

IEC 61499 (II)

Miguel Ángel García Blanco

Entornos de desarrollo y ejecución

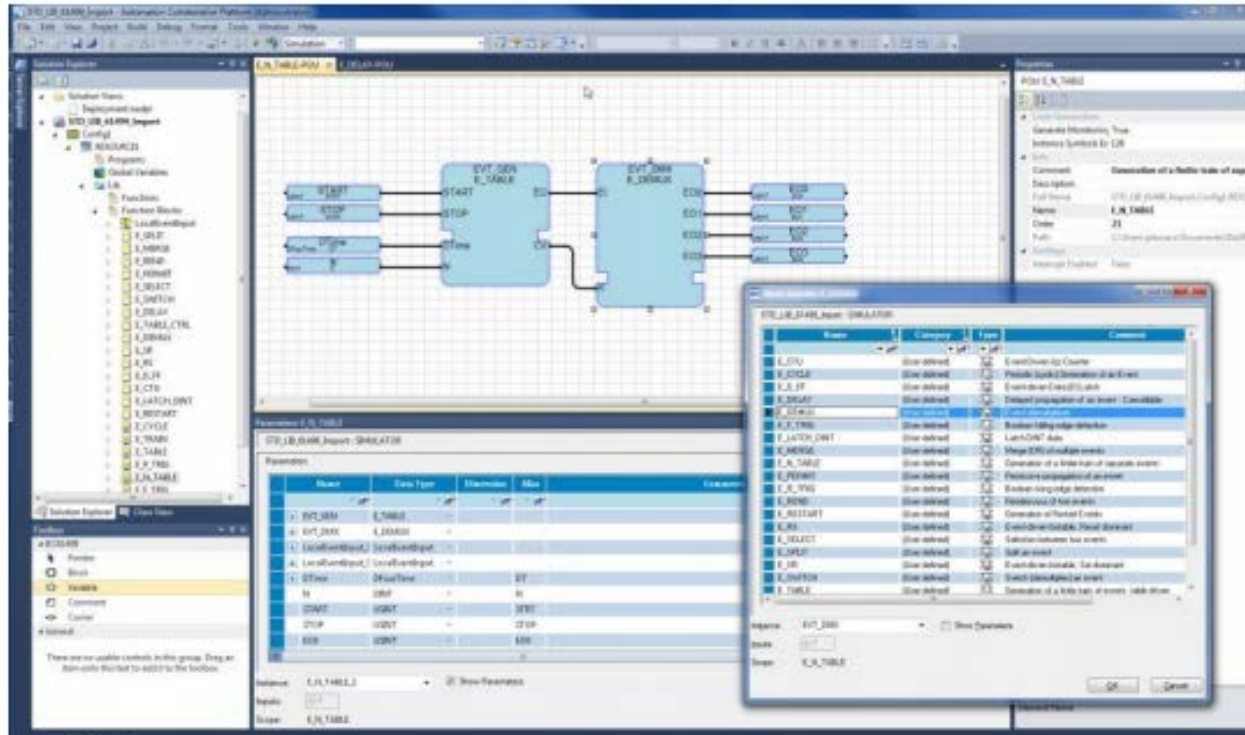
- IEC 61499 se estandarizó en 2005
- Entre el 2000 y el 2005 estuvo disponible en forma de especificación de dominio público
- Versión nueva en 2010
- La mayor parte del trabajo existente es académico y ha dado lugar solo a prototipos
- Los entornos de desarrollo constan de:
 - IDE (Integrated Development Environment)
 - Runtime: Ejecución de las FBN

Entornos de desarrollo

- FBDK (Runtime: FBRT): Desarrollo académico y de investigación
 - Editor basado en Java
- 4DIAC (Runtime: FORTE): Herramienta de código abierto

