

12

CAMINO HACIA LA CREACIÓN DE CLIENTES Y SERVIDORES BAJO EL ESTÁNDAR UA DE LA FUNDACIÓN OPC*

Revista Inge-CUC / Vol. 6 - No. 6 / Octubre 2010 / Barranquilla - Colombia / ISSN 0122-6517

157

Por: Rubén Darío Sánchez Dams**

Fecha de recibido: 6 de julio de 2010 • Fecha de aceptación: 30 de septiembre de 2010

RESUMEN:

El presente artículo recorrió la creación de las aplicaciones cliente y servidor OPC, las cuales permiten el acceso a datos de procesos industriales en el campo de forma estandarizada. Se describieron conceptos fundamentales, y las justificaciones de la tecnología analizando las posibilidades e implicaciones con las que se cuenta en la implementación de una nueva aplicación. A su vez se mostró la implementación de una solución de este tipo, para lo cual se expusieron las principales alternativas para la creación de estos clientes y servidores, y finalmente se describió la implementación de ellos.

PALABRAS CLAVE:

OPC UA DA, Servidor OPC, Cliente OPC, Fundación OPC, Procesos industriales, SCADA, Redes de campo y Variables de campo.



* Derivado de la investigación Estudio e implementación de un servidor y cliente en el estándar OPC UA-DA, grupo GIACUC, en la línea de Sistemas de Automatización y Control de Procesos.

** Ingeniero Electrónico, Especialista en Ingeniería de Procesos Industriales, Profesor de Tiempo Completo del programa de Ing. Electrónica de la Corporación Universitaria de la Costa, Facultad de Ingeniería. rsanchez5@cuc.edu.co.





12

ROAD TO THE CREATION OF CLIENTS AND SERVERS UNDER THE OPC UA STANDARD OF THE OPC FOUNDATION

By: Rubén Darío Sánchez Dams

ABSTRACT:

This article road through the creation of an OPC client and server applications, which allow the access to industrial process data on the field in a standardized way. Fundamental concepts were described and the justification for the choosed technology, analyzing the implications and possibilities which we account in

an implementation of a new application. At the same time was shown the implementation of a solution of this sort, for which was presented the main alternatives for the creation of these clients and servers, and finally was described the implementation of these.

KEY WORDS:

OPC UA DA, OPC Server, OPC Client, OPC Foundation, Industrial Processes, SCADA, Field Networks, and Field Variables.



INTRODUCCIÓN

“OPC” es un estándar que establece un idioma común para el intercambio de información entre diferentes gamas de dispositivos y fabricantes en el área de la automatización y el control industrial. Esto permite que diferentes fabricantes y gamas de productos puedan compartir información mediante un lenguaje común que todos entiendan. Este estándar nació como respuesta de la gran diversidad y disparidad de protocolos propietarios que existía en los inicios de la automatización con equipos electrónicos. Como respuesta a esta problemática nació la “OPC Foundation”, la cual se creó como la organización reguladora de los estándares que permiten la comunicación entre dispositivos de automatización. Cada estándar nace del consenso de la industria y el sector académico, para tal fin se generan comités creadores multidisciplinarios, en donde participan las principales empresas del mundo en el área de la automatización y universidades con investigaciones en el área. A partir de los estándares OPC mencionados, nacen aplicaciones, de las cuales las comerciales son las que se aplican como soluciones en la industria. Pero para llegar a aplicar esta tecnología y dar solución a problemas concretos, es indispensable conocerla, y saber con qué alternativas y formas de implementación se cuenta, cuestionamientos que llevaron a realizar una investigación que dio como resultado el presente escrito.

JUSTIFICACIÓN

El OPC es un conjunto de estándares utilizados internacionalmente de forma amplia por todo tipo de industria el cual es soportado por los principales fabricantes de equipos de automatización y control, este estándar cuenta con varias especificaciones adecuadas a las principales necesidades de intercambio de información con el proceso. Actualmente hay una nueva versión unificada del estándar el cual se divide en dife-

rentes perfiles que componían las especificaciones anteriores, esta fue liberada en su primera versión en febrero de 2009 comprendiendo las principales partes de toda la especificación (parte I a la 8, 10 y 11) la parte 9 (Alarmas y condiciones) fue la última en liberarse la cual se dio en su versión definitiva en 2010.

