

Tecnologías Middleware: ThingWorx

Raúl Castro Moreno

¹ Universidad de Granada, Hospital Real, Avenida del Hospicio, S/N C.P. 18010

² ETS Ingeniería Informática y Telecomunicaciones, Calle Periodista Daniel Saucedo Aranda,
s/n, 18014 Granada
raulcastromor@correo.ugr.es

Resumen. El documento trata sobre un middleware en específico llamado ThingWorx y comercializado por PTC. En este trabajo se habla específicamente de los objetivos que quiere llevar a cabo este middleware, también de las características que posee de cara al usuario y las propiedades que se tuvieron en cuenta a la hora construirlo. Sobre esa construcción de este software se habla también del diseño que se lleva a cabo y también los mecanismos y protocolos que usa para la comunicación con sus servidores y los clientes que lo usan. Finalmente se concluye con un análisis crítico donde se comentan las ventajas y desventajas que tiene, y además unas conclusiones del trabajo y unos comentarios al respecto sobre su realización.

Palabras Clave: Middleware, ThingWorx, IoT, API, REST, cosa(refiriendose a Thing de IoT)

1 Introducción y Objetivos

En primer lugar comentaremos que es un Middleware [1], el cual se trata de un software que se encuentra entre los Sistemas Operativos, y las aplicaciones que se ejecutan en él. Funciona básicamente como una capa traductora oculta, que permite la comunicación y la administración de datos en aplicaciones distribuidas. El uso de un middleware puede permitir a nivel de usuario, el realizar solicitudes tales como el envío de formularios en un navegador web, o que un servidor web devuelva páginas al usuario que son dinámicas en función del perfil de un usuario. En el caso de este trabajo, nos centraremos en explicar una tecnología de Middleware llamada ThingWorx.

ThingWorx es la plataforma de innovación industrial líder en la industria que está diseñada para entregar rápidamente aplicaciones de IoT y experiencias de realidad aumentada (AR) que desbloquean el valor de los mundos digital y físico convergentes. Es por ello que nace el concepto de IIoT (Industrial Internet of Things) en el cual se basa ThingWorx por su aplicación al ambiente industrial.

ThingsWorx consigue realizar desde la supervisión y mantenimiento remotos al aumento de la eficacia de la plantilla y la optimización de activos, logrando así resolver los retos comunes de los diversos sectores industriales actuales.

Ya que el desarrollo de soluciones de IIoT se suele considerar un punto débil, ThingWorx creó con el objetivo de eliminar ese impedimento, permitiendo así el

paso de soluciones a proyectos piloto, a soluciones a escala empresarial mediante el uso de aplicaciones preintegradas y herramientas de desarrollador.

Ha sido desarrollado por PTC (Parametric Technology Corporation) , empresa que actualmente tiene productos y servicios que incluyen IoT , AR y softwares de colaboración.

2 Diseño

ThingWorx [3] proporciona una capa semántica en su modelo y no debe usarse en formas de programación tradicionales. Deben considerarse las cosas y características de las cosas necesarias para la solución. Con el modelo orientado a objetos de ThingWorx, se puede descomponer en funciones generales de objetos de nivel superior, encapsular estas funciones y utilizar lógica empresarial en los objetos base, que pueden heredar de cosas digitales. Este proceso crea una capa semántica que permite realizar cualquier tipo de preguntas sobre las cosas. La semántica del modelo se expone como un conjunto de API de REST bien conocidas y fuertemente tipadas.

Se utiliza también un generador de aplicaciones visuales que son necesarios para poder consumir los resultados del modelo semántico centrado en cosa y que proporcionan el desarrollo rápido de aplicaciones. Para ello se usa una solución que comprenda la capa semántica del modelo de datos y se ajuste cuando cambie el modelo de datos. ThingWorx utiliza para ello ThingWorx Mashup Builder, que proporciona una capa en el servidor Edge, la nube y el modelo de cosa. También ofrece widgets visuales diseñados para el IoT, crear páginas de interfaz de usuario sin necesidad de codificación.