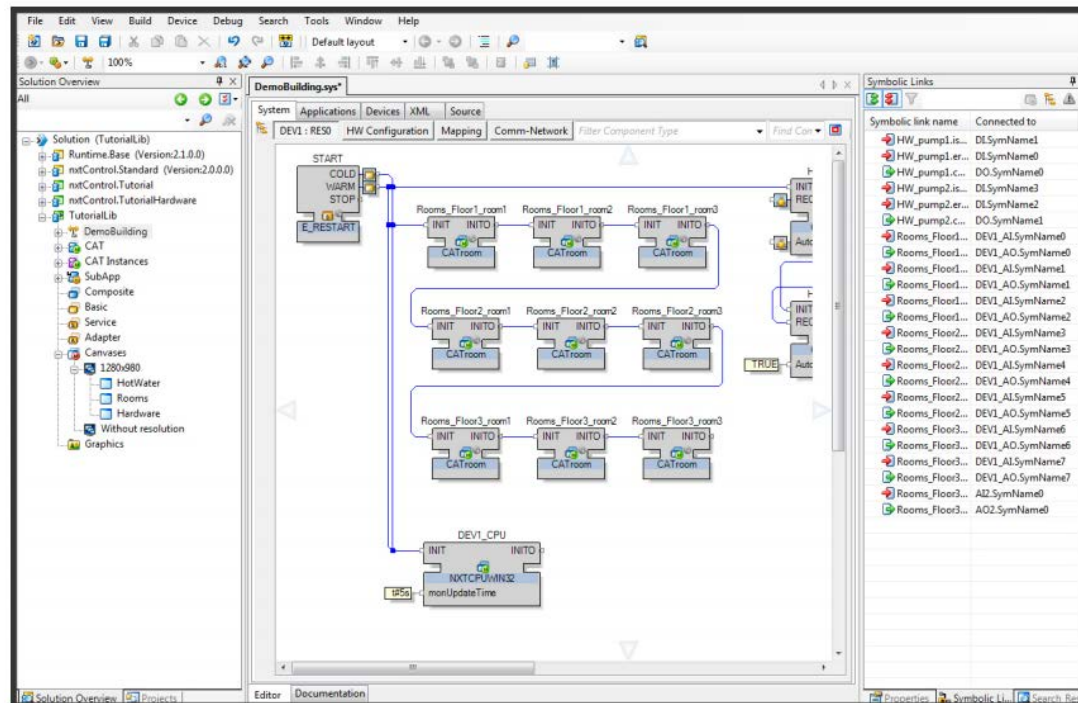
Entornos de desarrollo

- ISaGRAPH: Entorno propietario de Allen-Bradley
 - Cientos de plataformas hardware, más de un millón de dispositivos de base instalada



Entornos de desarrollo

- NxtOne: Herramienta comercial
 - Soporta hardware de Bechhoff, Wago, Siemens, Advantech, Mitsubishi, Bosch
 - IDE de nxtStudio:



Entornos de desarrollo

- Otras herramientas de código abierto
 - FBench: Clases Java
 - ICARU-FB: Universidad de Santa Catarina, Brasil
 - GASR-FBE: Entornos Windows
- Otros desarrollos académicos y de investigación
 - Blok-IDE
 - Corfu-ESS / Archimedes: Entorno de ejecución de tiempo real
 - COSME: Universidad de Zaragoza

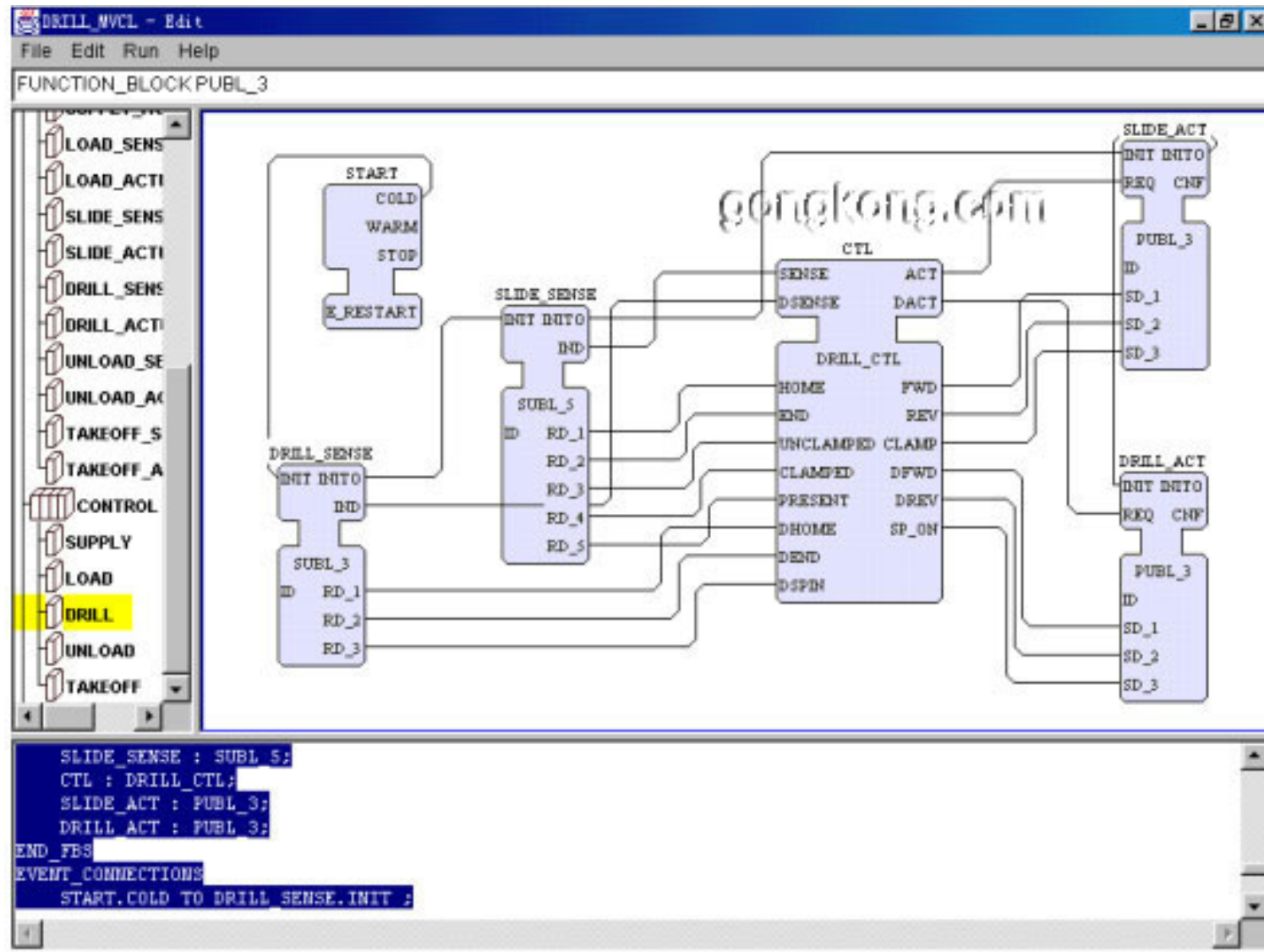
Function Block Run-Time Environment (FBRT)

- Es la primera implementación de un entorno de ejecución para IEC 61499
- Desarrollado por James H. Cristensen
- Disponible públicamente en www.holobloc.com
- Es una implementación de referencia, que se usó para testear y verificar los modelos del estándar
- Implementación basada en eventos que cumple prácticamente todos los requisitos del estándar.

