirigidas por dichas ontologías que asistan al diseñador en las decisiones de diseño y en la detección de inconsistencias. Se espera obtener:

- Ontologías sobre videojuegos especificando conceptos tales como dispositivos, software de base, motores de videojuegos, y las dependencias entre estos.

- Ontologías especificando una teoría lógica para responder a preguntas de diseño como las indicadas en los puntos A, B y C.
- Una aplicación prototípica que resulte una respuesta innovadora a los desafíos de diseño arquitectónico en el ámbito de los videojuegos.

Formación de Recursos Humanos

Esta línea de I-D-I continúa con la formación en investigación de alumnos de grado y posgrado de la UNSL. Está previsto que uno de los autores desarrolle en la línea su Tesis de Maestría en Ingeniería de Software, así como también dos alumnos de grado sus tesinas. Es importante destacar el apoyo académico y científico para los temas desarrollados en la formación de recursos humanos calificados, ya que son carreras de posgrado categorizadas "A" para los desarrollos de las tesis de posgrado y carreras de grado acreditadas por 6 años para los desarrollos de las tesinas.

Referencias

Baader, Franz et al. (2010). *The Description Logic Handbook: Theory, Implementation and Applications*. 2nd. New York, NY, USA: Cambridge University Press.

BBC-Tech (2016). No Man's Sky investigated over 'misleading' adverts.

Bernaras, A., I. Laresgoiti, and J. Corera (1996). "Building and Reusing Ontologies for Electrical Network Applications". In: *Proceedings of the 12th European Conference on Artificial*

Intelligence (ECAI'96). Ed. by Wolfgang Wahlster. Budapest, Hungary: John Wiley and Sons, pp. 298–302.

Corcho, Oscar, Mariano Fernández-López, and Asunción Gómez-Pérez (2003). “Methodologies, Tools and Languages for Building Ontologies: Where is Their Meeting Point?” In: *Data Knowl. Eng.* 46.1, pp. 41–64.

Dingman, Hayden (2014). Atari: Game Over film review: Unearthing the El Dorado of video games. PC World.

Essential facts about the computer and video game industry (2016). Tech. rep. Entertainment Software Association.

Euzenat, Jérôme and Pavel Shvaiko (2013). *Ontology matching*. 2nd. Heidelberg (DE): Springer-Verlag.

Gómez-Pérez, Asunción, Mariano Fernández-López, and A.J De Vicente (1996). “Towards a Method to Conceptualize Domain Ontologies”. In: *ECAI-96 Workshop on Ontological Engineering*. ECAI-96 Workshop Proceedings.

Gruber, Thomas R. (1993). “A Translation Approach to Portable Ontology Specifications”. In: *Knowl. Acquis.* 5.2, pp. 199–220.

Grüninger, M. and M. Fox (1995). “Methodology for the Design and Evaluation of Ontologies”. In: *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing*, April 13, 1995.

Guarino, Nicola (1998). *Formal Ontology in Information Systems: Proceedings of*

the 1st International Conference June 6-8, 1998, Trento, Italy. 1st. Amsterdam, The Netherlands, The Netherlands: IOS Press.

Leon Z., Annie C. and Abraham L. Sanchez (2010). “An Ontology for Mobile Video Games”. In: *2010 Ninth Mexican International Conference on Artificial Intelligence*, pp. 154–159.

Llansó García, David (2014). “Metodología ontológica para el desarrollo de videojuegos”. PhD thesis. Universidad Complutense de Madrid.

Pan, Jeff Z. et al. (2012). *Ontology-Driven Software Development*. Springer Publishing Company, Incorporated.

Schiesel, Seth (2011). 14 Years of Waiting Have Come to an End. NY Times.

Staab, Steffen et al. (2001). “Knowledge Processes and Ontologies”. In: *IEEE Intelligent Systems* 16.1, pp. 26–34.

Tautz, Carsten and Christiane Gresse von Wangenheim (1998). *REFSENO: A representation formalism for software engineering ontologies*. Tech. rep. 15. Fraunhofer IESE.

Trindade, Glauco Ofranti (2015). “Video game development ontology”. PhD thesis. Universidade Federal do Rio de Janeiro.

Uschold, Mike and Martin King (1995). “Towards a methodology for building ontologies”. In: *Workshop on Basic Ontological Issues in Knowledge Sharing*.