2.1 Estructura de Componentes

Los componentes de ThingWorx se pueden plantear en 3 capas:

- **Cliente:** Esta capa esta compartida por las **Cosas/dispositivos** , donde se incluyen los dispositivos , agentes y demás activos que le envían y reciben información de ThingWorx ; y los **Usuarios/Clientes** incluyendo los productos(siendo estos principalmente navegadores Web) que las personas utilizan para acceder a ThingWorx.

- **Aplicación:** En esta capa es donde reside ThingWorx, la cual proporciona la conectividad con la capa Cliente , y también realiza verificaciones de autenticación y autorización, y procesando y analizando contenido que le llega.

- **Datos:** Aquí se conservan y almacenan los metadatos del modelo y los datos del sistema. En los metadatos se incluyen definiciones de entidad de ThingWorx, definiciones de dispositivos y de propiedades asociadas. Los datos, que pueden ser tabulares o temporales, se almacenan como filas de contenido de blogs, wikis, flujos de valor, tablas de datos, etc.

Además, tiene **2 opciones de despliegue**, siendo la primera la de PTC Cloud Services, siendo esta una implementación de servicios gestionados, donde la aplicación de ThingWorx se aloja y gestiona un servidor de terceros siendo este a menudo una nube privada. La otra sería la implementación local donde se instala y ejecuta el software de ThingWorx en servidores del sitio o centros de datos propios del usuario, siendo este responsable entonces del abastecimiento, instalación y mantenimiento de la infraestructura y el software.

3 Características

ThingWorx posee unas funcionalidades software [4] totalmente innovadoras donde se destacan:

- **ThingWorx Composer:** Es un entorno de modelado de aplicaciones el cual permite cubrir todo el proceso de construir aplicaciones desde el inicio hasta el final, de forma fácil e intuitiva.

- **Mezclas sin Código:** Se ofrece un constructor de mezclas sin código por parte del software pudiendo crear aplicaciones IoT de manera que sean completamente escalables sin ninguna necesidad de escribir código. Esto se consigue mediante la tecnología “arrastrar y soltar” utilizada por la plataforma.

- **Integration Hub:** Siendo este un marco de integración en el cual convergen la información digital y física, permitiendo así la innovación tanto de nuevas aplicaciones como de procesos empresariales.

Ya hablando más de características [5] como tal que deba saber un usuario final sobre ThingsWorx tendríamos:

- **Análisis :** Permite obtener información útil en tiempo real, a partir de los datos recogidos mediante IoT, consiguiendo así optimizar las operaciones proactivamente y evitando así posibles problemas futuros.

- **Creación :** Permite utilizar unas potentes aplicaciones y herramientas preintegradas que dan la facilidad para la creación de soluciones empresariales de IoT y de experiencias de AR .

- **Gestión:** Se puede controlar y gestionar todos los sistemas ,dispositivos y procesos conectados a la plataforma pudiendo fomentar el rendimiento y dando visibilidad a las operaciones realizadas.

- **Experimentación:** Ofrece un gran abanico de formas más eficaces y seguras a los equipos de ingeniería y desarrollo, para que interactúen con los objetos físicos y sistemas, y además, consigue tener más capacidades para los equipos.

- **Conexión:** Consigue ampliar la conectividad industrial estandarizada entre aplicaciones y dispositivos consiguiendo así el acceso a varias fuentes de datos.

4 Mecanismos, Protocolos y Servicios para la Comunicación

Se utiliza el **protocolo de Odata (Open Data Protocol)**, que se define como un protocolo que sirve para realizar consultas y actualizar los datos que utilizan los protocolos Web existentes. Está basado en REST, y se ha creado sobre tecnologías estandarizadas como JSON, XML y HTTP. Es una tecnología flexible que consigue proporcionar una gran interoperabilidad entre las aplicaciones, los clientes, los orígenes de datos y los servicios.