FBRT

- Implementado en Java
- Elementos de IEC 61499 implementados en clases de Java
- Elementos activos, tales como temporizadores, son threads de Java.
 - Los elementos mapeados relacionados, como espera de timeouts, se manejan dentro de los threads para actualizar datos y dispara eventos de salida.
 - El mecanismo de notificación de eventos se implementa como llamada a funciones

Interfaz de FBRT



Interfaz de FBRT

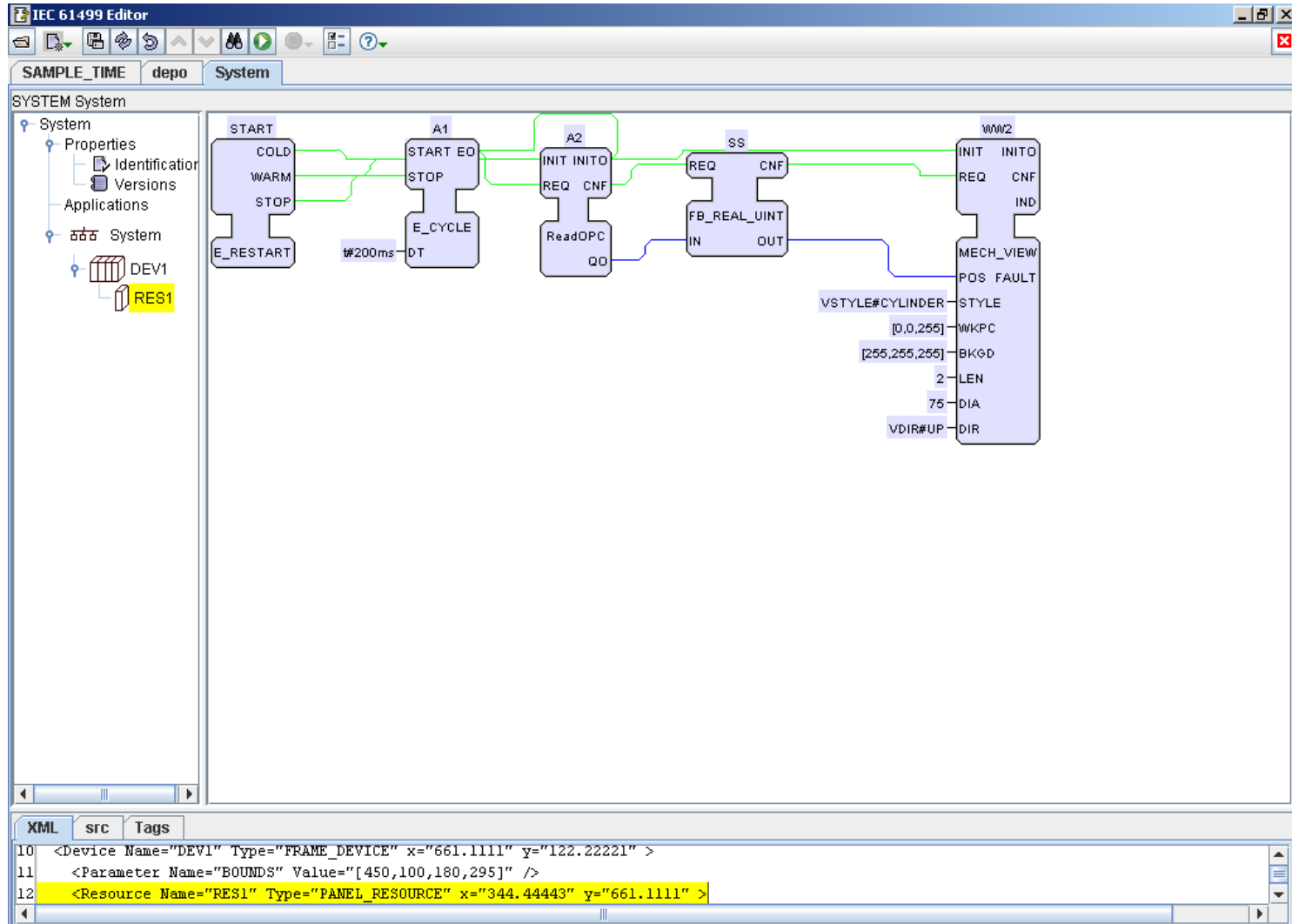
The screenshot displays the IEC 61499 Editor window, titled "IEC 61499 Editor". The main workspace shows the "FUNCTION_BLOCK LOAD_CTL" configuration. On the left, a tree view shows the project structure, including "LOAD_CTL", "Properties", "Identification", "Versions", "CompilerInfo", "Compilers", "ECC", and "Algorithms". The central diagram illustrates the block's interface with the following connections:

- Inputs: "EVENT" (top left), "REQ" (top center), "IND" (top right), "BOOL" (middle left), "BOOL" (bottom left), "BOOL" (bottom center), and "BOOL" (bottom right).
- Outputs: "EVENT" (top right), "SRC_RDY FWD" (middle right), "SRC_PRESENT" (bottom center), "DST_RDY" (bottom center), and "DST_PRESENT" (bottom right).

