

Una breve historia de la gestión de dispositivos

Parte 2: del lenguaje EDDL al concepto FDT/DTM.

Mirko Torres Contreras
Associated Technical Consultant en PITC/PICC

Phoenix Contact
www.phoenixcontact.com.ar

Las ventajas de las tecnologías abiertas

A primera vista, las tecnologías EDDL y FDT/DTM (del inglés, 'lenguaje de descripción de dispositivo electrónico' y 'herramienta de dispositivo de campo/administrador de tipos de dispositivos', respectivamente) parecen ser simplemente diferentes caras de la misma moneda. Pero ciertamente no es así. Como recordarán, mi primera experiencia con la tecnología FDT/DTM no fue ni especialmente reveladora ni sorprendente.

Entonces algo sucedió, comencé a trabajar en un proyecto temprano basado en Profibus PA que presentaba acopladores de segmento y acopladores de válvulas de Pepperl + Fuchs, dispositivos de campo Endress + Hauser, válvulas de control Samson y un sistema RIO para recopilar las señales que no se podían conectar a través de Profibus.

Y de repente me di cuenta de la gran idea que era este concepto FDT/DTM.

Nota del autor.
Quiero expresar mi agradecimiento a Maggie Carlson y Steve Biegacki, del FDT Group, por la colaboración y el apoyo en la preparación de este artículo.

Nota del autor.
Phoenix Contact patrocina este artículo. Las opiniones expuestas en este artículo son estrictamente personales. Toda la información requerida y empleada en este artículo es de conocimiento público.

Sobre el autor
Mirko Torrez Contreras es un consultor y entrenador especializado en automatización de procesos. Mirko ofrece consultoría en automatización de procesos, y consultoría y entrenamiento en redes industriales y protección contra explosiones. Está reconocido como Consultor Asociado en el Centro Internacional de Capacitación y Competencia de Profibus ubicado en Argentina (<http://profibus.com.ar/>). Además, presta servicios de escritura y traductorado técnicos (inglés o español).

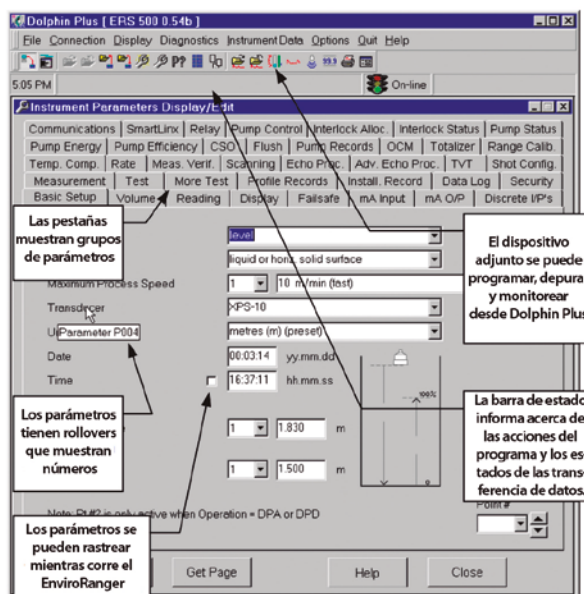


Figura 1. Captura de pantalla de un software de configuración de un antiguo transmisor de nivel

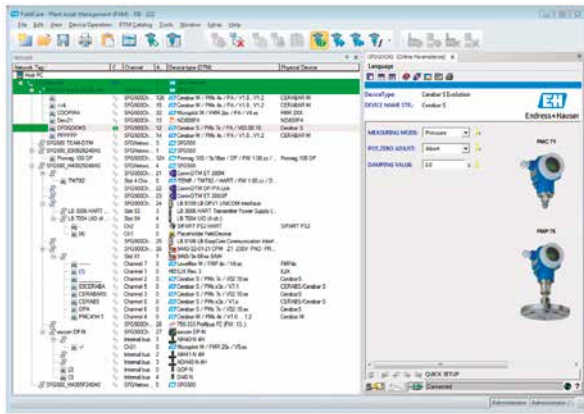


Figura 2. Representación jerárquica en forma de árbol de la implementación de FDT/DTM de una planta

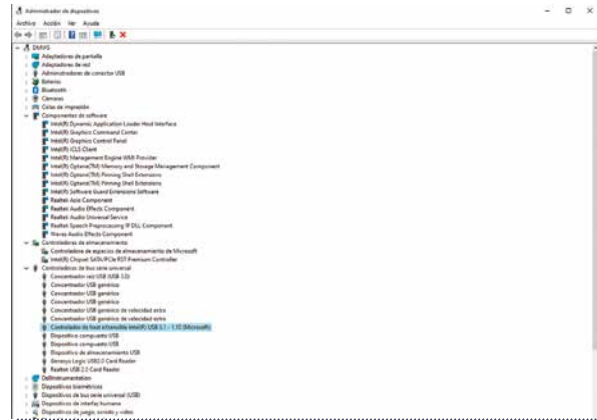


Figura 3. Administrador de dispositivos de Windows

La era anterior a FDT/DTM

Hasta la aparición de la tecnología FDT/DTM en 2001, las interfaces de usuario que se utilizaban en las herramientas de software para la parametrización y puesta en marcha de dispositivos de campo no eran fáciles de usar para el usuario final. De hecho, solían ser bastante crípticas. Habían sido desarrolladas por ingenieros, no por diseñadores de interfaz de usuario. Por lo tanto, las numerosas opciones y configuraciones que presentaban los dispositivos de campo inteligentes más nuevos generalmente estaban ocultas detrás de nuevas ventanas y pestañas.

Los gráficos eran bastante crudos. Y si se requería contar con un acceso centralizado a todos los dispositivos utilizados en una sola planta, realmente se necesitaba un DCS o una herramienta de gestión de activos del proveedor del sistema de control. Finalmente, este nivel de integración a menudo requería el uso de instrumentación inteligente de un solo proveedor.

Un administrador de dispositivos para su planta

Lo que la tecnología FDT/DTM permitía a los usuarios hacer era administrar los dispositivos, de una manera que antes era privilegio de los usuarios de DCS.

Proporcionaba una estructura jerárquica estandarizada, en forma de árbol único, que hacía posible navegar a través de los dispositivos de campo, sistemas RIO, acondicionadores de señal y dispositivos de diagnóstico de una manera que era familiar para cualquier usuario de PC: el concepto FDT/DTM hizo que la gestión de activos fuera tan fácil como usar el administrador de dispositivos de su PC.