Se utiliza también un **mecanismo** proporcionado por ThingWorx, llamado **Remote Access Client (RAC)**, el cual permite utilizar las funciones de acceso remoto de ThingWorx de un mashup personalizado. Un mashup son las visualizaciones de la página web que se utilizan para entregar información al modelo de ThingsWorx.

ThingsWorx además presenta una API de REST, la cuál es una herramienta muy importante de comunicación, proporcionando esta muchas de las bondades de comunicación que son utilizadas por los servicios Web, pero omitiendo la mayoría de las complejidades que presentan. El resultado de esto, es un trabajo con REST mucho más sencillo que puede ser utilizado por cualquier cliente que realice una solicitud HTTP.

En cuanto a la **comunicación entre sistemas y dispositivos**, se usa el concepto de **Federación** que permite la descarga de una gran carga de trabajo de solución entre servidores ThingsWorx. Esto se basa en que si hay servidores regionales o agrupaciones de servidores que sean independientes, pues se pueden establecer con ellos conexiones de datos de uso compartido, entre un gran abanico de usuarios, utilizando estos una federación para descargar y manipular datos. ThingsWorx se puede utilizar además para implementar estas conexiones, y los dispositivos pueden descargar servicios y propiedades entre servidores. También se puede utilizar una federación para obtener servicios y propiedades de otros dispositivos, pudiendo acceder a cualquier dispositivo publicado como dispositivo remoto desde un servidor federado.

Por último, explicar el trabajo **dirigido por eventos, alertas y suscripciones**. Estos eventos son utilizados para indicar un cambio de condición en la plataforma de ThingWorx. Si se cumple una de las condiciones motivadas por algún evento, es entonces cuando se llama a un servicio o se realiza una llamada REST. Estos eventos requieren una definición de datos predefinida donde se almacenan datos asociados al evento, a los que puede acceder una suscripción. Una suscripción es un servicio especial, el cual se ejecuta en el momento que se activa el evento al que se está suscrito. Este servicio puede ejecutar una alerta cuando se activa un evento. Las alertas son un tipo de evento estándar, donde se basa en el estado de una propiedad. Estas se pueden activar automáticamente si el valor de una propiedad, cumple alguna condición establecida previamente por un usuario.

5 Propiedades

Como propiedades [4] del middleware, comentaremos los atributos de calidad que presenta basado en los requisitos no funcionales en los cuales se han basado el equipo de desarrollo a la hora de construir este middleware.

- **Rapidez:** Se consigue una velocidad de implementación 10 veces más rápida debido al desarrollo basado en modelos, aumentando así la velocidad de producción.

- **Compleitud:** Un gran abanico de funciones muy completos y con sentido y lógica entre ellos, siendo así una plataforma muy completa y moderna.

- **Escalabilidad:** Se permite una gran evolución y crecimiento de la aplicación con el tiempo, permitiendo así adaptarse a nuevas tecnologías futuras, que aumenten las funcionalidades, y la optimización y rapidez del software.

- **Flexibilidad:** Permitir a los usuarios implementar sus aplicaciones de la manera que deseen basándose en las necesidades que ellos tengan.

- **Simplificación:** Que todas las funcionalidades, no requieran un gran nivel de conocimiento, siendo éstas más simples de comprender y utilizar en comparación con otras del mercado.

- **Combinación:** Conseguir que haya una buena sincronización entre las personas, los sistemas y la maquinaria, fomentando así una gran anticipación a los posibles fallos.