Debido a la amplia acogida que han tenido los estándares anteriores de la OPC Foundation (ahora conocidos como OPC clásicos) y al apoyo que las principales empresas en automatización le han dado a todos los estándares que ha realizado la fundación, se puede afirmar sin riesgo a equivocación que OPC seguirá liderando la convención para el intercambio de información entre el nivel de planta, el manejo de procesos y los niveles administrativos. Esta afirmación es válida incluso para el último estándar OPC UA, debido a que los principales actores de este tipo de tecnología, estuvieron desarrollando la especificación desde 2003¹ (establecimiento de la visión) hasta 2010, intervalo de tiempo en el que se desarrolló todo el nuevo estándar, con las primeras aplicaciones de las mismas apareciendo entre el 2009 y 2010. Todo este tiempo y esfuerzo en el desarrollo muestran un gran compromiso que ha durado alrededor de siete años, que junto al consenso del sector de la automatización industrial, hace vislumbrar un amplio despliegue de esta nueva tecnología en los próximos años. Por lo tanto el presente trabajo socializa la construcción de aplicaciones básicas que usan tecnología servidores y clientes OPC, pretendiendo facilitar la transición hacia el uso de la nueva especificación.

GENERALIDADES DE LA TECNOLOGÍA

El servidor OPC, es el software que se encarga de acceder directamente a los datos comunicándose con la fuente primaria de datos; este intercambio de datos con la fuente (comúnmente un autómatas programable), se realiza usual-

1. [6] p9 - p96 y [7].



mente mediante un bus de campo. Así mismo se puede definir al cliente OPC, como el software que sirve de mecanismo genérico de acceso a datos de campo, poniéndolos a disposición de las aplicaciones de usuario. La ventaja de separar el acceso a los datos con un servidor y un cliente, es que de esta forma y gracias a OPC, los clientes pueden acceder de forma genérica a cualquier servidor de datos OPC, evitando que cada aplicación de usuario necesite conocer o implementar la comunicación con todos los posibles hardware de campo, los cuales son principalmente autómatas programables de los diferentes fabricantes. Así se le delega a cada fabricante el acceso a los datos de sus controladores programables, exigiéndole no obstante, que los publique de forma estándar bajo algún modelo de la OPC Foundation mediante un servidor OPC.

Inicialmente las primeras tecnologías representadas por las siguientes siglas OLE, DDE, COM y DCOM OLE permitieron la comunicación entre aplicaciones y dispositivos. DDE² permitía compartir sesiones para el intercambio de datos entre aplicaciones y llegó a ser utilizado como interfaz entre los autómatas programables de los procesos, y los computadores. Sin embargo, el desarrollo de posteriores tecnologías para optimizar el rendimiento de DDE trajo consigo que los fabricantes diseñaran soluciones de tipo propietario con características diferentes e incompatibles entre ellas. Para 1993 Microsoft lanza la plataforma COM³ que no solo permitía la comunicación entre aplicaciones sino que también estas podían estar construidas con distintos lenguajes de programación. Pero la tecnología COM también presentaba áreas susceptibles de mejorar por lo cual se reemplazó con su sucesora DCOM⁴, mediante esta versión se mejoraron las comunicaciones entre aplicacio-

nes y ordenadores que hacían parte de una red, hasta el punto que hoy todavía se usa junto con la tecnología actual. “Net”; todas estas tecnologías son propietarias de la compañía Microsoft.