El concepto FDT/DTM hizo que la gestión de activos fuera tan fácil como usar el administrador de dispositivos de su PC

Y al igual que en el administrador de dispositivos de Windows, cada subárbol correspondía a una rama de dispositivos, y cada rama compartía el mismo protocolo de comunicaciones.

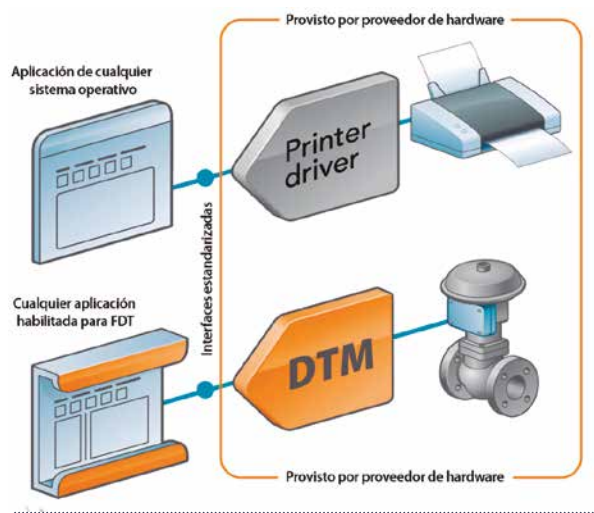


Figura 4. El concepto FDT/DTM

El enfoque modular

La tecnología FDT/DTM se basa en dos conceptos:

- » El componente FDT. También conocido como la aplicación marco, su propósito es trabajar como un entorno de tiempo de ejecución para DTM. El componente FDT puede funcionar en modo independiente o también se puede integrar en una herramienta de ingeniería/programación DCS o PLC, o en una herramienta de gestión de activos.
- » El componente DTM. El DTM es un componente de software que permite la interacción entre dispositivos de campo inteligentes, hardware de comunicaciones y controladores en una red digital. Un DTM funciona como una interfaz o contenedor para todos los parámetros, funciones, interfaz de usuario, etc. del dispositivo, creando así una representación de las características del dispositivo a través de una pantalla gráfica. Los DTM son suministrados por el fabricante con el dispositivo de campo correspondiente.

La implementación de un entorno FDT/DTM proporciona un único punto de acceso a toda la base instalada de dispositivos de campo en una planta

La implementación de un entorno FDT/DTM proporciona un único punto de acceso a toda la base instalada de dispositivos de campo en una planta, con total independencia de los protocolos de comunicación subyacentes, proporcionando una interfaz de usuario unificada que es independiente de los proveedores, el tipo de dispositivo o la función.

Internamente, los DTM funcionan como servidores, proporcionando información y funcionalidad según lo requiera el componente FDT o la aplicación marco, que funciona como cliente, solicitando la información que será empleada por las aplicaciones cliente que este contiene.

La aplicación marco puede intercambiar información con los DTM para parametrización, diagnóstico y aplicaciones específicas del dispositivo.

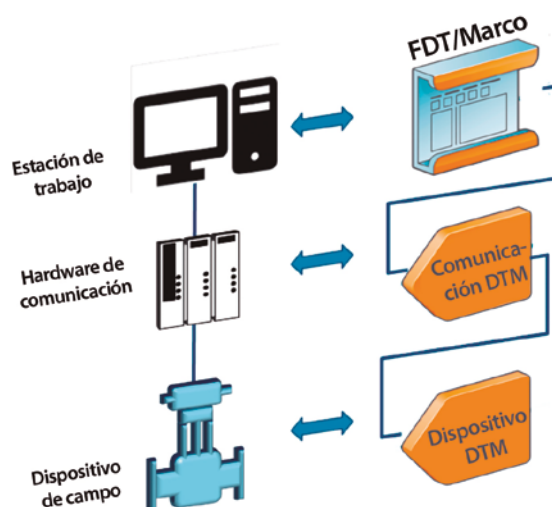


Figura 5. La estructura básica del concepto FDT/DTM

Una variedad de DTM

Los DTM contienen varios objetos, como el objeto servidor, el objeto interfaz de usuario y los objetos denominados canales. El DTM utiliza estos canales de comunicación para intercambiar información con la aplicación marco y con el dispositivo de campo real.

Estos canales crean una representación de las fuentes de datos contenidas en el dispositivo de campo.

Hay dos tipos o categorías principales de DTM:

- » DTM de dispositivos. Empleados para dispositivos de campo estándar, contienen el software de aplicación correspondiente para interactuar con dispositivos inteligentes específicos o genéricos, como un transmisor de presión.
- » DTM de comunicación. Se utilizan para la integración y configuración de hardware de comunicaciones específico, es decir, un módem HART.

Y los DTM de dispositivos pueden ser:

- » DTM de dispositivos estándar.
- » DTM de puerta de enlace (puerto). Una categoría especial de DTM que representan un dispositivo que conecta diferentes segmentos de bus de campo como, por ejemplo, un acoplador Profibus DP/PA o un puerto IO-Link para conexión con una red Profinet. Un DTM de puerta de enlace se puede ver como una combinación de un DTM de comunicación y un DTM de dispositivo. Un dispositivo conectado a la puerta de enlace física aparecería como un DTM secundario, conectado al DTM de puerta de enlace a través de un canal de comunicaciones. Esta propiedad está detrás del concepto "comunicaciones anidadas", que se discutirá más adelante.
- » DTM de dispositivo compuesto. Otra categoría especial de DTM que representa dispositivos con una estructura modular compuesta

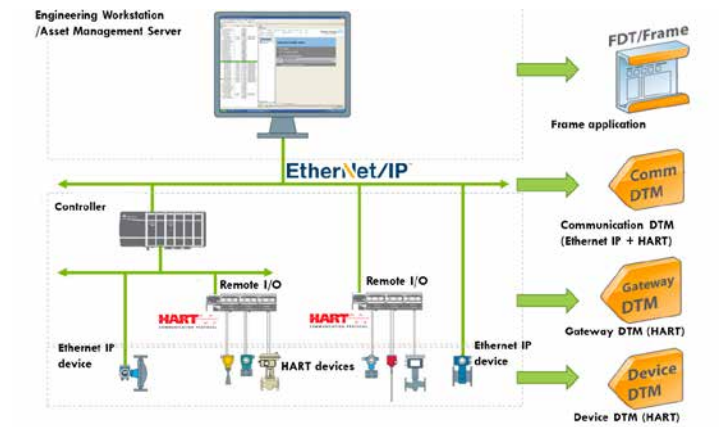


Figura 6. Ejemplo de aplicación del uso de categorías DTM

por varios módulos de hardware que se pueden conectar a ranuras físicas. Esta disposición de hardware, típica de los sistemas de E/S remotas, puede representarse mediante una jerarquía de DTM.

