



OPC: ¿De qué se trata, y cómo funciona?

“Guía para entender la Tecnología OPC”

Darek Kominek, P. Eng. Alberta, Canada - 2009

Resumen Ejecutivo

Este artículo sobre Tecnología OPC es una sencilla introducción al estándar abierto de conectividad industrial más popular. En él se introduce la idea clave que se esconde tras OPC, muestra porqué OPC es diferente a los protocolos de comunicación convencionales (a menudo propietarios), y expone cómo OPC ayuda a superar las limitaciones de dichos protocolos nativos.

A continuación, se explica cuáles son los bloques fundamentales de OPC (Clientes OPC y Servidores OPC) y cómo trabajan juntos para hacer posible que se puedan compartir datos permitiendo una interoperabilidad sin límites. Utilizando ilustraciones claras y la mínima jerga técnica, esta guía ayuda a lectores de todos los niveles técnicos a conocer qué es la Tecnología OPC y cómo la pueden utilizar en beneficio propio dentro de sus empresas y proyectos.

CÓMO SOLUCIONA OPC EL PROBLEMA DE LA CONECTIVIDAD

Hoy en día la automatización se utiliza de modo generalizado en todas las principales industrias. Mientras distintas industrias utilizan distintos dispositivos especializados, sistemas de control y aplicaciones, todas comparten un reto común, que crece rápidamente – cómo compartir información entre todos estos componentes y el resto de la empresa.

Antes de introducir qué es OPC y cómo resuelve una de las mayores pesadillas de la automatización, vale la pena analizar cuáles eran los problemas anteriormente. A continuación, se enumeran los factores que tradicionalmente causaban los mayores problemas al compartir información, junto a una breve explicación del impacto que ha tenido OPC sobre cada uno de ellos:

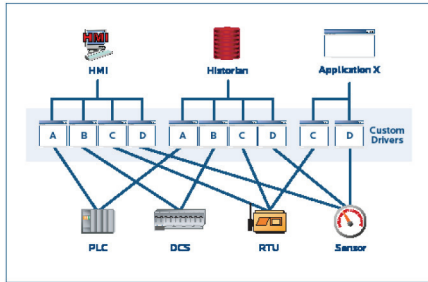


Figura 1: Problema de controladores personalizados - Cada aplicación requiere un dispositivo o un controlador de protocolo específico que le permita comunicarse con los respectivos dispositivo. Los drivers de comunicación no son reutilizables entre aplicaciones, ya que cada aplicación utiliza su propio formato de datos.

Protocolos propietarios: Los fabricantes utilizaban frecuentemente protocolos que permitían a productos de una determinada gama comunicarse entre ellos, pero requerían protocolos personalizados para comunicar con productos de otros fabricantes. Para empeorar las cosas, distintas gamas de productos del mismo fabricante frecuentemente no eran capaces de comunicar entre sí, necesitando conectores adicionales.

OPC resuelve este problema haciendo innecesario que el cliente de Datos tenga que conocer cómo comunica el servidor de Datos o cómo organiza dichos datos.

Drivers de comunicación propietarios: Todas las conexiones punto a punto requerían un protocolo propietario para posibilitar la comunicación entre los extremos específicos. Por ejemplo, si un HMI necesitaba comunicar con un PLC, se requería de un driver en el HMI escrito específicamente para el protocolo utilizado por el PLC. Si los datos de este PLC necesitaban ser registrados además en un histórico de datos, el programa de registro de datos requería su propio driver porque el driver del HMI sólo se podía utilizar para el HMI y no para el software de registro histórico de datos (que necesitaría otro Driver propietario diferente). Si el driver específico para establecer la comunicación entre los dos extremos no estaba previamente desarrollado y disponible, las comunicaciones eran muy difíciles y caras de establecer.

OPC elimina la necesidad de disponer de drivers específicos entre cada aplicación y la Fuente de Datos. En la figura 1, un único protocolo estándar de PLC puede ser compartido simultáneamente por el HMI y la aplicación de registro de datos históricos mediante un conector OPC, con el beneficio añadido de que el conector OPC requiere una única conexión con el PLC, reduciendo así la carga del procesador.