At the bottom, the XML editor shows the following code:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml" >
3 <FBType Name="LOAD_CTL" Comment="Workpiece LOAD Sol" >
4   <Identification Standard="61499-1" Classification=" " >
5   <VersionInfo Organization="Holobloc Inc" Version="1.0" >
6   <VersionInfo Organization="Holobloc Inc" Version="1.0" >
7   <VersionInfo Organization="Rockwell Automation" Version="1.0" >
```

Interfaz de FBRT



Interfaz de FBRT

SYSTEM TANK_MVCL

TANK_MVCL

- Properties
 - Identificator
 - Versions
- Applications
 - TANK_MVCL
 - CTRL
 - RECORD
 - TANK_VIE

CTRL

RECORDER

TANK_VIEW

FRAME_DEVICE

FRAME_DEVICE

FRAME_DEVICE

BOUNDS

BOUNDS

BOUNDS

GRID

GRID

GRID

[100,100,140,275]

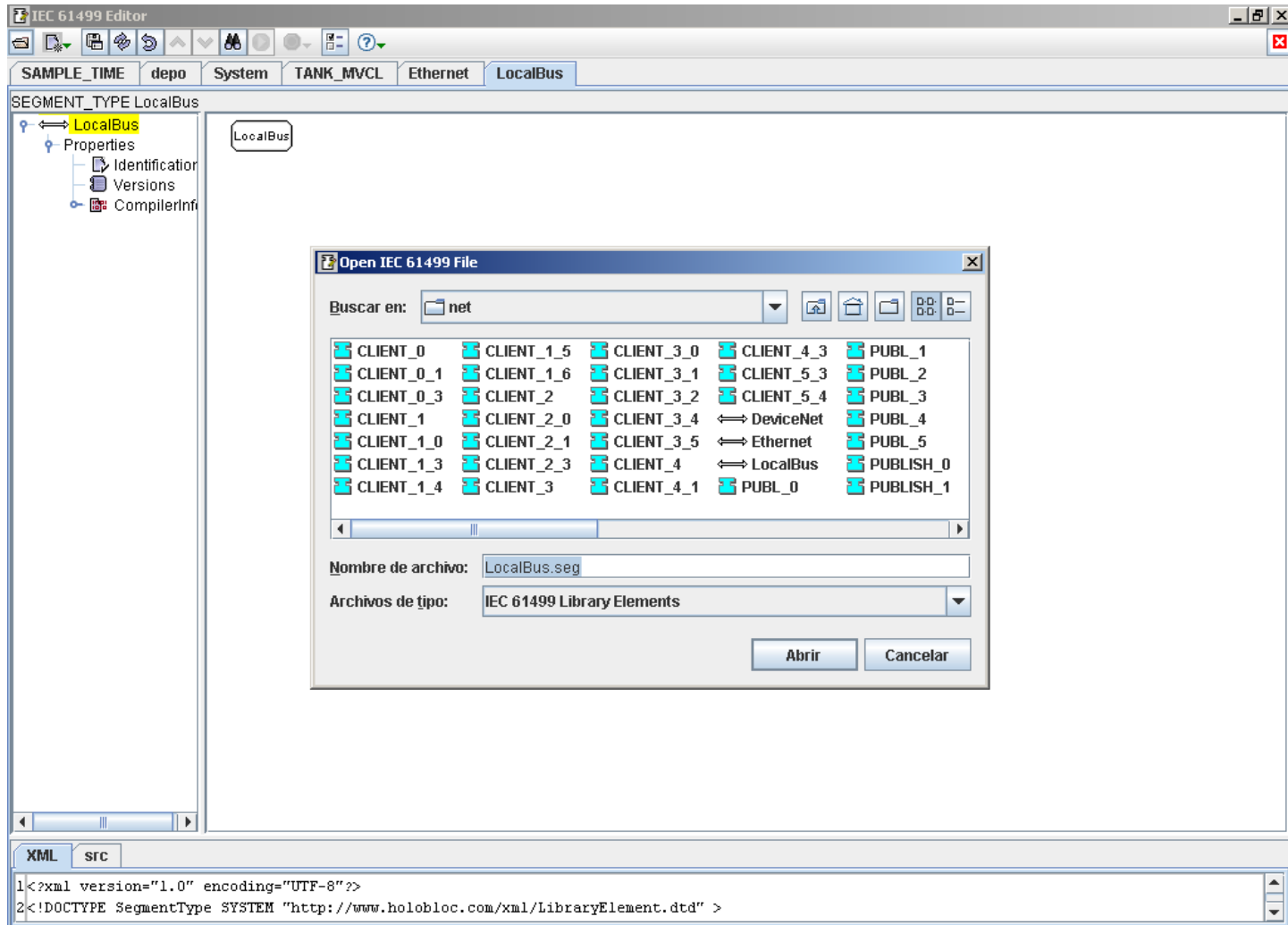
[240,100,210,210]

[460,100,180,295]