Las tareas de parametrización, configuración, supervisión y diagnóstico se pueden realizar desde la aplicación marco

Se requiere mantenimiento		oder		oder	
Fuera de especificación		oder		oder	
Verificar función		oder		oder	
Falla		oder		oder	
Diagnóstico activo		oder		oder	
Diagnóstico pasivo		oder		oder	

Figura 7. Colores, iconos y significados de la recomendación NAMUR NE 107

Gracias a combinar estas categorías de DTM en una aplicación marco, los usuarios pueden ensamblar una representación digital de la infraestructura de hardware real que se utiliza en la aplicación, lo que les permite obtener acceso remoto completo a los dispositivos de campo representados por su DTM correspondiente. Las tareas de parametrización, configuración, supervisión y diagnóstico se pueden realizar desde la aplicación marco.

Para las tareas de diagnóstico, se seleccionó la recomendación NE 107 de NAMUR

Para las tareas de diagnóstico, se seleccionó la recomendación NE 107 de NAMUR, lo que garantiza una apariencia coherente en todas las aplicaciones Frame relacionadas con el diagnóstico.

Habilitación de la comunicación vertical

La tecnología FDT permite la creación y el uso de comunicaciones anidadas, lo que permite el intercambio de datos a través de una topología jerárquica implementada con DTM.

La tecnología FDT permite la creación y el uso de comunicaciones anidadas, lo que permite el intercambio de datos a través de una topología jerárquica implementada con DTM

Como ejemplo, un Comm DTM PROFIBUS (interfaz Profibus/Ethernet) puede tener un puerto

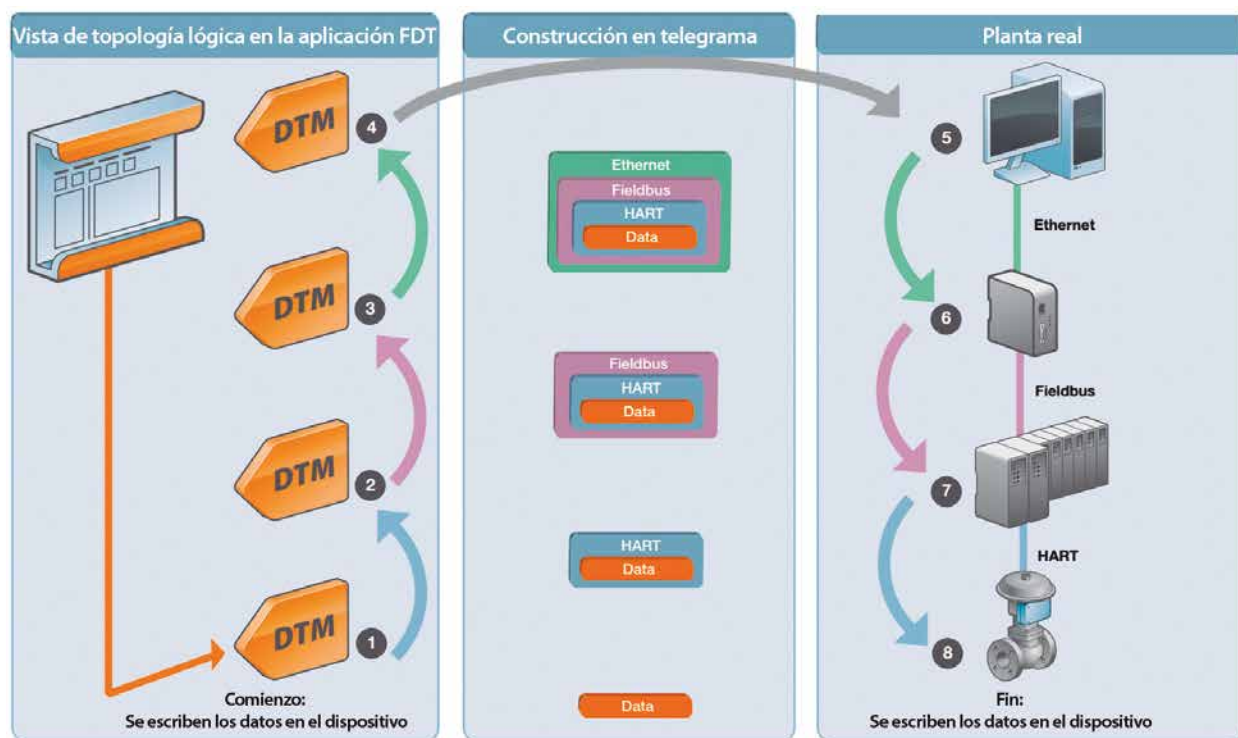


Figura 8. Tunelización a través de diferentes sistemas de comunicación utilizando tecnología FDT

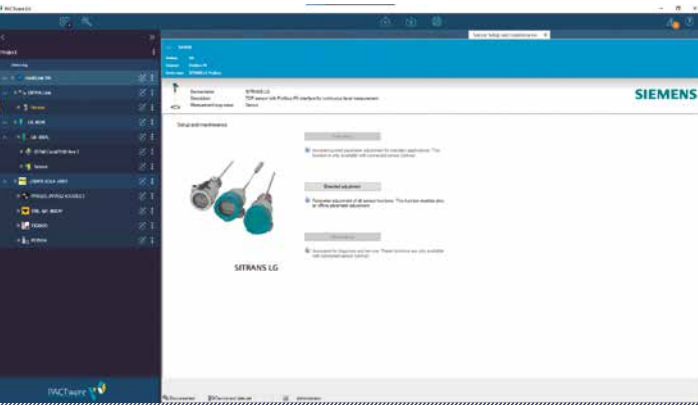


Figura 9. Pactware accediendo a un transmisor Siemens a través de un DTM DP/PA Link no nativo