Debido a los inconvenientes sufridos de incompatibilidad en las comunicaciones entre diferentes fabricantes, con desarrollos propios basados en DDE, un grupo de empresas decidió trabajar por la unificación y estandarización de la comunicación entre los equipos de automatización creando la OPC Foundation. Ellos empezaron a trabajar en un conjunto de especificaciones para tal fin, siendo la especificación de Acceso a Datos (DA) la primera, la más importante, y la más usada, este grupo definió unos mecanismos fundamentales para la lectura y escritura de datos de proceso. Luego se desarrollaron otros estándares que dieron solución a problemas concretos de comunicaciones en la industria de la automatización, estos fueron⁵ “Alarmas y Eventos OPC” denominado OPC AE, “Acceso a Datos Históricos OPC” (OPC HDA), “Intercambio de Datos OPC” (OPC DX), entre otros, los cuales a su vez evolucionaron en versiones que agregaban nuevas funcionalidades o corregían otras, de acuerdo a las necesidades que experimentaba la industria.

Posteriormente y gracias a la optimización de los recursos web, le siguió OPC XML-DA que desde entonces permitió asociar el mapeo de la especificación de acceso a datos a los servicios Web y la utilización de componentes OPC a través de Internet y en diferentes plataformas del sistema operativo sin necesidad de la tecnología propietaria DCOM. La arquitectura OPC basada en COM fue reevaluada en este punto debido a que esa plataforma se consideraba desactualizada para el uso en servicios Web y SOA⁶, sin embargo no se migraron todos los perfiles de OPC

2. De sus siglas en inglés para Dynamic Data Exchange, intercambio dinámico y vinculación de objetos.
3. De sus siglas en inglés para Component Object Model, modelo de objetos de componentes.
4. De sus siglas en inglés para Distributed Component Object Model, modelo de objetos de componentes distribuidos.
5. Las siglas de los diferentes estándares corresponden a su denominación en inglés.
6. De sus siglas en inglés para Servicios Orientados a Arquitectura.

a tecnologías basadas en la Web, concibiendo en cambio una nueva arquitectura unificada, la cual permitiría la integración de los anteriores modelos OPC (DA, A&E, HDA...). Como resultado se ideó y desarrolló el estándar OPC de arquitectura unificada, el cual en el momento en que se escribió este artículo la OPC Foundation se encontraba concentrada en la difusión y publicidad, que permita la transición de OPC Clásico (DA, A&E, HDA, etc.) hacia la arquitectura unificada (UA), basada en la implementación sobre XML y otras tecnologías de redes.

METODOLOGÍA

Esta investigación fue de tipo descriptiva en cuanto a que identificó el estado actual en que se encuentran las soluciones tecnológicas de intercambio de datos de procesos industriales, centrándose en la generalmente aceptada, la cual es OPC. Adicionalmente se sintetizaron las temáticas de acuerdo a una revisión bibliográfica dando como resultado del trabajo de investigación una monografía, por lo que también se utilizaron lineamientos del método sintético. Adicionalmente para la construcción de la aplicación se utilizaron actividades y prácticas de los métodos de desarrollo de software Programación Extrema y del Proceso Unificado de Rational (XP y RUP, por sus siglas en inglés), de estos procesos solo se usaron algunas actividades y lineamientos, debido a que en el desarrollo del proyecto, trabajo un solo ingeniero de tiempo parcial y estas metodologías necesitan más recursos para ser implementadas en su totalidad. Por lo anterior también se considera en parte una investigación de tipo aplicada.

DESARROLLO DEL PROYECTO, CREACIÓN DE UN CLIENTE Y SERVIDOR OPC

La creación de una aplicación demostrativa no es una necesidad arbitraria ni audaz, el autor del presente trabajo ha experimentado anteriormente con la tecnología pero desde la perspectiva de un usuario, mediante el uso y desarrollo de sistemas de supervisión y control a través