6 Análisis crítico

En este análisis comentaré las ventajas y desventajas de este middleware, sin embargo hay que decir, que desventajas tiene muy pocas, por no decir casi ninguna. Esto es debido a que ThingsWorx es uno de los software top, en IIoT, colocado en ese puesto por varios analistas como pueden ser el **Cuadrante mágico de Gartner**, **IDC MarketScape** y el **Informe Forrester Wave**

Basado en lo comentado en los apartados anteriores, como ventajas podemos decir que :

- Mejora la calidad
- Reduce costes y tiempos
- Es flexible para evolucionar
- Acelera la innovación
- Gracias a las 4 anteriores, aumenta los beneficios económicos.

Y como desventajas podemos encontrar que :

- Hay que pagar por el servicio.
- Se tiene que adaptar todo el entorno industrial a la aplicación para poder realizar la transformación a lo digital y así conseguir que todo vaya perfectamente , obteniendo de esta manera las ventajas comentadas anteriormente en la empresa

Hay que tener en cuenta también otros middleware, pero a diferencia de la mayoría de ellos después de indagar un poco, ThingsWorx consigue ofrecer una plataforma IIoT completa con una gran cantidad de capacidades integrales, permitiendo así hacer frente a todos los aspectos de la transformación digital.

7 Conclusiones

Mis conclusiones que puedo sacar de este trabajo, y sinceramente, de la mayoría del contenido que se ha visto en la asignatura de DSD relacionado con los middlewares y la nube y demás, es que el futuro es la transformación de todos los ámbitos a lo digital, y más con el ejemplo de este middleware el cuál tiene que ver con llevar a lo digital a todo el ámbito industrial, abarcando esto, desde la agricultura, las ciudades y el agua, hasta las naves industriales, la construcción y la telemática. El poder equipar todo ello con una conectividad que permita la continua obtención y análisis de datos que permita incluso prever hasta posibles errores es algo fascinante. Además de la cantidad de ventajas que suponen ya que no hay muchas empresas que comenten casos de que el transformarse a lo digital les haya ido mal y no hayan conseguido reducir costes y tiempo y aumentar el beneficio. Ya hablando específicamente de ThingWorx, es una de las mejores soluciones del mercado, y aparte de todas sus características y demás, lo mejor es que la empresa que lo desarrolla y comercializa es PTC , ya que esta empresa lleva desde 36 años en el desarrollo software, y siempre se han dedicado a innovar sus productos, y lo más importante, a mantenerlos, actualizarlos y arreglarlos en caso de errores, dando así una seguridad de que al utilizar ThingsWorx, siempre vas a tener un gran soporte que te ayude.

8 Comentarios finales

Comentar, que en este trabajo, se ha buscado información por todos sitios, desde libros, páginas web y vídeos explicativos. Sin embargo la mejor información y de la cuál esta basada la mayoría del trabajo, sobre todo cuando esa información es más basada en el nivel técnico del middleware, está sacada del centro de ayuda que da PTC sobre ThingsWorx, enlace el cuál esta referenciado. Es ahí donde aparecen explicados la gran mayoría de los aspectos de este Middleware, sobre todo el diseño, y los protocolos que usa. Comento esto básicamente para informar, que todos los

apartados del trabajo en los cuales se está hablando de algo obtenido del centro de ayuda, no se ha marcado con la referencia para no tenerla repetida un montón de veces.

Referencias

1. Que es Middleware? , <https://azure.microsoft.com/es-es/overview/what-is-middleware/> , last accessed 2021/06/05.
2. PTC ThingWorx, <https://www.ptc.com/es/products/thingworx>, last accessed 2021/06/05.
3. Centro ayuda ThingWorx, https://support.ptc.com/help/thingworx/platform/r9/es/index.html#page/ThingWorx%2FHelp%2FBest_Practices_for_Developing_Applications%2Fbuild_deploy_applications.html%23, last accessed 2021/06/07.
4. Funcionalidades y Propiedades ThingWorx, <https://blablanegocios.com/que-es-thingworx-como-funciona/> , last accessed 2021/06/09.
5. Características ThingWorx , <https://integralplm.com/thingworx/#contacto> , last accessed 2021/06/09.