XML src Tags

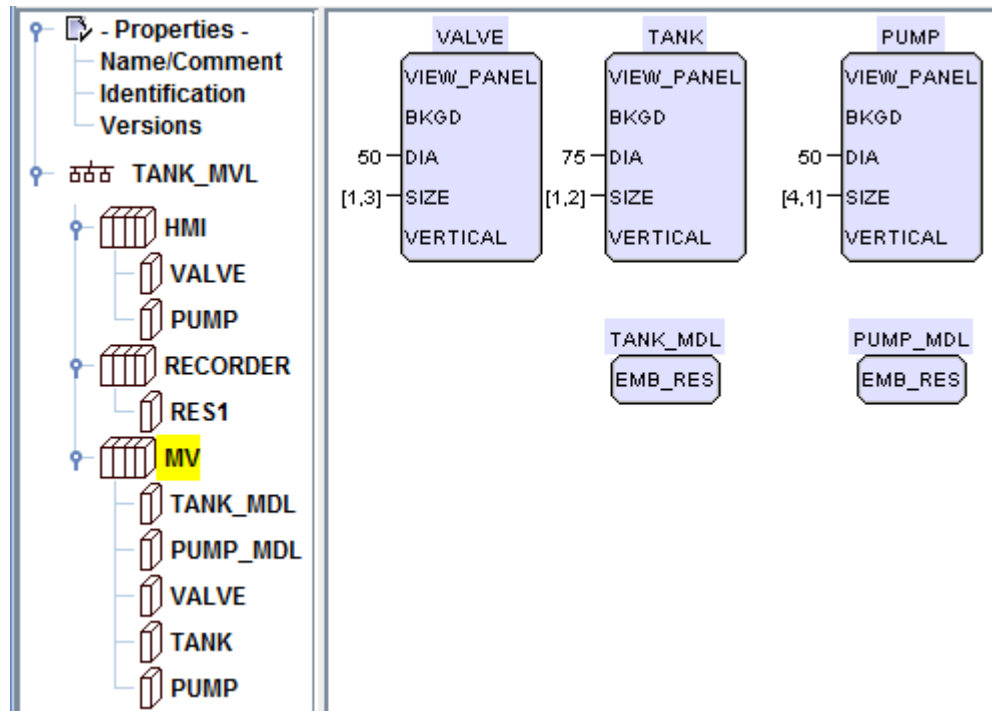
```
7 <Device Name="CTRL" Type="FRAME_DEVICE" Comment="Controller with Faceplate" x="561.1111" y="11.111111" >
8   <Parameter Name="BOUNDS" Value="[100,100,140,275]" />
9   <Resource Name="CTL" Type="EMB_RES" x="372.2222" y="488.88885" >
```

Interfaz de FBRT



PROCESS/TANK_MVL

- TANK_MDL y PUMP_MDL son EMB_RES y reciben todos sus bloques al empezar la ejecución del sistema
 - Ver src de los modelos



Comentario de aplicaciones

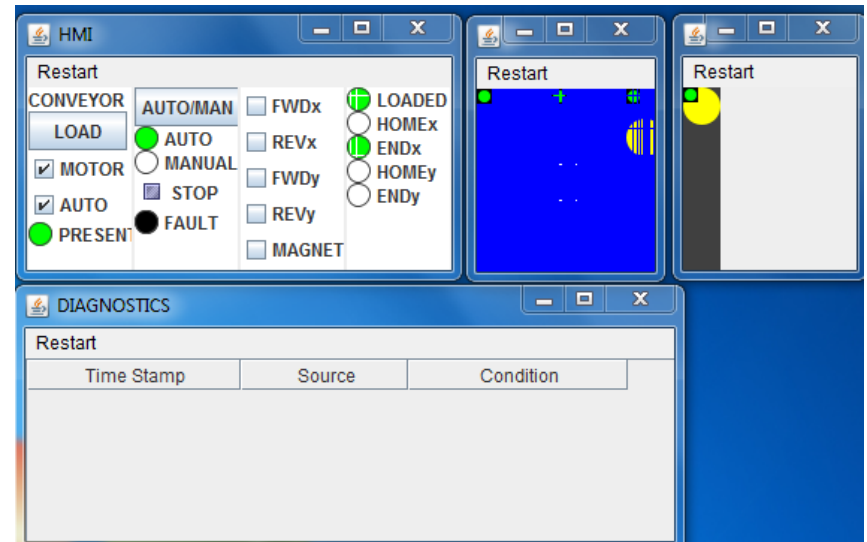
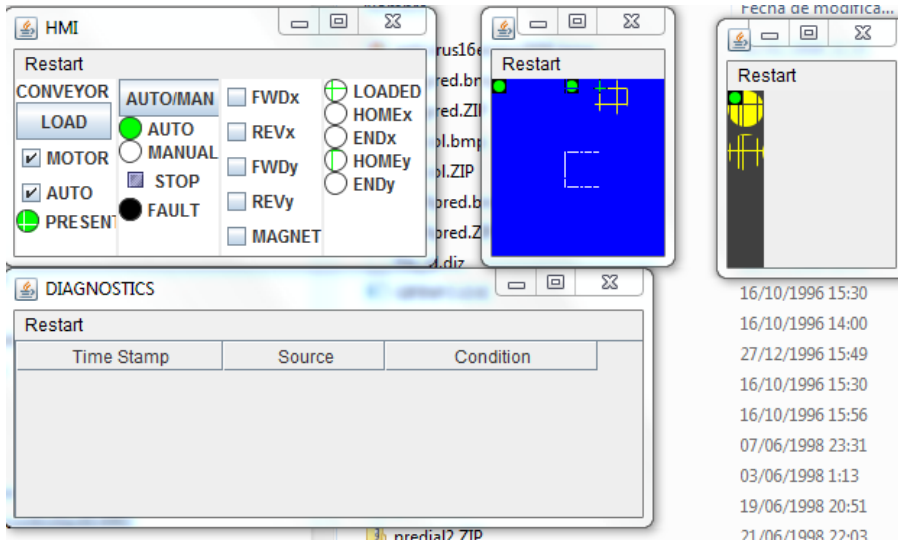
- DOC/FLASHER_TESTD
- HVAC/VAV_VIEWWL: PUBL/SUBL completables
- ITA/RMT_FRAME_TEST: Descarga de bloques en device remoto con frame al iniciar la ejecución
- ITA/MGR_TEST2: Descarga de bloques desde dispositivo en tiempo de ejecución en device remoto sin frame

