

# GESTIÓN E IMPLEMENTACIÓN DE UNA PASARELA HTTP-OPC APLICADO A UNA PLANTA SOLAR

Beatriz Pontes

Dto. Ingeniería de Sistemas y Automática, Universidad de Sevilla, [bepontes@cartuja.us.es](mailto:bepontes@cartuja.us.es)

Alfonso Cepeda

Dto. Ingeniería de Sistemas y Automática, Universidad de Sevilla, [cepeda@us.es](mailto:cepeda@us.es)

Eduardo F. Camacho

Dto. Ingeniería de Sistemas y Automática, Universidad de Sevilla, [eduardo@esi.us.es](mailto:eduardo@esi.us.es)

## Resumen

*El propósito de este artículo es detallar cómo conseguir un acceso eficiente a datos provenientes de un dispositivo genérico o planta industrial situada dentro de una red local. Dicho dispositivo debe estar conectado a un servidor OPC que permita acceder a las distintas variables. La situación que se intenta resolver es aquella en la que solamente es posible acceder a las variables de un dispositivo cualquiera desde un equipo que esté necesariamente situado dentro de la misma red local. La solución abordada en este artículo se basa en la elaboración de una aplicación web que permita proporcionar el acceso a distintos usuarios tanto desde dentro como desde fuera de la red a la que pertenezca la planta. Esta solución permite también el control remoto ya sea de una planta industrial o cualquier otro tipo de dispositivo haciendo uso de OPC. En concreto, se mostrará una Planta Solar sobre la que se explicarán los aspectos de implementación.*

**Palabras clave:** Pasarela HTTP-OPC, OPC, XML, Supervisión, Control remoto.

alternativas, teniendo todas un punto en común, exigen la existencia de un equipo intermedio que esté situado dentro de la misma red local del dispositivo en cuestión y que sea accesible desde el exterior. Este equipo tiene una doble funcionalidad, por un lado actúa como cliente a la hora de acceder a los datos del dispositivo y por otro actúa como servidor de dichos datos hacia el exterior, constituyendo así una pasarela. Es importante tener en cuenta que si el equipo conectado directamente a la planta dispone de una dirección IP pública no es necesaria la instalación de este otro equipo intermedio (la pasarela estaría implementada en el mismo ordenador).

La estructura de este artículo es la siguiente: en el apartado 2 se realiza un análisis de las posibles tecnologías a utilizar y de la situación inicial de las plantas a las que se les puede aplicar este sistema de control. El apartado 3 explica cuál es el proceso del sistema de supervisión propuesto, siguiéndole en el apartado 4 la especificación de cómo realizar el control de una planta industrial general y en particular la Planta Solar situada en el Instituto de Automática de Andalucía, que se muestra en la Figura 1. Por último, el artículo finaliza con una sección de conclusiones referidas a su contenido.

## 1. INTRODUCCIÓN

El origen de este artículo nace de la necesidad de la supervisión y control remoto de dispositivos a través de Internet ([9]). La situación de partida es aquella en la que el dispositivo a controlar y el equipo al que se encuentra físicamente conectado estén situados dentro de la misma red local, no teniéndose acceso desde un equipo exterior a dicha red. Para conseguir que el acceso desde una red externa sea posible se pueden tomar varias

## 2. SITUACIÓN DE PARTIDA

Dentro de la industria del control, varias tecnologías han tenido mucho auge en los últimos tiempos. Cabe destacar los protocolos de comunicaciones CORBA, Java RMI, XML-RPC o SOAP y las soluciones basadas en web, entre las que se han analizado las posibles soluciones para afrontar el control remoto, teniendo en cuenta que la opción que se elija debe ser aquella que proporcione una mayor facilidad al usuario final



Figura 1: Planta Solar situada en el Instituto de Automática de Andalucía

sin repercutir en la eficiencia.

- En primer lugar, se podría optar por la utilización de CORBA, que es una arquitectura estándar para el desarrollo de aplicaciones distribuidas, pero que tiene el inconveniente de ser muy compleja, presentando problemas de implementación debido a la dificultad de sus especificaciones ([2]).

- También es importante mencionar el protocolo XML-RPC. Éste es un protocolo más sencillo de utilizar que CORBA que utiliza HTTP para la transmisión y está basado en XML para la codificación de los datos ([3]). La característica que hace que no sea apropiado para el acceso remoto es que necesita de la instalación de determinados módulos en el equipo del usuario.