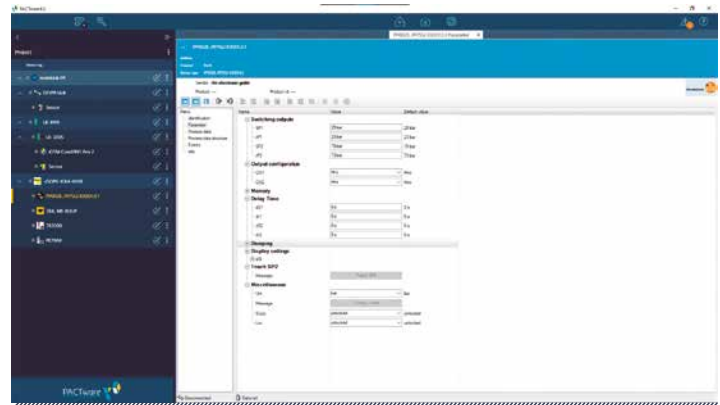


Figura 10. Pactware accediendo a un sensor de presión IFM IODD a través de un DTM intérprete para IO-Link

DTM Profibus (una cabecera de un sistema RIO Profibus) como un DTM hijo, el cual puede tener un módulo DTM (una tarjeta de E/S analógica con capacidad HART) como un DTM hijo. Este último puede tener un canal asignado a un dispositivo DTM (un transmisor compatible con HART). Esta característica permite la tunelización de datos a través de varios sistemas de comunicación.

El usuario final no es consciente de todas las conversiones de protocolo implicadas en esta topología, ya que todos los procedimientos necesarios se realizan a través de la aplicación marco.

Dado que la tecnología FDT/DTM define qué son las interfaces, pero no cómo funcionan, el proveedor del DTM es libre de implementar esas interfaces sin restricciones

La tecnología funciona como una capa de abstracción que oculta al usuario las complejidades de las traducciones de protocolos y las interfaces de capa física. Dado que la tecnología FDT/DTM define qué son las interfaces, pero no cómo

funcionan, el proveedor del DTM es libre de implementar esas interfaces sin restricciones.

Si el proveedor del dispositivo no proporciona una DTM, hay dos alternativas:

- » Algunos desarrolladores han creado DTM para dispositivos que no tienen soporte DTM nativo de su fabricante, como DTM de terceros para los equipos Siemens DP/PA Link y los sistemas de E/S remotas ET200.
- » También existen DTM intérpretes. Estos son DTM especiales que interpretan las descripciones de los dispositivos de los archivos DD o EDD en tiempo de ejecución, lo que permite la administración de dispositivos que no tienen soporte DTM nativo pero que admiten descriptores de dispositivos DD, EDD o IODD.

Aquí viene el soporte multiprotocolo

Inicialmente, el enfoque en el soporte de comunicaciones industriales se centró en los estándares Profibus y HART. A medida que pasaba el tiempo y el concepto de FDT ganaba terreno entre los proveedores, otras SDO comenzaron a mostrar interés en el concepto. Eventualmente,

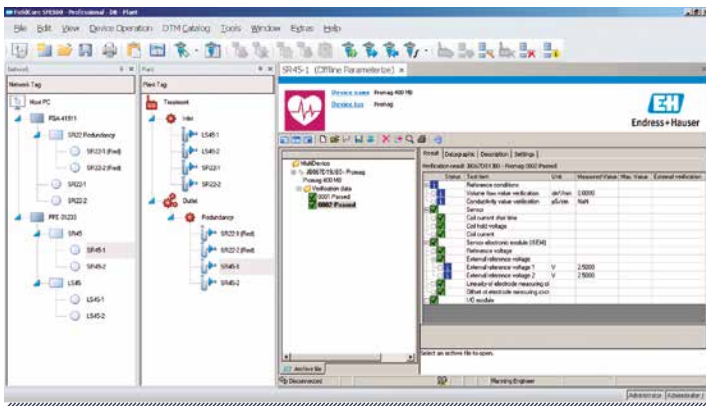


Figura 11. Software de manejo de activos basado en FDT/DTM SFE500 de Endress+Hauser



Figura 13. Vista de la estructura tipo árbol de una planta usando un marco en FDT

el soporte para más protocolos se expandió para incluir Modbus (tanto en el RS-485 como en las versiones TCP) y Ethernet IP.

Hasta la fecha, la tecnología FDT ofrece soporte para diecisiete protocolos de comunicación estándar.

Volviendo al principio de este artículo, es posible que recuerden mi sorpresa cuando pude integrar un proyecto con un conjunto diverso de dispositivos de diferentes proveedores con una sola herramienta de ingeniería.

Hoy en día damos estas cosas por sentado, pero ese proyecto fue el primero en el que pude acceder a todos los componentes y realizar su

instalación, configuración y puesta en marcha desde un solo paquete de software.

Los beneficios de un concepto modular

Uno de los primeros paquetes de software basados en el concepto FDT/DTM fue un software llamado Pactware, desarrollado por el Consorcio Pactware, que se hizo popular después de ser lanzado como descarga gratuita. Era una implementación relativamente básica de la especificación FDT, pero para el 80% de las aplicaciones era suficiente. Versiones más sofisticadas de las aplicaciones marco entraron en el mercado más tarde, con el software Field Care de Endress+Hauser, una de las implementaciones con mayor funcionalidad disponible.

Al utilizar Pactware o cualquier otra aplicación marco que cumpla con la especificación FDT, si tuviera algún dispositivo de campo, la puerta de enlace de comunicaciones requerida y sus DTM correspondientes, su configuración y puesta en marcha podría realizarse de forma local o remota. En ambos casos, la experiencia del usuario era la misma: la pantalla principal de la aplicación Frame, que presentaba un árbol jerárquico con nodos que representaban las puertas de enlace

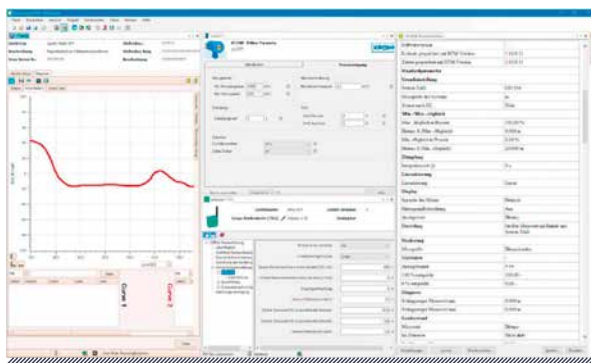


Figura 12. Pactware en acción

de la comunicación y subnodos que representaban los dispositivos de campo.