de redes de campo, como WinCC de la empresa Siemens, e Intouch de la compañía Wonderware. Para un usuario, los detalles técnicos se encuentran ocultos, liberándolo de las complicaciones de la tecnología, se le quita la carga del “cómo”, centrándolo en el “para que”, a esto en informática se le denomina encapsulamiento, técnica que permite manejar la complejidad de los sistemas actuales. Los fabricantes de sistemas de supervisión y servidores OPC de autómatas programables utilizan el encapsulamiento mediante el uso de asistentes para la configuración de las comunicaciones, o la puesta a disposición de pantallas de configuración. Un ejemplo de esto se da en proyectos de automatización y control con la necesidad de conectar autómatas del tipo Simatic con un sistema de supervisión de otro fabricante que podría ser Inthoch. Este planteamiento evidencia la necesidad del uso de un servidor OPC que acceda a la red de datos de los equipos Siemens y los pusiera a disposición del sistema SCADA Inthoch (supervisión), para lo cual habría solo que configurar por el lado de los equipos de campo el servidor OPC, y por el lado del SCADA el cliente OPC, ambos de fabricantes diferentes.

El autor posteriormente trabajó en soluciones propietarias de acceso a variables de campo sin el uso de OPC, permitiéndole identificar que los sistemas de supervisión genéricos de disposición general son costosos. Estos sin embargo son utilizados debido a que en una solución única para un proceso particular en una empresa X, son más baratos, que desarrollar un software desde cero, adicionalmente por lo genérico son más fáciles de mantener y actualizar. Cuando es necesario desarrollar una aplicación que acceda a variables de campo que se necesite replicar muchas veces sin cambio, se vuelve prohibitivo el uso de los sistemas SCADA y servidores OPC genéricos de los principales fabricantes, debido a que se tiene que comprar una para cada instancia de la solución.

Posteriormente al autor se le planteó la necesidad de construir servidores y funcionalidades



de clientes OPC, por lo que incursionó en la implementación mediante SDKs⁷ y asistentes para el desarrollo de servidores y clientes OPC clásicos, provista por terceros. Lo experiencia previa le permitió tener en la investigación desarrollada, acceso directo a las especificaciones y estándares, profundizando en el uso de SDKs comerciales y en los disponibles para los miembros de la Fundación OPC, lo que en últimas permitió realizar la investigación originaria del presente artículo.

Para el proyecto realizado y debido a la experiencia previamente adquirida a la ejecución del proyecto, y a los limitados recursos de personal disponibles para realizarlo (unipersonal) se centraron los esfuerzos solo en el desarrollo del Servidor OPC. La principal razón que justifica esta decisión es que en el desarrollo de la investigación se encontró que hay clientes gratuitos [1] [2] con todas las funcionalidades implementadas y con excelentes interfaces gráficas, las cuales los hacen idóneos para demostraciones de la tecnología y permiten dejarlo de lado, se comparte en el siguiente enlace información de clientes OPC gratuitos: <http://www.opcconnect.com/xml.php> (para DA-XML) y <http://www.opcconnect.com/uafree.php> para UA. Adicionalmente a lo anterior los clientes OPC tienen otra alternativa de desarrollo que requiere bastantes conocimientos de desarrollo de software, por lo que adicionalmente es necesario que el usuario conozca minucias técnicas para lidiar con todas las características de las especificaciones y así implementar con éxito una aplicación de este tipo, la ventaja es que no se necesita tener acceso a ningún SDK propietario. Esta posibilidad se da debido a que la tecnología Web de los servidores OPC UA permite generar toda su especificación en el lenguaje de descripción de servicios Web (WSDL), en formato XML⁸. Para esto es necesario tener un entorno de desarrollo capaz de capturar la

especificación del servidor WSDL y generar el código pertinente para interactuar con él, como Visual estudio de Microsoft o el entorno gratuito Eclipse soportado por IBM, después de lo cual el entorno y sus plugins generan unas funciones o métodos que pueden ser llamados desde el programa desarrollado cliente OPC. Por otro lado sin embargo los servidores OPC son los encargados de acceder mediante redes de campo a hardware específico, por lo que no hay un servidor genérico que permita comunicarse con cualquier hardware, así y debido a la necesidad futura de conectarse con un hardware propietario (PLC pedagógico de la CUC, Ing. Electrónica [8]), se concentró el desarrollo en un Servidor OPC demostrativo.