- SOAP (*“Protocolo de acceso a objetos simple”*) es un lenguaje de mensajería basado también en XML. La principal diferencia entre SOAP y XML-RPC es su complejidad. SOAP está creado para dar un soporte más completo y minucioso, lo que lo hace ser más complejo y a la vez más potente ([15]).

- Por otra parte, existe un mecanismo ofrecido en Java llamado RMI (*“Remote Method Invocation”*), que permite que un procedimiento pueda ser invocado remotamente ([4]). Presenta la gran ventaja de ser independiente de la arquitectura por la propia tecnología Java, y además es posible la elaboración de un applet de Java que facilite su uso por parte del usuario.

- Finalmente, es posible contar tan sólo con el uso del protocolo HTTP. La solución consiste en la creación de una aplicación web que permita la supervisión y el control de la planta, teniendo así el usuario la posibilidad de realizar estas tareas mediante un navegador web, eliminando posibles problemas de interoperabilidad e incompatibilidad así como la necesidad de instalar un software específico para el acceso. Aunque estas ventajas son compartidas con la utilización de Java RMI junto con applets, resulta más sencilla la elaboración de una aplicación web.

Esta última opción, que es la que se ha elegido para la elaboración del proyecto base de este artículo, tiene también la ventaja de que, al utilizar el protocolo HTTP, el acceso no estará restringido por ningún cortafuegos convencional en Internet.

El proceso de elaboración es el siguiente, en un ordenador de la red local, que de ser accesible desde una red externa puede ser el mismo donde esté la conexión física con la planta, se instala un servidor web que soporte algún lenguaje de programación. En este servidor se realiza un programa que por un lado acceda a la planta que se quiera controlar y por otro proporcione una página web dinámica que será enviada por el servidor web. Así se consigue poder realizar tareas tales como mostrar los valores reales de la planta, y a través de peticiones GET o POST, modificar parámetros de control de la planta.

Para poder realizar todo esto es necesario disponer de los siguientes elementos:

- **Planta industrial real a controlar:** En este caso se trata de una Planta Solar que se utiliza para climatización. Dicha planta estará conectada a un equipo con el hardware y software apropiado para adquirir datos directamente de ella. También se debe disponer del software específico de control avanzado y optimización multivariable ([12]). Se puede encontrar una descripción de la Planta Solar en ([8]).

- **Equipo conectado al dispositivo:** Es el equipo que se encuentra conectado directamente al dispositivo que se va a supervisar y controlar, se encuentra situado dentro de la red local y debe tener instalado un servidor OPC para poder servir los datos provenientes de dicho dispositivo. En el caso de que su dirección IP sea privada, y por tanto solamente sea accesible desde dentro de la misma red local, será necesario contar con otro

equipo que permita el acceso desde el exterior.

- **Equipo accesible desde el exterior:** Equipo para el alojamiento de la aplicación web, situado en la misma red local que el equipo conectado directamente al dispositivo. Este equipo debe tener instalado un servidor de Internet que de soporte a la aplicación y debe disponer de una dirección IP pública para poder acceder a él a través de Internet con una dirección específica. En caso de que no sea necesaria la instalación de este equipo, todos sus requisitos deberán estar situados en el equipo conectado directamente a la planta a controlar.

- **Cliente OPC:** Para permitir que la aplicación web acceda a los datos de la planta industrial es necesario disponer de un cliente OPC que esté implementado en forma de un ActiveX. En el caso de la Planta Solar, se ha desarrollado dicho ActiveX a partir de un conjunto de librerías en C++ de las que ya se disponía con anterioridad.

Una vez que se han analizado las distintas tecnologías que se podrían haber utilizado y los elementos necesarios para poder llevar a cabo el acceso remoto a un dispositivo genérico, en los siguientes apartados se especificará de qué manera es posible realizar la supervisión y el control de una planta industrial.

### 3. DESARROLLO

En primer lugar, se presenta en la Figura 2 un esquema general de cuáles son las tareas involucradas en el proceso de supervisión o control desde un equipo cualquiera que accede a través de Internet.