Este era el mismo tipo de estructura de árbol que los usuarios de PC habían estado empleando para trabajar en sus máquinas desde la introducción de la GUI. De repente, la planta tenía el equivalente del administrador de dispositivos de Windows. Y no era necesario gastar una fortuna en hardware y software para DCS, ya que las aplicaciones marco integradas en las herramientas de programación e ingeniería de los PLC pronto se convirtieron en algo común.

Si el software de aplicación del sistema de control tenía un marco integrado compatible, la interfaz de usuario y la forma en que la GUI de un dispositivo aparecía al usuario final eran las mismas para cualquier sistema. El tiempo y la complejidad del soporte y la capacitación podrían reducirse drásticamente, así como la cantidad de paquetes de software necesarios para realizar tareas relacionadas con la gestión de activos.

FDT/DTM se convirtió en un ecualizador entre las soluciones DCS/PCS y las soluciones PLC, al menos desde el punto de vista de la integración de dispositivos.

Además, aunque el énfasis principal de FDT se ha centrado en la industria de la automatización de

procesos, algunos proveedores de equipamiento de automatización de fábricas eligieron la tecnología FDT/DTM para equipos específicos de FA, como accionamientos, arrancadores suaves y redes de sensores discretos. Este tipo de aplicaciones muestran la flexibilidad que ofrece esta tecnología.

Algunos proveedores de equipamiento de automatización de fábricas eligieron la tecnología FDT/DTM para equipos específicos de FA, como accionamientos, arrancadores suaves y redes de sensores discretos

La ventaja de poder administrar dispositivos conectados a, por ejemplo, una red Control Net, una red Profibus y una red Modbus TCP, todas ubicadas en la misma planta, utilizando la misma aplicación única fue excelente, pero también presentó algunos problemas sutiles.

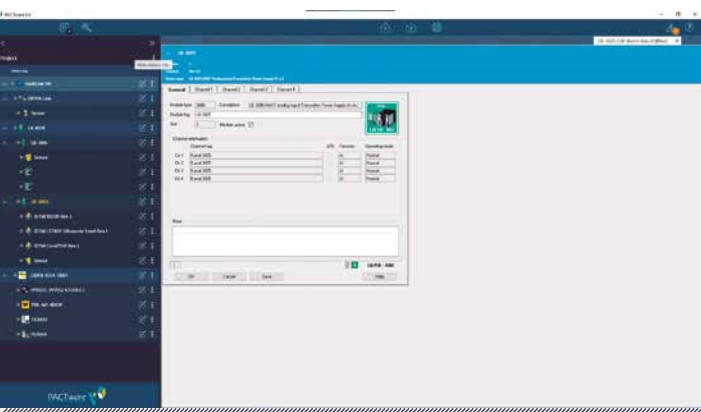


Figura 14. Pactware mostrando un proyecto que cuenta con un DP/PA Link, dispositivos Profibus PA, una E/S remota Profibus con transmisores HART y una puerta de enlace Profibus/IO-Link con sensores IO-Link.T

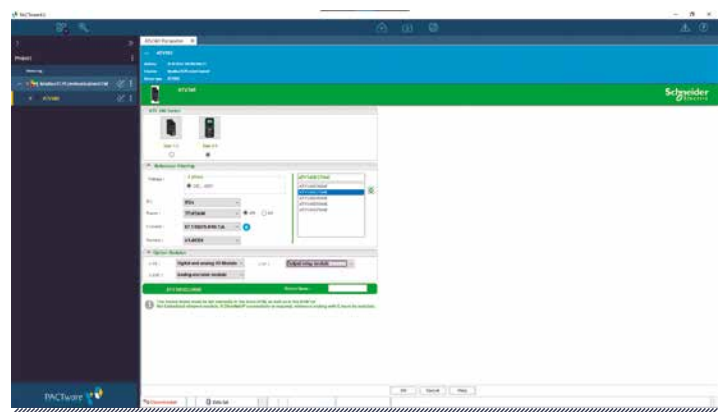


Figura 15. Configuración DTM de un arrancador suave Altivar en Pactware

La evolución de FDT/DTM: Versión 1.2

La primera generación de soluciones FDT/DTM se basó en el estándar FDT 1.2. Eran aplicaciones de escritorio de un solo usuario sin funcionalidad de red fuera de la aplicación marco.

La implementación de la arquitectura interna cliente/servidor, descrita anteriormente, para las comunicaciones entre los DTM y el FDT se logró mediante el empleo del estándar de interfaz COM (del inglés, 'modelo de objeto común') de Microsoft, XML y tecnologías Active X para la interfaz de usuario y el intercambio de datos. Por lo tanto, dependía en gran medida de las tecnologías de Microsoft. Si Microsoft actualizaba la interfaz COM, las cosas podrían salir mal.

Cualquier problema requería una versión actualizada de cualquier DTM en conflicto, lo cual no es una situación conveniente. Además, el nivel de seguridad que se implementó consistió en la configuración de seguridad según el rol del usuario. Por supuesto, esta fue en una época anterior a la actual, en la que los problemas y políticas de ciberseguridad aún no tenían la importancia que tienen hoy en día.

Pero se estableció el concepto básico, y la estructura de árbol jerárquico empleada como metáfora de la pirámide jerárquica de automatización física era flexible, amigable y replicaba el hardware real de una manera que era bastante intuitiva.

Una de las características más útiles de la especificación FDT era la posibilidad de realizar la instalación y configuración fuera de línea de los dispositivos de campo. Esta característica es especialmente útil para los constructores de máquinas o skids, ya que pueden configurar el primer conjunto de dispositivos, guardar la configuración en la computadora, y luego simplemente descargarla en cualquier número de conjuntos de dispositivos idénticos que puedan necesitar.

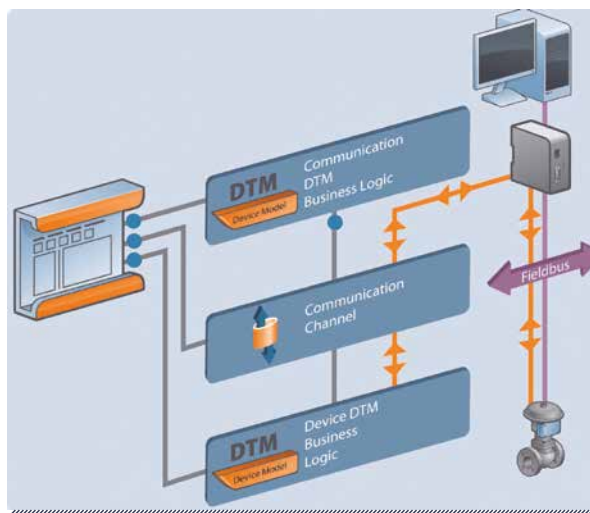


Figura 16. Dispositivos reales representados por una implementación FDT/DTM

La tecnología FDT/DTM suele compararse con EDDL, pero desde mi punto de vista son dos herramientas diferentes. EDDL funciona como un archivo Profibus GSD, es decir, un archivo de texto que describe las características de un dispositivo para informar a la herramienta de configuración qué parámetros se pueden ajustar. FDT/DTM funciona como la combinación de un navegador web y páginas web dinámicas, la aplicación marco muestra el contenido de los DTM y permite al usuario interactuar con el dispositivo de campo real o con una representación virtual del mismo.