Comentario de aplicaciones (II)

- TEST/TEST_TAGS: Quitar MGR del Device PUBLISHER
- TEST/TEST_SWING
- TEST/VSTYLE
- MACH/ASM_MVCDL: Ver ECC de CONTROL.ASSY_CTL.ASM_CTL
- MACH/DRILL_MVCDR: Con dispositivo remoto, ladder en LOAD.LOAD_CTL y en TAKEOFF.XBAR_CTL
- MACH/LIFTER_MVCDR: FEED_MDL.CNVG_MDLL es un composite, CNVG_MDL con algoritmo en java por eventos
- MACH/PNP_MVCDR: CONTROL.AUTO_CTL.PNP_CTL_LD en ladder

MACH/PNP_MVCDM

- Dos RMT_FRM: CNV y PNP, cada uno con su puerto
 - Descargas en los dos al inicio de la ejecución
- Dos FRAME_DEVICE de supervisión conjunta: HMI y DIAGNOSTICS



Ficheros en FBDK

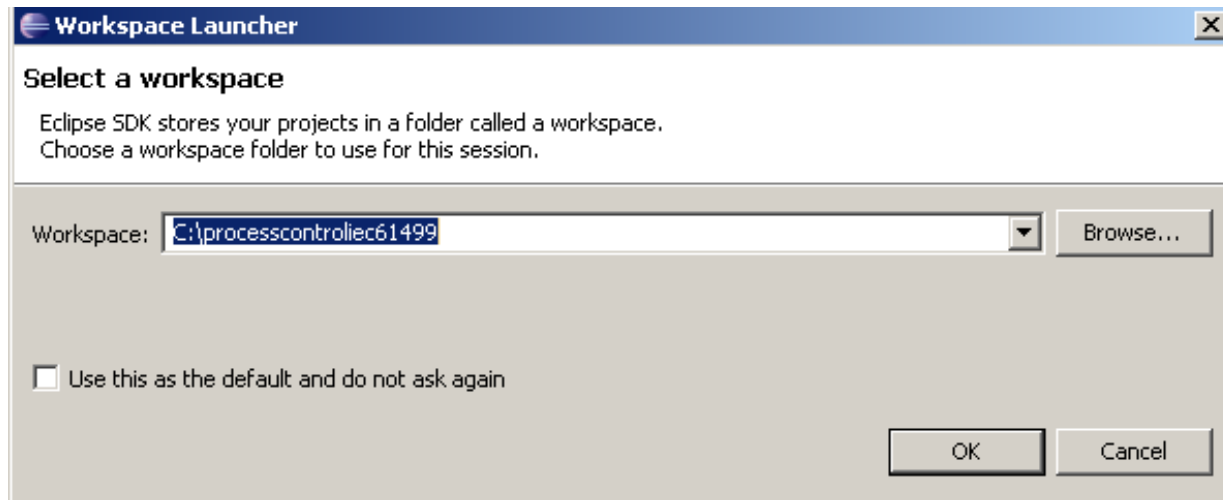
- Ficheros *.sys: Sistema en XML, bloques conectados
- Ficheros *.fbt: Bloque funcional en XML