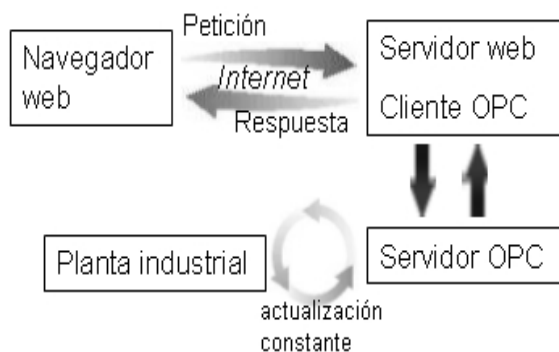


Figura 2: Esquema general de acceso

En dicha figura se puede observar que el proceso comienza cuando un usuario realiza una petición a un servidor de páginas web mediante una dirección específica por medio de Internet. En el servidor web se procesa la petición y se realiza un acceso mediante un cliente OPC a un servidor OPC situado en la misma red local (nótese que ambos procesos cliente y servidor OPC pueden residir en el mismo equipo). Una vez que el servidor OPC haya proporcionado los datos necesarios, el servidor web se encarga de enviar la respuesta al usuario que inició todo el proceso. De esta manera se realizan cada uno de los accesos ya sea para supervisión o control de la planta.

#### 3.1. CLIENTE OPC

El cliente OPC es un programa que debe desarrollarse en el equipo que es accesible desde el exterior. Su implementación está ligada a un control ActiveX del que se debe ya disponer (en el caso particular de la Planta Solar ha sido creado expresamente para este fin). Es posible utilizar cualquier lenguaje de programación que permita el uso de controles ActiveX, escribiendo las instrucciones necesarias para definir cuál es el servidor OPC al que se quiere realizar la conexión y cuál es su localización. Posteriormente es necesario invocar al método que realice la conexión al servidor OPC y añadir los ítems que se vayan a utilizar ([5], [14]). Estos ítems serán accedidos posteriormente ya sea para la petición de documentos XML o para la supervisión o control de la planta.

#### 3.2. PETICIÓN DE DOCUMENTOS XML

Aprovechando que el protocolo HTTP permite realizar la transferencia de ficheros en cualquier formato, es posible realizar un programa en el servidor web que envíe los valores actualizados de las variables de la planta a controlar en tiempo real y codificados en XML. La petición de este tipo de ficheros se realizará de la misma manera en que se realiza la petición de una página web, pero en este caso el código ejecutado en el equipo servidor detectará la petición devolviendo un archivo en formato XML, incluyendo los valores actuales de las variables, a las que accede mediante el cliente OPC ([11]). De esta manera cualquier programa local con capacidad para leer dicho formato puede elaborar análisis a partir de los datos obtenidos. La estructura del documento que se obtiene para el caso particular de la Planta Solar es el que se muestra a continuación:

```

<?xml version="1.0"?>
<plant>
<Variable name="SOLAR\RT13003_PV">108.632</Variable>
<Variable name="SOLAR\TT10114_PV">58.8</Variable>
<Variable name="SOLAR\PT11001_PV">4.246</Variable>
<Variable name="SOLAR\TT10113_PV">60.5</Variable>
<Variable name="SOLAR\TT10116_PV">58.3</Variable>
<Variable name="SOLAR\PT11002_PV">1.102</Variable>
</plant>

```

### 3.3. SUPERVISIÓN DE LA PLANTA SOLAR

La supervisión de los datos de un dispositivo permite visualizar las variables de dicho dispositivo de manera que éstas se actualicen constantemente, o una vez transcurrido un intervalo de tiempo determinado. La implementación a realizar está basada en un objeto JavaScript, denominado *XMLHttpRequest* ([1]), que permite hacer llamadas al equipo servidor para obtener documentos XML que contengan los valores actualizados. De esta manera es posible interpretar y mostrar los datos obtenidos sin necesidad de refrescar la página completa en cada acceso al servidor. Esta alternativa se ha escogido tanto por su sencillez como para reaprovechar recursos ya generados, ya que el documento XML que el servidor envía como respuesta formaba ya parte de uno de los requisitos del sistema. El inconveniente de la utilización del objeto *XMLHttpRequest* es que éste no está implementado por todos los navegadores, aunque sí suelen incluirlo las últimas versiones. Para evitar que usuarios que no dispongan de una versión actualizada de su navegador puedan realizar la supervisión es posible realizar la implementación de otro modo, el cual se explica más adelante.

El código ejecutado en el equipo servidor no varía de una implementación a otra. La diferencia se encuentra en que en el caso de la utilización del objeto *XMLHttpRequest*, dicho código solamente se ejecutará en el primer acceso por parte del usuario. Para el resto de los accesos producidos para la actualización de los datos se realizarán peticiones de ficheros XML que serán interpretados en el equipo del usuario, tal y como puede verse en la Figura 3. El código ejecutado en el equipo servidor cuando se realizan peticiones de este tipo se especifica en el apartado 3.2: *Petición de documentos XML*.