Integración compleja: El uso habitual de protocolos propietarios para cada dispositivo significaba que, incluso con un pequeño número de dispositivos y aplicaciones, se requería rápidamente el uso de muchos protocolos. La misma aplicación HMI ejecutándose en múltiples ordenadores, comunicando con el mismo dispositivo, requería instalar y configurar múltiples instancias del mismo protocolo en cada ordenador. Si las aplicaciones HMI comunicaban a su vez con dispositivos adicionales, cada HMI requería su propio conjunto de drivers para cada uno de los dispositivos. El mantenimiento de las versiones de las aplicaciones se convertía en una pesadilla.

“OPC elimina la necesidad de disponer de drivers específicos entre cada aplicación y la Fuente de Datos”

“El tráfico y la carga de trabajo de los dispositivos se reduce enormemente utilizando comunicaciones OPC”

Utilizar OPC simplifica enormemente la integración porque, una vez que se configura un Servidor OPC para una Fuente de Datos en particular, todas las aplicaciones que utilizan OPC pueden empezar a compartir datos con esa Fuente de Datos, eliminando la necesidad de drivers adicionales.

Carga de trabajo en el dispositivo: Cada protocolo establece su propia conexión al dispositivo o controlador para el que está diseñado para comunicar. Dado el gran número de protocolos que se utilizan en una instalación típica, frecuentemente el controlador se veía bombardeado por múltiples peticiones de la misma información para cada aplicación que necesitaba comunicar con él. Además, la mayoría de dispositivos sólo pueden aceptar un número limitado de conexiones simultáneas. Si el número de drivers tratando de conectar a un dispositivo excedía el número máximo de conexiones, había que buscar soluciones adicionales para que las aplicaciones Cliente no se quedasen sin datos.

El tráfico y, por tanto, la carga de los dispositivos se reduce enormemente utilizando conectores OPC, porque un conector OPC específico para un dispositivo requiere una única conexión a la Fuente de Datos mientras que puede comunicar simultáneamente con múltiples aplicaciones para enviarles los Datos obtenidos.

Obsolescencia de infraestructuras antiguas: A medida que los fabricantes lanzan nuevos productos, eventualmente dejan de dar soporte a los antiguos. Cuando una nueva versión de HMI o SCADA ve la luz, es posible que requiera su propio juego de protocolos que, en ocasiones, dejan de soportar comunicaciones con dispositivos con los que la anterior versión de HMI o SCADA comunicaban.

OPC extiende la vida útil de sistemas antiguos porque, una vez que se ha configurado un Servidor OPC para el sistema, permite que cualquier aplicación Cliente que utilice OPC (la mayoría) pueda comunicar con el sistema antiguo, sin importar si la aplicación Cliente soporta o no de forma nativa la comunicación con dicho sistema antiguo. Por tanto, OPC permite que aplicaciones Cliente nuevas continúen comunicando con sistemas antiguos.

Conectividad corporativa: A medida que crece la necesidad de disponer de datos procedentes de la automatización en otros niveles de la empresa, los problemas de conexión se hacen más complejos, porque las aplicaciones empresariales no están diseñadas para comunicar con dispositivos y controladores. Esto puede añadir, potencialmente, una carga extra a la infraestructura de automatización y sumar preocupaciones adicionales de seguridad.

OPC hace posible de forma real que se puedan compartir datos provenientes de la automatización a lo largo de toda la red corporativa, permitiendo que aplicaciones validadas reciban datos con Fuentes de Datos de la red de automatización, eliminando la necesidad de instalar nuevos drivers de comunicación. Todo lo que se requiere es un Servidor OPC.

¿Hay algunos ejemplos reales de utilización de la Tecnología OPC en proyectos?

Sí. Para ver algunos ejemplos en el mundo real de cómo OPC se ha utilizado para resolver diversos problemas de conectividad eche un vistazo a estos Casos de Estudio: (<http://www.matrikonopc.com/resources/case-studies.aspx>).