En cuanto al acceso a datos de campo, la solución más sencilla es la utilización de un producto ya construido, que permita el acceso a datos del proceso. Los sistemas de supervisión son el software que comúnmente se usa para el acceso, manejo y visualización de los datos de proceso, tienen sus propios clientes y servidores OPC implementados, con los que el usuario en tiempo de desarrollo interactúa para crear la aplicación. Sin embargo como se ha indicado, el uso de esta solución no siempre es posible. Si es necesario repetir muchas veces la misma aplicación, es mejor desarrollar una propietaria, principalmente por costos y control de la solución.

Se decidió entonces implementar una solución propietaria que cumpla con lo dispuesto por la Fundación OPC, para lo cual es necesario decidirse en cuál de los estándares hay que hacerlo. Se presentó la opción de utilizar la especificación clásica, que comercialmente hablando mientras se escribía el artículo se encontraba vigente. Esta opción se puede desarrollar en cualquiera de sus estándares DA, AE, HD, etc.; a su vez su escogencia depende del tipo de aplicación al que

7. De sus siglas en inglés para Software Development Kit, kit de desarrollo de software.

8. Para más detalle se recomienda al lector ver la página siguiente: <http://lists.opcfoundation.org/RandyBlog/archive/2009/08/31/a-roadmap-to-ua-downloads.aspx>, ver el ítem 4 "I have a development environment that allows me to ingest a WSDL and start writing code" (Tengo un entorno de desarrollo que me permite ingerir una especificación WSDL y comenzar a escribir código).



se destina la solución. Así mismo se contó con la elección de utilizar el nuevo estándar OPC UA, que en vez de disponer de estándares diferentes para distintas aplicaciones, cuenta con perfiles incluidos dentro del estándar, que como se esperaba son DA, AE, HD, etc., correspondiendo respectivamente a sus predecesores. Estando ahora todas las especificaciones dentro de un solo estándar unificado, con una sola arquitectura, se facilitan los desarrollos que impliquen múltiples necesidades de acceso a variables de campo, por ejemplo datos de procesos periódicos, sistemas de alarmas, y datos por batches, pueden ser implementados juntos dentro de un servidor OPC UA.

Habiéndose decidido en qué perfil se realizaría la aplicación se definió cómo se implementó el desarrollo, para lo que se analizaron tres casos posibles. El caso más sencillo que se podría presentar es contar con un producto OPC legado basado en DCOM, que se desee migrar a la nueva especificación UA, para esto lo más fácil es usar un software que sirva de puente entre las dos tecnologías. Para el caso de los servidores, este software se conecta con el servidor legado mediante la tecnología DCOM, y brinda los mismos servicios disponibles convertidos en la especificación UA. La misma solución aplica a los clientes, quienes de igual forma acceden a un intermediario mediante DCOM pero este a su vez convierte las solicitudes en formato UA para poder acceder a un servidor del nuevo estándar. Hay productos comerciales que permiten esto, con muy poca o ninguna programación; son parametrizables en vez de programados y sus precios son accesibles.

El segundo caso que se puede presentar, es el reescribir el programa en la nueva especificación UA, implementando solo los servicios ofrecidos previamente en la versión DCOM. Si se utiliza un SDK puesto a disposición por algún fabricante, la implementación no debería ser problemática, debido a que solo se construirían unas cuantas funcionalidades del servidor OPC de arquitectura unificada, y no todas, mientras que el SDK

encapsula los detalles técnicos. Esta opción sería buena cuando se quiere un primer producto con las funcionalidades a las que el mercado se encuentra acostumbrado, dejando para versiones posteriores las nuevas funcionalidades que ofrece la nueva tecnología. La tercera opción es la generación de un producto nuevo con las funcionalidades ofrecidas por DCOM, y con las nuevas como la publicación de los métodos del servidor, o la publicaciones de datos complejos, ya sea porque se contaba con una aplicación legada que se pretende actualizar, o porque se desee realizar un producto nuevo con proyección hacia el futuro utilizando el nuevo estándar.