La tecnología FDT/DTM suele compararse con EDDL, pero desde mi punto de vista son dos herramientas diferentes

La evolución de FDT/DTM: Versión 2.0

La siguiente versión del estándar FDT/DTM, 2.0, corrigió gran parte de las deficiencias de la primera versión. Empezando con el aspecto cada vez más importante de la seguridad, la versión 2.0 incorporó mejoras de seguridad integrales, como DTM firmados y certificados digitalmente y la adopción de la tecnología Microsoft.NET para habilitar la seguridad de acceso a código, pero manteniendo la compatibilidad con versiones anteriores de los DTM versión 1.2 al mismo tiempo. El intercambio de datos se implementó utilizando tipos de datos pasados mediante interfaces en lugar de a través de XML.

La siguiente versión de FDT/DTM, 2.0, corrigió gran parte de las deficiencias de la primera

La interfaz de usuario empleaba Windows Presentation Foundation y WinForms. Fue una actualización que abrazó la tendencia hacia componentes de software libres y de código abierto que eventualmente hicieron de FDT/DTM un concepto a prueba de futuro. Y al dejar atrás la tecnología basada en COM y DCOM, FDT/DTM se convirtió en una plataforma de software más estable y robusta.

Al dejar atrás la tecnología basada en COM y DCOM, FDT/DTM se convirtió en una plataforma de software más estable y robusta

Esto es posible porque la interacción entre estos componentes se realiza exclusivamente a través de la aplicación marco, mejorando así la

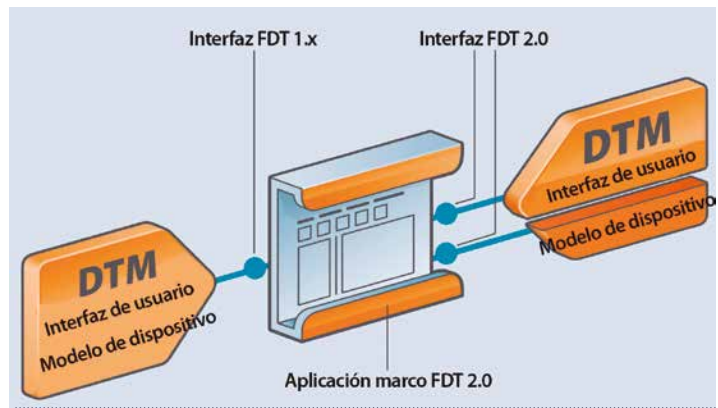


Figura 17. En la versión 1.2 la interfaz de usuario podía interactuar directamente con la lógica de negocio del DTM. FDT 2.0 cuenta con una arquitectura cliente/servidor que permite ejecutar la interfaz de usuario y la lógica empresarial en diferentes PC.

interoperabilidad gracias a estandarizar cómo los diferentes componentes trabajan juntos e incorporan la funcionalidad de red.

Con esta característica se crean dos grandes ventajas:

- » Es posible la implementación de repositorios de DTM centralizados en servidores.
- » Los DTM ahora se pueden implementar utilizando soluciones móviles.

Con la primera característica, la tecnología FDT comienza a abordar una de las críticas más comunes que recibió en comparación con las descripciones de dispositivos basadas en texto, la posibilidad de actualizaciones en línea en un repositorio de DTM centralizado. Gracias a permitir esta opción, cada usuario de una configuración multiusuario siempre tendrá acceso a la última versión de los DTM que pueda necesitar.

La segunda característica también depende del uso de una arquitectura distribuida. Dado que las interfaces de usuario, las lógicas de negocio y las aplicaciones marco pueden ejecutarse en diferentes dispositivos, es posible utilizar dispositivos móviles para ejecutar aplicaciones marco. Por lo tanto, cualquier dispositivo móvil puede funcionar como una estación de trabajo de gestión de activos o como un configurador portátil.

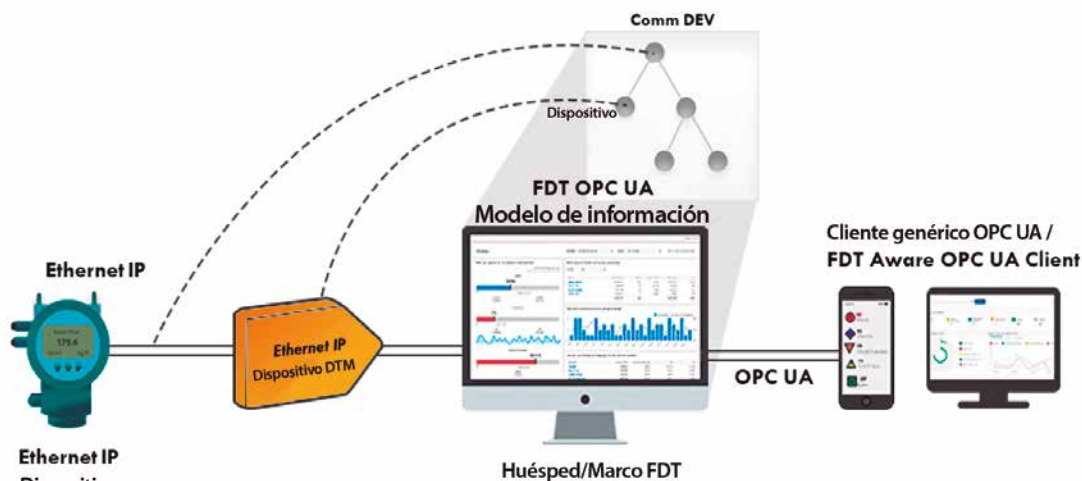


Figura 18. Asignación de un dispositivo IP Ethernet DTM en un modelo de objetos OPC UA

Agregue un poco de OPC a la mezcla, y revuelva

FDT 2.0 tiene una característica adicional: la incorporación de un servidor OPC UA. Este servidor puede funcionar como una aplicación marco o como un servidor OPC UA. Al hacer esto, el servidor OPC UA puede crear un modelo de información OPC UA que representa cómo interactúan los DTM entre sí en la aplicación frame, lo cual es de hecho una representación de cómo funcionan los dispositivos en la vida real.