“OPC extiende la vida útil de sistemas antiguos... OPC permite a las aplicaciones mas nuevas poder seguir comunicándose con las aplicaciones más antiguas.”

INTRODUCCIÓN A OPC

¿Qué es OPC?

OPC es el método de conectividad de datos basado en los estándares más populares del mundo. Es utilizado para responder a uno de los mayores retos de la industria de la automatización: cómo comunicar dispositivos, controladores y/o aplicaciones sin caer en los problemas habituales de las conexiones basadas en protocolos propietarios.

OPC no es un protocolo, sino más bien un estándar para la conectividad de datos que se basa en una serie de especificaciones OPC gestionadas por la OPC Foundation. Cualquier software que sea compatible con estas especificaciones OPC proporciona a usuarios e integradores conectividad abierta e independiente tanto del fabricante del dispositivo como del desarrollador de la aplicación Cliente.

¿Por qué triunfa OPC donde los protocolos propietarios fallan?

La clave del éxito de OPC en crear comunicaciones auténticamente independientes del fabricante estriba en que OPC abstrae de los detalles de la implementación de Fuentes de Datos (i.e. PLC) y Clientes de Datos (i.e. HMI/SCADA), con lo que los datos se pueden intercambiar entre ellos sin que tengan que saber nada de sus respectivos protocolos de comunicación nativos y de la organización interna de sus datos. Esto, en clara oposición a la aproximación basada en crear aplicaciones basadas en protocolos propietarios que, por definición, son requeridos para comunicar, de forma nativa, la Fuente de Datos con el Cliente de Datos.

Cómo trabaja la comunicación OPC (conceptualmente)

Se puede representar como una capa de “abstracción” intermedia que se sitúa entre la Fuente de Datos y el Cliente de Datos, permitiéndoles intercambiar datos sin saber nada el uno del otro.

Cómo funciona OPC (Funcionalmente)

La “abstracción de dispositivo” OPC se consigue utilizando dos componentes OPC especializados llamados Cliente OPC y Servidor OPC. Cada uno de ellos es descrito en la siguiente sección. Es importante resaltar que el hecho de que la Fuente de Datos y el Cliente de Datos puedan comunicar entre sí mediante OPC no significa que sus respectivos protocolos nativos dejen de ser necesarios o hayan sido reemplazados por OPC. Al contrario, estos protocolos y/o interfaces nativos siguen existiendo, pero sólo comunican con uno de los dos componentes del software OPC. Y son los componentes OPC los que intercambian información entre sí, cerrando así el círculo. La información puede viajar de la aplicación al dispositivo sin que estos tengan que hablar directamente entre sí.

Beneficios de utilizar conectividad OPC

A primera vista, crear un driver propietario para dos componentes OPC (Cliente OPC y Servidor OPC) puede parecer que no sea una gran mejora, pero la experiencia ha demostrado que sí lo es. A continuación se enumeran algunos de los beneficios clave de utilizar OPC:

1. Una aplicación Cliente OPC puede comunicar libremente con cualquier Servidor OPC visible en la red sin la necesidad de utilizar ningún driver específico para la Fuente de Datos.
2. Las aplicaciones Cliente OPC pueden comunicar con tantos Servidores OPC como necesiten. No hay ninguna limitación inherente a OPC en el número de conexiones que se pueden establecer.
3. Hoy en día OPC está tan extendido que hay un Servidor OPC disponible para prácticamente todos los dispositivos nuevos o antiguos que existen en el mercado.

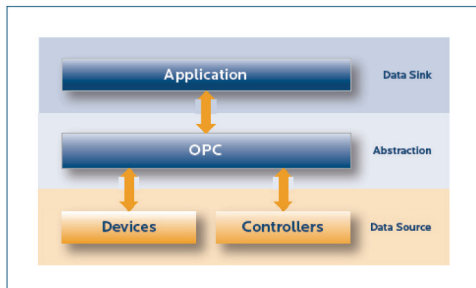


Figura 2: Los Servidores OPC se muestran como un nivel intermedio entre la fuente de datos y el cliente de datos, habilitando la intercomunicación sin que ningún lado conozca el protocolo nativo del otro.