Por el contrario, cuando la supervisión se realiza por parte de un usuario que disponga de un navegador que no soporte estas características, el código que se ejecuta en el servidor web es siempre el mismo en cada uno de los accesos, ya sea la

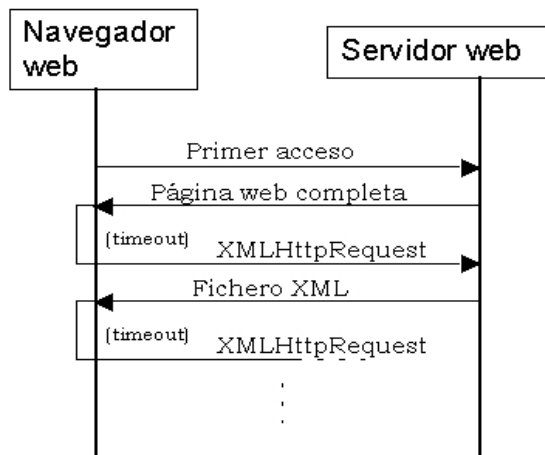


Figura 3: Diagrama de supervisión con utilizando el objeto *XMLHttpRequest*

primera vez o cualquier acceso de actualización.

Además de implementar el código necesario en el servidor web, es necesario escribir también algunas líneas de código que sean ejecutadas en el equipo del usuario. Este código permitirá realizar la petición e interpretar el documento XML que se recibe en cada actualización de los datos.

El modo de operación en cada una de estas peticiones es síncrono, esto quiere decir que cuando el cliente realiza una petición queda en espera de una respuesta que seguidamente analizará. En primer lugar, cuando se carga la página web en el equipo del cliente se realiza una llamada a un método inicial, que establece un periodo de tiempo para realizar una determinada acción cada vez que transcurra dicho periodo.

En este caso, la acción a realizar es una petición del fichero XML que contiene los datos actualizados de la planta. Una vez que dicho fichero sea recibido, se invocará al método encargado de interpretar el documento y mostrar los nuevos valores por pantalla.

Como se ha comentado antes, pueden existir algunas versiones de navegadores que no soporten estas características, por lo que se ha realizado otra implementación basada en la utilización del método *“meta refresh”*, que permite que una página web se recargue cada vez que transcurra un cierto tiempo especificado. Esto significa que cada vez que se refresca la página, el equipo del cliente realiza una nueva petición de la misma al equipo servidor. Éste método va incluido en la cabecera del código HTML de la página en cuestión.

Al realizar una nueva petición de la página, el servidor la reenvía como si el acceso se hiciera por primera vez, con todos sus controles y componentes (ver Figura 4), lo que significa que no solamente se actualizan los datos de la planta en cada refresco como en el caso anterior, y que se pueden producir mayores tiempos de respuesta, dependiendo de la velocidad de conexión del equipo del usuario.

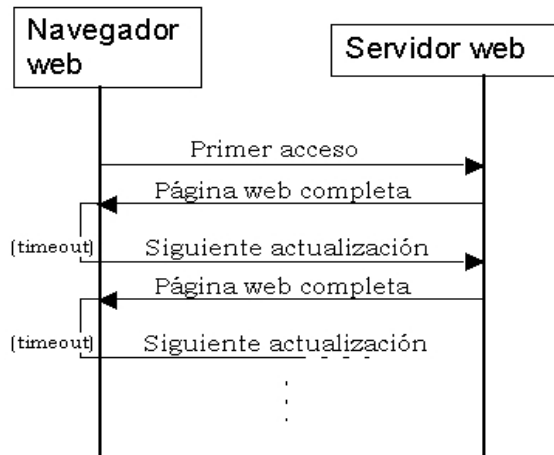


Figura 4: Diagrama de supervisión sin la utilización de XMLHttpRequest

Para el caso particular de la Planta Solar, se muestra a continuación la apariencia final de la aplicación web desarrollada en modo de supervisión. En la Figura 5 puede verse unas etiquetas de fondo azul en las que los valores de todas las variables se irán actualizando constantemente.

#### 4. CONTROL DE LA PLANTA SOLAR

Este sistema de control se ha aplicado a una Planta Solar gracias a que, a través de una petición HTTP, se pueden modificar valores de variables de control ([6],[7],[10]).