El uso de un SDK, facilita significativamente el desarrollo de cualquier producto OPC UA o OPC clásicos, de tal forma que el programador solo tiene que concentrarse en la lógica de su implementación delegando la gestión de las minucias técnicas al propio SDK. A menudo estos paquetes de software también traen asistentes que mediante menús, y de acuerdo a las opciones fijadas, se pre-configura la generación de plantillas de código en el cual se puede trabajar. Para el caso del servidor este debe enlazar el acceso a datos de campo con la publicación estándar de los mismos construyendo un adecuado espacio de dominio. Para el caso de desarrollar un cliente, los SDKs se enfocan en la presentación de la información, restricciones de accesos, asistentes, e interfaces de usuarios.

Como se ha mencionado la OPC Foundation suministra gran cantidad de información y herramientas, y por tanto su SDK ofrecido es una alternativa seria en una implementación de OPC UA, suministrado mediante la adquisición de una membresía paga con la OPC Foundation. Sin embargo la intención de este SDK no es de tipo comercial, siendo más bien demostrativa o de apoyo a las empresas que efectivamente crean productos comerciales, generando SDKs con mayores facilidades de uso, y con entornos de desarrollo simplificados para la generación de clientes y servidores OPC.

Por todo lo anterior, en el desarrollo de la investigación se pretendió trabajar y utilizar el estándar OPC UA, concretamente en el perfil DA, por ser el más utilizado en la vieja tecnología DCOM. Sin embargo y debido a que el desarrollo de aplicaciones SDKs bajo el nuevo estándar apenas comienza, solo se pudieron llegar a realizar pruebas con este a través de las librerías de desarrollo suministradas por la OPC Foundation, por lo que no se hizo la implementación del demostrativo en OPC UA.

Para el momento de la implementación del prototipo académico (último trimestre 2009 y primer mes 2010) [3] [13], para OPC UA estaba solo disponible el SDK de la empresa Techno-Software AG [14] [15], solo soportando las funcionalidades básicas y el perfil DA. Las empresas Softing [16], y Prosys [17] anunciaban que lanzarían sus respectivas versiones. Habiendo escogido que el desarrollo se iba a realizar con un SDK comercial y a pesar que para el momento en el que se desarrolló el proyecto no se contaba con muchas alternativas que permitieran esto, se pretendía tener las funcionalidades ofrecidas por UA, por lo que la única alternativa fue utilizar la especificación OPC XML-DA, escogiendo en definitiva el OPC Toolbox para Windows PC-XMLDA-SDK-WI, de la empresa Softing, la cual está a mitad de camino entre las especificaciones clásicas y la nueva OPC UA. Las razones de esta escogencia fueron que esta solución está implementada con la misma arquitectura conceptual, basada en clientes/servidores tipo web, mediante SOAP⁹, a que el fabricante en su SDK lo hizo compatible con la futura versión de su toolbox con soporte para OPC UA, por lo que la migración sería trivial, y por último debido a

que la documentación y el entorno de trabajo de este fabricante fue el más completo y amigable entre los evaluados.

CONCLUSIONES

Después de lo presentado se concluye que es posible crear con pocos recursos en países en vía de desarrollo, clientes y servidores OPC, en el estándar UA con perfil DA, para el desarrollo local de aplicaciones propietarias de alta calidad, con la finalidad de acceder a variables de campo de sistemas embebidos propietarios, o genéricos como son los autómatas programables. Mediante el uso de SDKs comerciales se simplifica significativamente el desarrollo de estas aplicaciones; al sopesar los recursos necesarios para realizar una aplicación desde cero, o con un SDK ofrecido por la OPC Foundation, versus la reutilización de código mediante el uso de SDKs comerciales, asumiendo los costos adicionales de licenciamiento, la balanza se inclina por los productos comerciales. Con estos se obtienen años de experiencia y desarrollo, con la información pertinente, dejando solo al usuario el desarrollo propio del protocolo de comunicaciones de acceso a la red de campo de dispositivos. Adicionalmente las pruebas demostraron que es posible migrar aplicaciones entre sistemas operativos como Windows y Linux (antes imposible), sin tener que reescribir todo el código, solo viéndose afectada la interface gráfica de usuario. Así mismo se comprobaron las ventajas de la nueva plataforma pudiendo acceder de forma remota vía redes LAN e Internet a variables de procesos, mediante el acceso a un computador local al proceso de prueba.