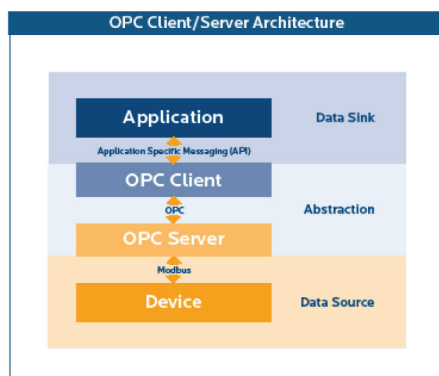


Figura 3: La arquitectura Cliente/Servidor OPC – Una mejor vista del funcionamiento OPC nos revela dos componentes: el Cliente OPC y el Servidor OPC. La especificación OPC define el mensaje entre estos dos componentes.

“OPC asume el reto de trabajar con estas distintas categorías de datos especificando de forma independiente cómo se va a transmitir cada uno de ellos.”

4. Las Fuentes de Datos (hardware o software) que utilizan OPC pueden ser intercambiadas o actualizadas sin la necesidad de actualizar los drivers utilizados por cada aplicación que comunique con ellas mediante OPC. Sólo hay que mantener actualizado el Servidor OPC para esa Fuente de Datos.
5. Los usuarios pueden elegir libremente los dispositivos, controladores y aplicaciones que mejor se ajusten a sus proyectos sin preocuparse del fabricante del que provienen o de si comunicarán entre sí... la intercomunicación se da por sentado.

¿Qué tipos de datos soporta OPC?

Los tipos de datos más comunes transferidos entre dispositivos, controladores y aplicaciones en automatización se pueden encuadrar en tres categorías:

- Datos de tiempo real
- Datos históricos
- Alarmas y Eventos

A su vez, cada una de las categorías anteriores soporta una amplia gama de tipos de datos. Estos tipos de datos pueden ser enteros, coma flotante, cadenas, fechas y distintos tipos de arrays, por mencionar algunos. OPC asume el reto de trabajar con estas distintas categorías de datos especificando de forma independiente cómo se va a transmitir cada uno de ellos a través de la arquitectura Cliente OPC - Servidor OPC.

Las tres especificaciones OPC que se corresponden con las tres categorías de datos son:

1. OPC Data Access Specification (OPC DA) – utilizada para transmitir datos de tiempo real
2. OPC Historical Data Access Specification (OPC HDA) – utilizada para transmitir datos históricos
3. OPC Alarms & Events Specification (OPC A&E) – utilizada para transmitir información de alarmas y eventos

¿Todos los Servidores OPC incluyen todas las especificaciones OPC?

No. No es obligatorio que todos los Servidores OPC incluyan todas las especificaciones OPC. Históricamente, la mayoría de los Servidores OPC sólo soportan datos de tiempo real (OPC DA). Es aconsejable investigar que especificaciones OPC incluye un Servidor OPC antes de elegirlo para un proyecto.

¿Es importante saber qué especificaciones OPC incluye un Cliente OPC o un Servidor OPC?

Sí, es crucial. Mientras que las tres especificaciones OPC (OPC DA, OPC HDA, OPC A&E) utilizan la misma arquitectura OPC cliente/servidor subyacente para transmitir las distintas categorías de datos, tanto Cliente OPC como Servidor OPC deben incluir la misma especificación OPC para coordinar de forma correcta el flujo de datos entre ellos y para trabajar correctamente con la Fuente de Datos y el Cliente de Datos respectivamente.

“Es aconsejable investigar que especificaciones OPC incluye un Servidor OPC antes de elegirlo para un proyecto.”

SERVIDORES OPC

¿Qué es un Servidor OPC?

Un Servidor OPC es una aplicación de software. Un driver “estandarizado” desarrollado específicamente para cumplir con una o más especificaciones OPC.

La palabra “Server” en “OPC Server” no hace referencia en absoluto al ordenador donde este software se estará ejecutando. Hace referencia a la relación con el Cliente OPC.

¿Qué hacen los servidores OPC?