La versión FDT 2.1 incorporó mejoras en la funcionalidad y eliminó errores

La versión FDT 2.1 incorporó mejoras en la funcionalidad y eliminó errores. También permitió el soporte para múltiples conexiones de comunicación en línea, ya que anteriormente el estándar FDT/DTM tenía soporte para una sola conexión de comunicación en línea.

El soporte de protocolos nuevos se habilita mediante la instalación de cualquier DTM que los

usara. Anteriormente esto solo era posible instalando los DTM de comunicación específicos del protocolo y los DTM de puerta de enlace. Una gran cantidad de proveedores de DCS y PLC comenzaron a emplear la tecnología FDT/DTM integrada en su software de ingeniería para mayor simplicidad.

Desde descripciones de dispositivos hasta modelos de información

El cambio más significativo experimentado por la tecnología FDT/DTM en su evolución de la versión 1.2 a la 2.1 fue en la forma en que los datos son organizados e intercambiados por los objetos definidos estándar.

En lugar de emplear documentos basados en XML para el intercambio de datos, la versión 2.0 utiliza objetos .NET. Este es un método más rápido porque los documentos XML deben analizarse y reestructurarse cada vez que se intercambian. Mediante el uso de objetos .NET, los DTM se convierten en objetos que funcionan como representaciones digitales de los dispositivos de campo reales.

Mediante el uso de objetos .NET, los DTM se convierten en objetos que funcionan como representaciones digitales de los dispositivos de campo reales

Este cambio crea un requisito: los DTM intercambian información como tipos de datos. Un módulo de tipo de datos define las operaciones que se pueden realizar en los datos, el significado de los datos y la forma en que se pueden almacenar los valores de ese tipo. El uso de tipos de datos requiere la creación de un modelo de información que funcione como base para la DTM. En los DTM de la versión 2.0, la interfaz de usuario está separada del modelo lógico del dispositivo, mientras que en los DTM de la versión 1.2 se combinaron ambos.

De este modo, los DTM se vuelven más flexibles y, en consecuencia, las interfaces de usuario se

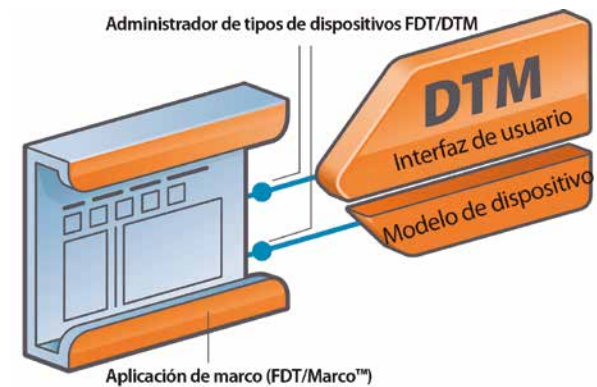


Figura 19. DTM que muestra la separación entre el modelo lógico del dispositivo y la interfaz de usuario

pueden ejecutar en una estación de trabajo mientras que la lógica del dispositivo se ejecuta en un servidor. La estrecha relación entre un dispositivo y su DTM correspondiente permite la detección automática del DTM correcto para un tipo o versión de dispositivo específico. En la versión 1.2, la asignación de DTM se realizaba manualmente, una práctica que causaba problemas

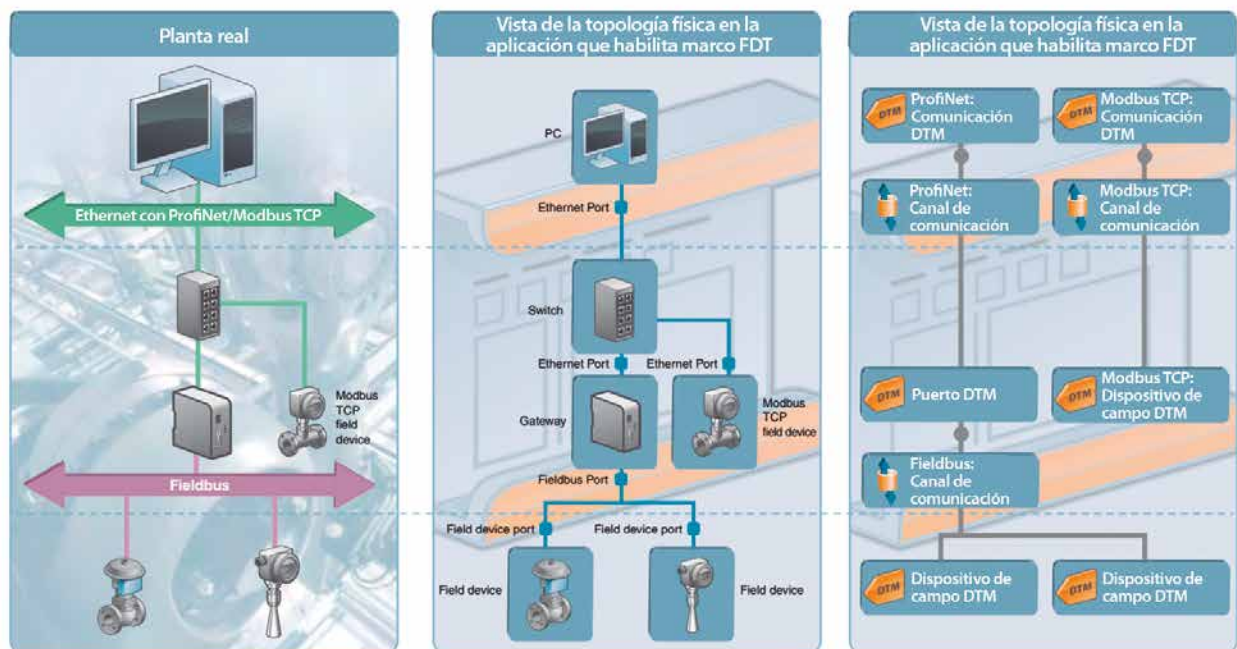


Figura 20. Comparación entre topología de plantas de la vida real, topología física y topología lógica