La modificación de variables se lleva a cabo de una manera muy sencilla gracias al control ActiveX que se utiliza como cliente OPC, tan sólo es necesario realizar una llamada a un método específico del control. Antes de proceder a la escritura del valor es necesario asegurarse que el cliente OPC se encuentra conectado al servidor OPC, así como de que los ítems estén

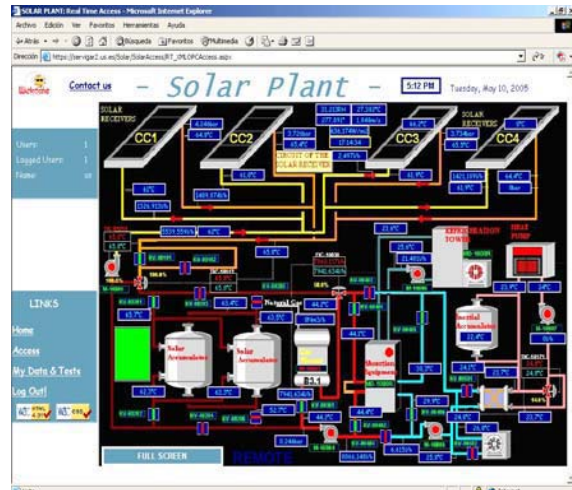


Figura 5: Supervisión de la Planta Solar

correctamente añadidos y tengan permiso de escritura en el servidor OPC. Es importante tener en cuenta que las características de implementación pueden variar según los controles ActiveX que se utilicen. Así, mientras que con el que ha sido desarrollado para el caso de la Planta Solar es posible modificar el valor de una variable con una sola instrucción, si se utilizan otro tipo de controles ActiveX puede ser necesario también realizar alguna instrucción de actualización para que los cambios producidos sean permanentes.

#### 5. CONCLUSIONES

A lo largo de este artículo se ha especificado cómo es posible realizar el acceso, ya sea para supervisión o control, a determinados valores pertenecientes a un dispositivo genérico, y en concreto a una Planta Solar. Como se ha expuesto, dicho acceso puede producirse desde un equipo cualquiera con conexión a Internet. Esto supone un gran avance respecto a la situación anterior en la que solamente se permitía el acceso a la Planta Solar desde equipos situados dentro de la misma red local. Asimismo, el control ActiveX desarrollado en el entorno de este proyecto puede ser utilizado como cliente OPC para acceder a cualquier otro dispositivo con un servidor OPC asociado.

#### Referencias

[1] Análisis del objeto XML Extras: XMLHttpRequest.

- <http://webfx.eae.net/dhtml/xmlextras/xmlextras.html>.
- [2] *CORBA Website*. <http://www.corba.org>.
  - [3] *Definición y estudio del protocolo XML-RPC*. <http://www.xmlrpc.com>.
  - [4] *The Java Remote Method Invocation system*. <http://java.sun.com/docs/books/tutorial/rmi/>.
  - [5] *Página web oficial OPC Foundation*. <http://opcfoundation.org>.
  - [6] Mastascusa Aburdene, M.F. A proposal for a remotely shared control systems laboratory. pages 589–592. Proceedings of the ASEE 1991 Frontiers in Education Conference, Session 24A3, 1991.
  - [7] C.A. Bohus Aktan, B. Distance learning applied to control engineering laboratories. volume 39(3), pages 320–326. IEEE Transactions on Education, 1996.
  - [8] E.F. Camacho and C. Bordons. *Model Predictive Control*. Springer, 2004.
  - [9] Marcos D. Reguera P. González J.J. Blázquez L.F. Domínguez, M. Connection pilot plant to the internet. IFACT Internet Based Control Education IBCE01, Madrid, España., 2001.
  - [10] S. Dormido. Control learning: present and future. 15th IFAC World Congress. Barcelona., 2002.
  - [11] Michael Floyd. *Building Web Sites with XML*. Prentice Hall, 2000.
  - [12] Daniel Gasa Fonollosa. Laboratorio remoto de automática: plantas de variable discreta. XXIII Jornadas de Automática, Leon.
  - [13] J. Henry. Controls laboratory teaching via the world wide web. In proceedings of the ASEE Annual Conference. Washington, USA, Session 3513.
  - [14] Marlon Martínez. OPC: “OLÉ for process control”. 2003.
  - [15] Aaron Skonnard. Understanding SOAP. 2003.
  - [16] J. Sánchez, F. Morilla, and S. Dormido. Teleoperation of an inverted pendulum through the world wide web. pages 12–14. in workshop on internet based control education. IBCE’01, Madrid, Spain, 2001.