Los Servidores OPC son conectores que se pueden asimilar a traductores entre el mundo OPC y los protocolos nativos de una Fuente de Datos.

OPC es bidireccional, esto es, los Servidores OPC pueden leer de y escribir en una Fuente de Datos. La relación Servidor OPC/Cliente OPC es de tipo maestro/esclavo, lo que significa que un Servidor OPC sólo transferirá datos de/a una Fuente de Datos si un Cliente OPC así se lo pide.

“Los servidores OPC pueden comunicarse prácticamente con cualquier fuente de datos.”

¿Con qué tipos de Fuentes de Datos puede comunicar un Servidor OPC?

Los Servidores OPC pueden comunicar prácticamente con cualquier Fuente de Datos cuyos datos puedan ser leídos o escritos por medios electrónicos. Una breve lista de posibles Fuentes de Datos incluye: dispositivos, PLCs, DCSs, RTUs, instrumentos de medición, bases de datos, historiadores, software de cualquier tipo (i.e. Excel), páginas web e incluso archivos CSV (texto separado por comas) de actualización automática. Para comunicar con cualquiera de estos dispositivos se requiere únicamente el uso de un Servidor OPC que utilice el protocolo o interfaz nativo apropiado. Una vez que se ha configurado dicho Servidor OPC, cualquier aplicación Cliente que utilice OPC (y tenga los permisos adecuados) puede empezar a comunicar con la Fuente de Datos sin que importe la forma en que esta comunica de forma nativa.

¿Dónde puedo encontrar un servidor OPC para el dispositivo X?

Mientras que muchos fabricantes ofrecen servidores OPC con sus dispositivos, controladores y aplicaciones, hay muchos que no lo hacen. MatrikonOPC es el mayor proveedor del mundo de software OPC, tiene Servidores OPC para cientos de dispositivos. Un buen lugar para comenzar es en la página web www.matrikonopc.es o llamando a la oficina de MatrikonOPC directamente.

¿Cómo trabaja un Servidor OPC?

Aunque los usuarios no necesitan saber nada a cerca de los entresijos de los Servidores OPC para poder utilizarlos, una mirada a lo que ocurre tras las bambalinas puede arrojar luz al porqué puede variar mucho la calidad y el rendimiento de Servidores OPC de distintos suministradores.

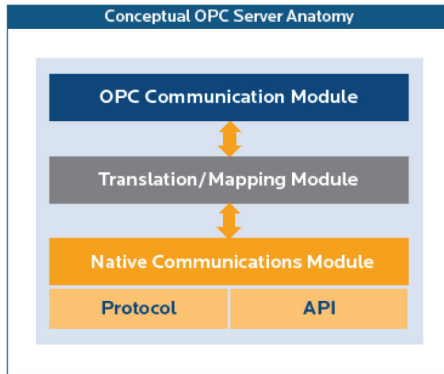


Figura 4: Anatomía conceptual de un Servidor OPC - Conceptualmente, un servidor OPC se puede desglosar en tres módulos: módulo de comunicaciones OPC, módulo de traducción / mapeado, y un módulo de comunicaciones nativas.

Un vistazo conceptual de cómo funciona un Servidor OPC puede ser el siguiente:

- **Módulo de comunicaciones OPC:** Esta es la parte del Servidor OPC responsable de comunicar adecuadamente con un Cliente OPC. Los Servidores OPC bien diseñados deben ser plenamente compatibles con las especificaciones OPC que implementen, para asegurar que comunican correctamente con cualquier Cliente OPC.
- **Módulo de comunicaciones nativas** El Servidor OPC debe emplear el método de comunicación más eficiente con la Fuente de Datos. En algunos casos, esto implica comunicar con la Fuente mediante su protocolo propietario de datos, mientras que en otros casos, esto significa comunicar a través de una Interfaz de Programación de la Aplicación (API). Típicamente, cuanto más experiencia tenga el desarrollador del Servidor OPC con el dispositivo, mejor utilizará las posibilidades de comunicación que ofrece el dispositivo.
- **Módulo de traducción/mapeado:** Aquí es donde sucede toda la "magia" de un Servidor OPC. La función de este módulo es interpretar de forma adecuada las peticiones OPC entrantes de un Cliente OPC, convirtiéndolas en peticiones nativas que se envían a la Fuente de Datos y viceversa. Si esto se hace eficientemente, se puede mantener al mínimo la carga sobre la Fuente de Datos mientras se maximiza la capacidad de transmisión de datos.