- Ficheros *.java: Bloque funcional, algoritmos en java
- Ficheros *.class: Versión compilada de los *.java
- Librerías de clases de java se organizan en ficheros con la extensión *.jar

Copia de bloques, sistemas y funciones en java en otros ordenadores

- Copiar *.fbt en la carpeta adecuada
- Copiar *.sys en la carpeta adecuada
- Incluir en el fichero map.ini:
 - ReadOPC_USB=template/ReadOPC_USB.fbt
 - DEPO_PROCON_USB=/DEPO_PROCON_USB.sys
- Copiar las clases con código modificado desde eclipse en la carpeta adecuada
 - Java\fb\rt*.java

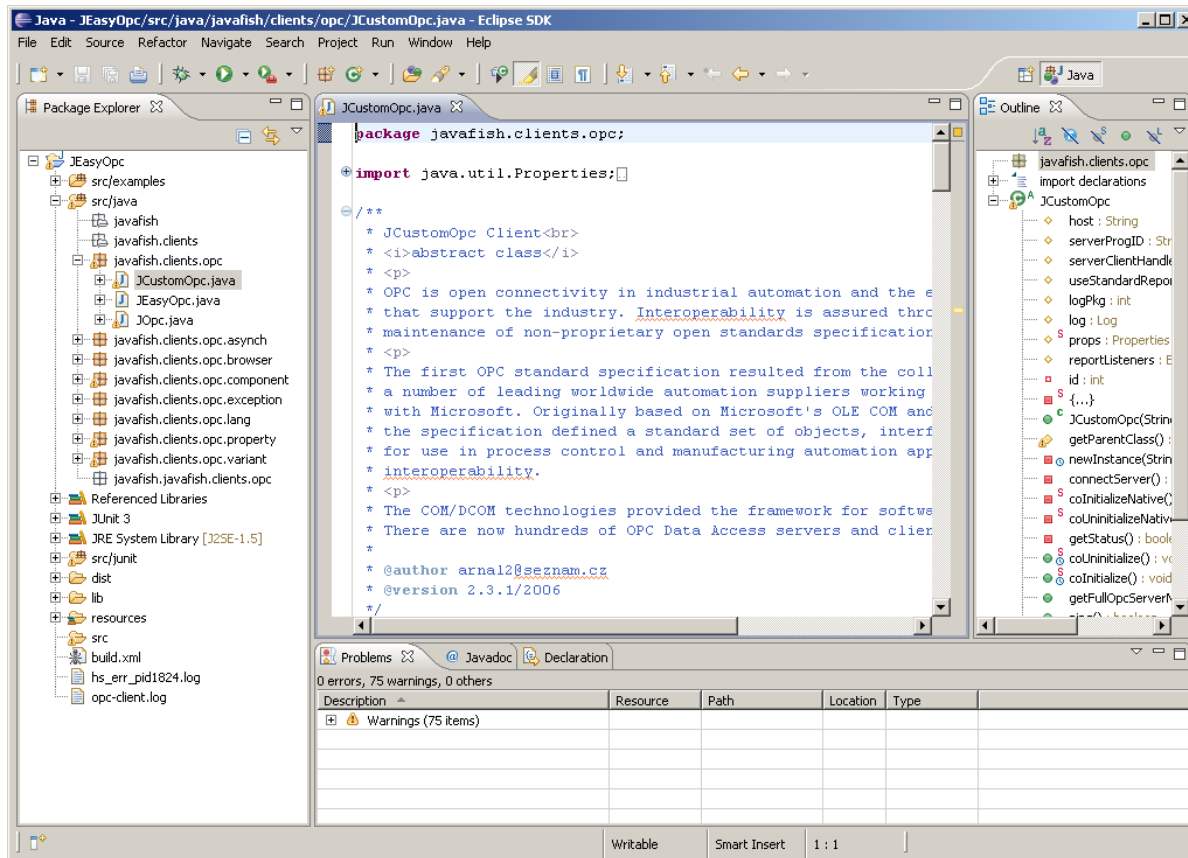
Integración con Eclipse

- Entorno de desarrollo en Java con acceso directo a FBDK
- Proyecto eclipse incluyendo carpeta del fbdk



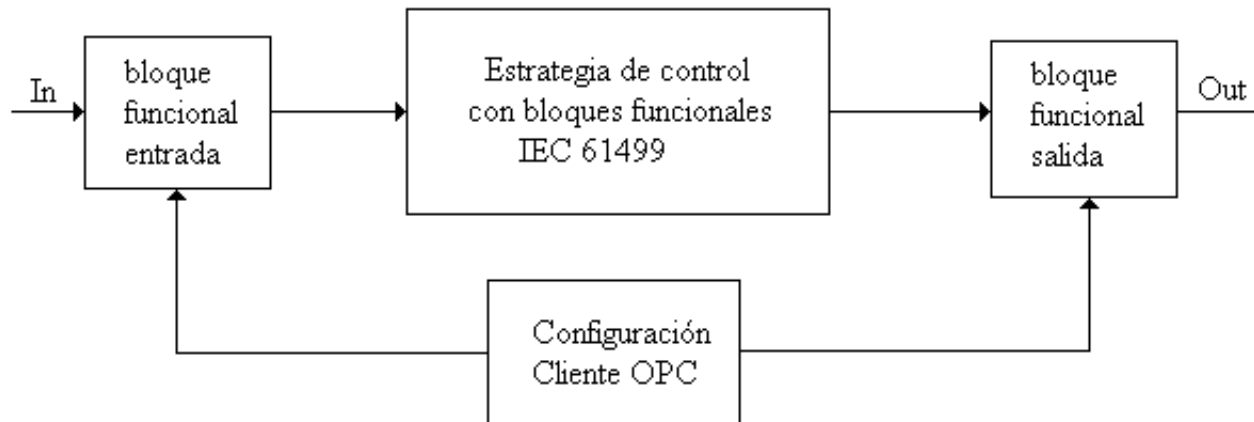
Entorno de desarrollo de Eclipse

