

Proyectos Tractores

MORFEO-Wasup:

Sistemas de Supervisión, mantenimiento y control de planta

Acrónimo: Morfeo-Wasup

Entregable D4.2

Plataforma ejecución agentes de control

Versión: 2

Fecha de preparación: 28/10/2008

Editor: CTIC



Tabla de contenidos

1	INTRODUCCIÓN.....	4
1.1	OBJETIVO.....	4
1.2	AUDIENCIA.....	4
1.3	DOCUMENTACIÓN RELACIONADA.....	4
1.4	ESTRUCTURA DEL DOCUMENTO.....	4
2	PASARELA DE COMUNICACIÓN CON ELEMENTOS DE PLANTA.....	5
2.1	INTRODUCCIÓN.....	5
2.2	AGENTES	5
2.3	FUNCIONAMIENTO GENERAL.....	5
3	REFERENCIAS.....	7
3.1	REFERENCIAS.....	7
	GLOSARIO DE TÉRMINOS Y ACRÓNIMOS.....	8

Tabla de Ilustraciones

1 Introducción

1.1 *Objetivo*

Este documento pretende describir la implementación y el funcionamiento de la plataforma de ejecución de los agentes participantes en la pasarela de comunicación con los elementos de planta. La definición del OPCClienteWasup y la comunicación http con las capas superiores de la arquitectura conformará la pasarela entre la planta y las capas de supervisión.

1.2 *Audiencia*

Este documento va dirigido al Ministerio de Industria y a todos los miembros del consorcio MORFEO-Wasup.

1.3 *Documentación relacionada*

- Todos los documentos del proyecto PROFIT Tractor MORFEO-Sistemas de supervisión, mantenimiento y control de planta (WASUP).

1.4 *Estructura del Documento*

En el capítulo 1 se resumen el objetivo, audiencia y documentación relacionada con el presente documento. En el capítulo 2 se describe la construcción de la pasarela de comunicación, implementación de cada agente participante y funcionamiento general.

2 Pasarela de comunicación con elementos de planta

2.1 Introducción

La pasarela de comunicación con elementos de planta hará posible que los datos que interesan estén dispuestos para su monitorización y supervisión. Es una arquitectura basada en el estándar OPC y en la tecnología cliente/servidor.

2.2 Agentes

La pasarela de comunicación está formada por los siguientes elementos software:

OPCClient:

Es el elemento encargado de acceder a los elementos captados por las sondas de la planta y ponerlos a disposición de las capas superiores de la arquitectura para su posterior tratamiento (supervisión). Está implementado en el lenguaje C++. Se ha utilizado como plataforma de implementación el Visual Studio 2005 Professional Edition, pero sería fácilmente adaptable a cualquier compilador de este lenguaje, puesto que no se usan librerías específicas de esta tecnología.

El OPCClientWasup se basa en la especificación OPC-DA (Acceso a datos en tiempo real) del estándar OPC. Accede al driver proporcionado por el fabricante del dispositivo al que queremos acceder (Servidor OPC Comercial) y por tanto a todos los datos manejados por ese dispositivo.

Mediante peticiones http, el OPCClientWasup, envía los datos que provienen de planta a las capas superiores (Receptor de Datos), encargadas de la supervisión de datos y de la gestión de alarmas relacionadas con dichos valores.

2.3 Funcionamiento General.

El OPCClient utilizando el estándar OPC dispone de determinadas funciones que le permiten acceder a los datos controlados por el PLC, concretamente, le permite acceder a ese Servidor OPC Comercial proporcionado por los fabricantes de los dispositivos.

1. El cliente OPC implementado en Wasup se conectará al servidor o servidores OPC Comerciales a través de hilos.
2. Siguiendo la especificación OPC-DA se accede a los datos manejados a través de grupos, que actúan de contenedores de las variables que nos interesa traer de planta.
3. Las variables (ítems) se envían junto con su valor y atributo de calidad a través de una petición POST al contenedor de recursos (recurso measured_values). Con esta acción se consigue crear una nueva medida en la base de datos general del sistema si ese ítem existía previamente, o bien, crear un nuevo ítem para que posteriormente puedan almacenarse sus valores.

4. Se realiza una petición PUT al contenedor de recursos (recurso measurement_type) con el fin de rellenar de forma automática en la base de datos, información correspondiente a cada ítem, (descripción y unidades de ingeniería).
5. El OPCClienteWasup posee un tiempo de refresco, es decir, el tiempo cada el cual el cliente lee variables de planta y examina su valor para vigilar posibles cambios. Si cada vez que transcurre el tiempo de refresco se produce una variación de valor en las variables, el cliente vuelve a hacer una petición al contenedor de recursos (recurso measured_values) para enviar los nuevos valores.

El cliente OPC incluye seguridad de comunicaciones HTTP Basic, de tal manera que para el acceso a los recursos se solicitará un usuario y contraseña, válido en el sistema.

3 Referencias

3.1 Referencias

- Wikipedia, la enciclopedia libre. <http://es.wikipedia.org/>
- Manual básico de programación en C++. <http://www.bairesrobotics.com.ar/poo/manual%20programacion%20c++.pdf>
- Especificaciones OPC de la OPCfoundation. <http://www.opcfoundation.org>
- MySQL Reference Manual. <http://downloads.mysql.com/docs/refman-5.0-es.a4.pdf>

Glosario de términos y acrónimos

C++

Lenguaje de programación que abarca tres paradigmas: programación estructurada, programación genérica y programación orientada a objetos.

HTTP

Protocolo de transferencia de hipertexto. Es el protocolo usado en cada transacción de la Web y que define la sintáxis y la semántica que utilizan los elementos software de la arquitectura web para comunicarse.

IP

Dirección de un dispositivo en una red informática.

MySQL

Sistema de gestión de base de datos relacional, multihilo y multiusuario desarrollado como software libre.

OPC

OLE for Process Control. Estándar de comunicación en el campo de control y supervisión de procesos.

PLC

"Programmable Logic Controller". Dispositivos electrónicos muy usados en Automatización industrial. Controlan no sólo la lógica de funcionamiento de máquinas, plantas y procesos industriales, sino que también pueden realizar operaciones aritméticas, manejar señales analógicas...

REST

"Transferencia de Estado Representacional". Describe cualquier interfaz web simple que utiliza XML y HTTP sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes.

XML

"Lenguaje de marcas extensible". Metalenguaje extensible de etiquetas que permite definir lenguajes para diferentes necesidades.