¿Puede comunicar un Cliente OPC de un determinado fabricante con Servidores OPC de otros fabricantes?

Sí, siempre que tanto el Cliente OPC como el Servidor OPC cumplan con las mismas especificaciones OPC, deben ser capaces de comunicar entre sí, independientemente de qué suministrador vengan.

¿Los Servidores OPC pueden compartir datos con otros Servidores OPC?

Los Servidores OPC no se comunican directamente unos con otros; sólo están diseñados para comunicar con Clientes OPC. Sin embargo, existen utilidades OPC diseñadas específicamente para que ésta comunicación entre Servidores OPC sea posible (y fácil). Ver MatrikonOPC Data Manager (<http://www.matrikonopc.com/products/opc-data-management/opc-data-manager.aspx>), designed to specifically make this OPC Server-to-OPC Server communication trivial.

CLIENTES OPC

¿Qué es un Cliente OPC?

Un Cliente OPC es una pieza de software creada para comunicar con Servidores OPC. Utiliza mensajería definida por una especificación concreta de la OPC Foundation.

“Los Servidores OPC bien diseñados deben ser plenamente compatibles con las especificaciones OPC que implementen”

¿Qué hace un Cliente OPC?

Conceptualmente Un Cliente OPC representa un destino de datos. Inician y controlan la comunicación con Servidores OPC basados en las peticiones recibidas desde la aplicación en la que están embebidos. Los Clientes OPC traducen las peticiones de comunicación provenientes de una aplicación dada en la petición OPC equivalente y la envían al Servidor OPC adecuado para que la procese. A cambio, cuando los datos OPC vuelven del Servidor OPC, el Cliente OPC los traduce al formato nativo de la aplicación para que ésta pueda trabajar de forma adecuada con los datos.

Técnicamente: Los Clientes OPC son módulos de software utilizados por una aplicación para permitirle comunicarse con cualquier Servidor OPC compatible visible en la red. Típicamente, los Clientes OPC están embebidos en aplicaciones como HMIs, SCADAs, graficadores, Historiadores o generadores de informes, convirtiéndolos en aplicaciones compatibles OPC.

Es muy común referirse a la aplicación que contiene un Cliente OPC embebido como "Cliente OPC" a pesar de que sólo la parte que implementa OPC es el verdadero Cliente OPC.

¿Pueden comunicar un Cliente OPC con múltiples dispositivos (Servidores OPC) de forma simultánea?

La respuesta tiene dos partes:

1. Primero, semánticamente: Es importante recordar que los Clientes OPC, por su diseño, sólo son capaces de comunicar con Servidores OPC, no con los dispositivos finales. Esta aclaración es necesaria porque los Clientes OPC deben ser independientes de protocolos, puesto que de otra forma caerían en la trampa dispositivo-driver del pasado.
2. Sí, efectivamente los Clientes OPC pueden comunicar de forma simultánea con múltiples Servidores OPC. Esto significa que un Cliente OPC puede leer y escribir datos desde y hacia múltiples dispositivos (Fuentes de Datos) a través de sus respectivos Servidores OPC.

¿Cómo trabaja un Cliente OPC?

Como se hizo con los Servidores OPC, un Cliente OPC se puede dividir conceptualmente en tres módulos:

Módulo de comunicaciones OPC: Aunque no tan involucrado como en el Servidor OPC (en los Servidores OPC esta parte es más compleja) es crucial para que el Cliente OPC se comporte como debe al conectarse a un Servidor OPC, intercambiar datos con él y desconectarse sin desestabilizar al Servidor OPC.