9. De sus siglas en inglés para Simple Object Access Protocolo, protocolo simple de acceso a objeto.



BIBLIOGRAFÍA

- [1] (2009 6). OPC XML-DA. [En línea] Disponible: <http://www.opcconnect.com/xml.php>
- [2] (2009 6). Free Stuff - OPC UA. [En línea] Disponible: <http://www.opcconnect.com/uafree.php>
- [3] OPC Foundation. (2009 04). Portal OPC Foundation. [En línea] Disponible: <http://www.opcfoundation.org/>
- [4] BOOCH, G.; JACOBSON, I.; RUMBAUGH, J. El proceso unificado de desarrollo de software. Madrid: Person Educación S.A., 2000.
- [5] KNIBERG, H. Scrum y XP desde las trincheras, como hacemos Scrum. Estados Unidos, C4Media Inc., 2007.
- [6] LANGE, J.; IWANITZ, F. *OPC, Fundamentals, implementation and application*. 3rd Rev. Hüthig GmbH and Co. KG Heidelberg, 2006.
- [7] MAHNKE, W.; LEITNER, S. H.; DAMM, M. *OPC unified architecture*. Berlin: Springer-Verlag Berlin Heidelberg, 2009, Capítulo 14.
- [8] SÁNCHEZ, R. D. *Controlador lógico programable, una mirada interna*. Barranquilla: Corporación Universitaria de la Costa, 2009.
- [10] (2009 6). History of OPC. [En línea] Disponible: <http://www.opcconnect.com/history.php>
- [11] (2009 6). OPC Foundation Downloads. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&SEI=ModificationTime&FVI=All&FTI=&SE2=ReleaseStatus&FV2=All&FT2=>
- [12] (2009 6). OPC Unified Architecture, Introduction. [En línea] Disponible: <http://www.opcconnect.com/ua.php>
- [13] (2009 9). OPC UA SDKs and Toolkits. [En línea] Disponible: <http://www.opcconnect.com/ua-kit.php>
- [14] (2009 10). OPC UA: Introduction. [En línea] Disponible: <http://www.technosoftware.com/opc-ua>
- [15] (2009 10). Technosoftware: Evaluations. [En línea] Disponible: <http://www.technosoftware.com/evaluations>
- [16] (2009 10). OPC Toolbox Unified Architecture. [En línea] Disponible: <http://www.softing.com/home/en/industrial-automation/products/opc/toolkits/toolbox-unified-architecture/>
- [17] (2009 10). Prosys OPC UA Java SDK. [En línea] Disponible: <http://www.prosysopc.com/opc-ua-sdk.php>

- [18] Randy Armstrong (2009 8). A Roadmap to UA Downloads. [En línea] Disponible: <http://lists.opcfoundation.org/RandyBlog/archive/2009/08/31/a-roadmap-to-ua-downloads.aspx>
- [19] OPC Foundation. (2009 2). OPC Unified Architecture, Part 1: Overview and Concepts. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>
- [20] OPC Foundation. (2009 2). OPC Unified Architecture, Part 3 :Address Space Model. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>
- [21] OPC Foundation. (2009 2). OPC Unified Architecture, Part 4 :Services. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>
- [22] OPC Foundation. (2009 2). OPC Unified Architecture, Part 6 :Mappings. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>
- [23] OPC Foundation. (2009 2). OPC Unified Architecture, Part 7 :Profiles. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>
- [24] OPC Foundation. (2009 2). OPC Unified Architecture, Part 8 :Data Access. [En línea] Disponible: <http://www.opcfoundation.org/Downloads.aspx?CM=I&CN=KEY&CI=283>