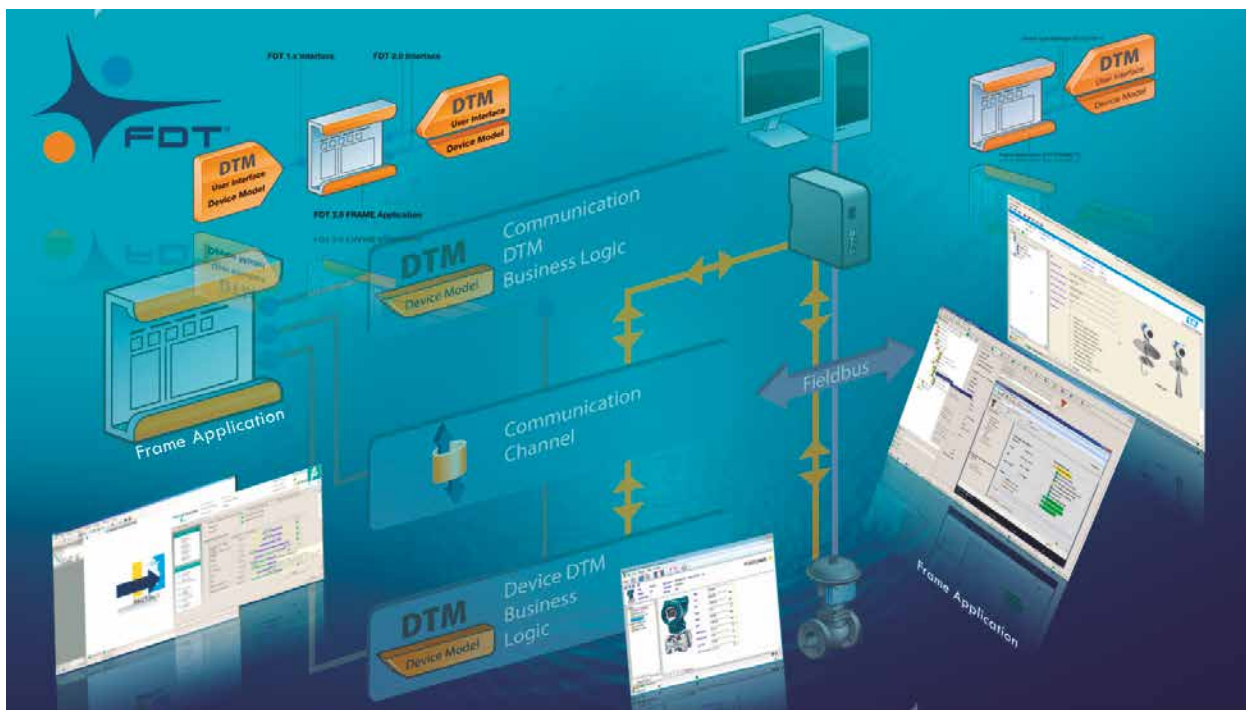
debido a la posibilidad de asignar DTM incompatibles con los dispositivos.

La implementación de un entorno FDT/DTM completo en una planta crea una topología lógica que imita la topología física

La implementación de un entorno FDT/DTM completo en una planta crea una topología lógica que imita la topología física. La topología física es una representación de la topología de red de la vida real formada por los dispositivos, puertas de enlace y convertidores de medios reales. La topología lógica es una representación de las relaciones de intercambio de datos entre DTM.

Para garantizar la interoperabilidad entre todos los componentes de software, se creó un conjunto de componentes comunes para controlar aspectos como el comportamiento de la interfaz. Este enfoque garantiza la interoperabilidad y simplifica la creación de DTM, reduciendo así los costos de desarrollo.

El uso opcional de un servidor OPC UA dentro de la aplicación marco requiere la compatibilidad entre el modelo de información utilizado por los DTM y los modelos de información utilizados en el servidor OPC. Esto crea una representación de los DTM usando el modelo de información OPC UA. Este modelo expone los datos de los nodos representados por los DTM utilizando los servicios OPC UA, poniendo los datos a disposición de cualquier cliente OPC UA.



Fuente: Mirko Torrez Contreras

Un gran salto hacia la digitalización

La innovación y las mejoras incluidas en FDT Versión 2.0 tuvieron una fuerte influencia conceptual sobre la importancia de los protocolos de comunicación. Dado que el uso de técnicas de modelado de información eliminó los problemas de traducción de protocolos para el usuario final, los protocolos se vuelven secundarios en importancia a la disponibilidad de un modelo de información de dispositivos coherente estructurado jerárquicamente.

La innovación y las mejoras incluidas en FDT Versión 2.0 tuvieron una fuerte influencia conceptual sobre la importancia de los protocolos de comunicación

El uso de tipos de datos fue el paso previo a la adopción de modelos de información basados en la semántica, por lo que FDT versión 2.0 se convirtió en la base fundamental a partir de la cual evolucionaron los conceptos posteriores.

Las prácticas de administración de dispositivos experimentaron un cambio dramático. Desde la pequeña herramienta de configuración de dispositivos que FDT/DTM era en sus inicios, el concepto ha crecido en escala para convertirse en una arquitectura de middleware disponible en toda la planta, basada en el concepto servidor/cliente, que hace posible el acceso a los datos en toda la pirámide de automatización completa.

FDT se ha convertido en un estándar ampliamente implementado, con millones de DTM de dispositivos en uso compatibles con una variedad de entornos de huésped FDT. Ha sido adoptado como estándar global por IEC (IEC 62453), ISA (ISA/ASNI 103) y GB-T 29618-2017.

La gama de aplicaciones que utilizan la tecnología FDT va desde aplicaciones de escritorio independientes, aplicaciones móviles personalizadas, integradas en software de configuración o ingeniería de PLC o DCS e integradas en entornos OPC UA.

La gama de aplicaciones que utilizan la tecnología FDT va desde aplicaciones de escritorio independientes, aplicaciones móviles personalizadas, integradas en software de configuración o ingeniería de PLC o DCS e integradas en entornos OPC UA

El tiempo avanza de manera implacable y también lo hace la tecnología. Nuevas alternativas para la administración e integración de dispositivos comenzaron a aparecer a finales de la década de 2010. Por lo tanto, la tecnología FDT requiere avanzar y mantener su liderazgo tecnológico.

Otro salto se estaba gestando con la versión 3.0, y esta vez el objetivo estaba en las nubes. ❖