Módulo de comunicaciones con la aplicación: El Cliente OPC típicamente está diseñado para trabajar en una aplicación específica, por lo que, para permitir que la información pase de la aplicación al Servidor OPC pasando por el Cliente OPC, realiza una serie de llamadas al interfaz para la programación de la aplicación (API). También es posible que un Cliente OPC genérico comunique con una aplicación mediante un protocolo en lugar de con llamadas al API siempre que la aplicación soporte ese protocolo.

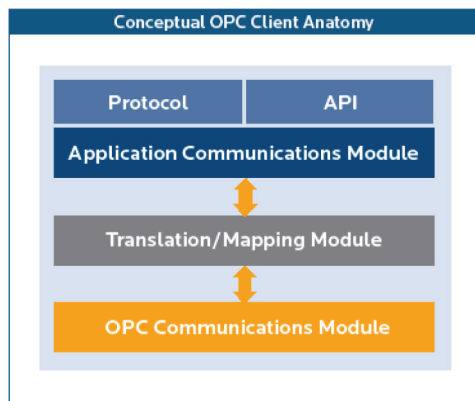


Figura 5: Anatomía conceptual cliente OPC - Reflejando un Servidor OPC, un Cliente OPC también puede ser considerado compuesto por tres módulos: Módulo de comunicaciones OPC, Módulo de comunicaciones con la aplicación y un Módulo de traducción/mapeado.

“Los Clientes OPC pueden comunicar de forma simultánea con múltiples Servidores OPC.”

“...no hay un límite teórico de a cuántos Servidores OPC puede conectar un Cliente OPC”

“La aproximación “plug-and-play” de OPC para la conectividad de datos le ha hecho ganar popularidad rápidamente”

Módulo de traducción/mapeado: Una de las funciones clave del Cliente OPC es la de traducir de forma bidireccional la información que su aplicación necesita leer de o escribir al dispositivo o Fuente de Datos.

¿Con cuántos Servidores OPC puede conectar un Cliente OPC?

La respuesta directa es: con todos aquellos con los que necesite. En la infraestructura OPC no hay un límite teórico de a cuántos Servidores OPC puede conectar un Cliente OPC.

¿Pueden los Clientes OPC comunicar directamente con otros Clientes OPC?

No. En OPC las comunicaciones Cliente-Cliente no están definidas. Sólo se soporta la arquitectura Cliente/Servidor. Por ello, si una aplicación debe proveer datos OPC a otros clientes, necesita tener su propio Servidor OPC. Este Servidor OPC permitirá a otros Clientes OPC de otras aplicaciones utilizar esta aplicación como Fuente de Datos.

¿Dónde está instalado el Cliente OPC?

Los Clientes OPC, típicamente, están embebidos en la aplicación que los utiliza, como por ejemplo HMIs o Historiadores. Si por alguna razón la aplicación que tenemos que utilizar no dispone de un Cliente OPC embebido, es posible que se pueda obtener uno externo del fabricante de la aplicación o de un tercero. Un Cliente OPC externo a la aplicación típicamente comunicara con ella a través de uno de sus protocolos nativos. En este caso, el Cliente OPC podría incluso no residir en el mismo ordenador que la aplicación.

CONCLUSIÓN

En este artículo se explica cómo la Tecnología OPC soluciona el reto creciente que tiene la industria moderna de acceder y compartir información entre dispositivos, controladores y aplicaciones, sin preocuparnos de cuál es su protocolo nativo de comunicaciones o quién es su fabricante. Se realiza una descripción a alto nivel de qué es OPC y se explica qué son y que papel juegan en la comunicación OPC los Servidores OPC y Clientes OPC (los bloques básicos de OPC). Mientras que no es necesario conocer en profundidad los mecanismos internos de OPC para poder utilizarlo, es beneficioso estar algo familiarizado con sus conceptos básicos de esta tecnología.

La aproximación “plug-and-play” de OPC para la conectividad de datos le ha hecho ganar popularidad rápidamente, convirtiéndolo en la tecnología para conectividad más utilizada en el mundo. Gracias a su versatilidad, la Tecnología OPC se utiliza en todos los niveles corporativos y en todas las industrias.

Copyright © Matrikon Inc 2010