- Algoritmos de los bloques funcionales accesibles desde Eclipse



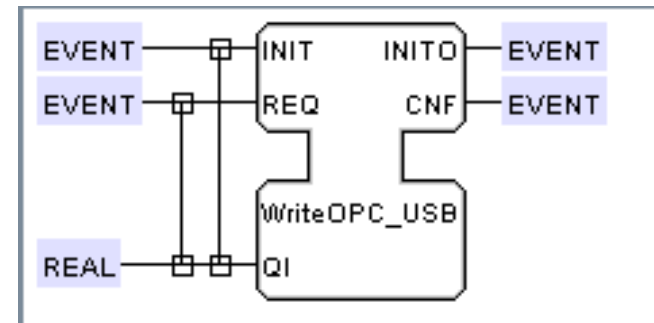
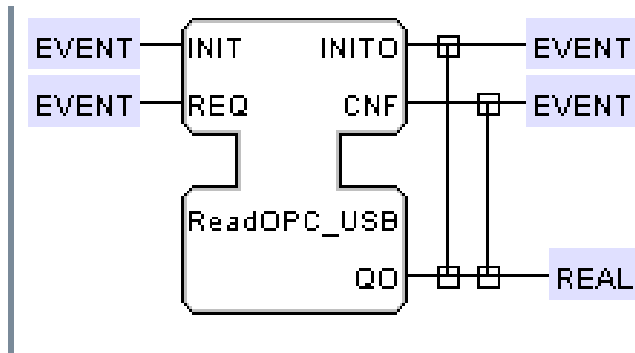
Bloques funcionales OPC

- Integración de cliente OPC open source JEasyOPC para adquisición de datos en la estructura de bloques funcionales del FBDK
 - Bloques funcionales Read_OPC, Write_OPC
 - Entrada y salida de datos mediante conexiones de los bloques funcionales implementados
 - Acceso a servidores y datos concretos en código Java



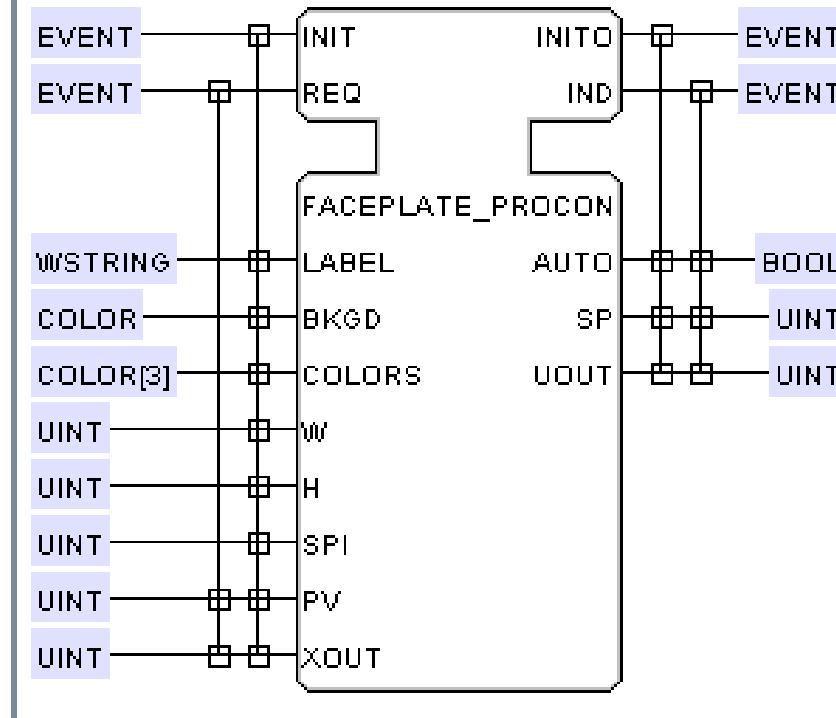
Bloques funcionales OPC

- Lectura y escritura en campo con tarjetas de adquisición en rack y por USB



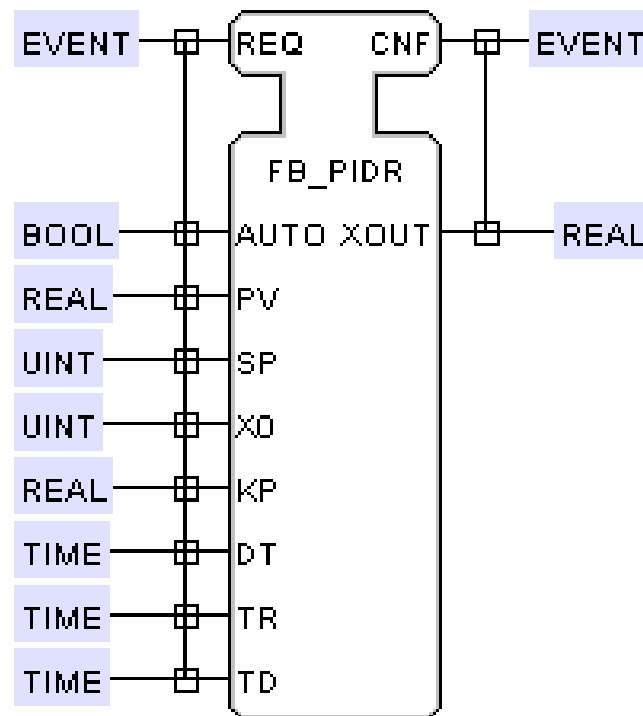
FACEPLATE_PROCON

- Modificación del FB FACEPLATE, para que en manual el slider aparezca sobre la variable manipulada, y en automático sobre la referencia



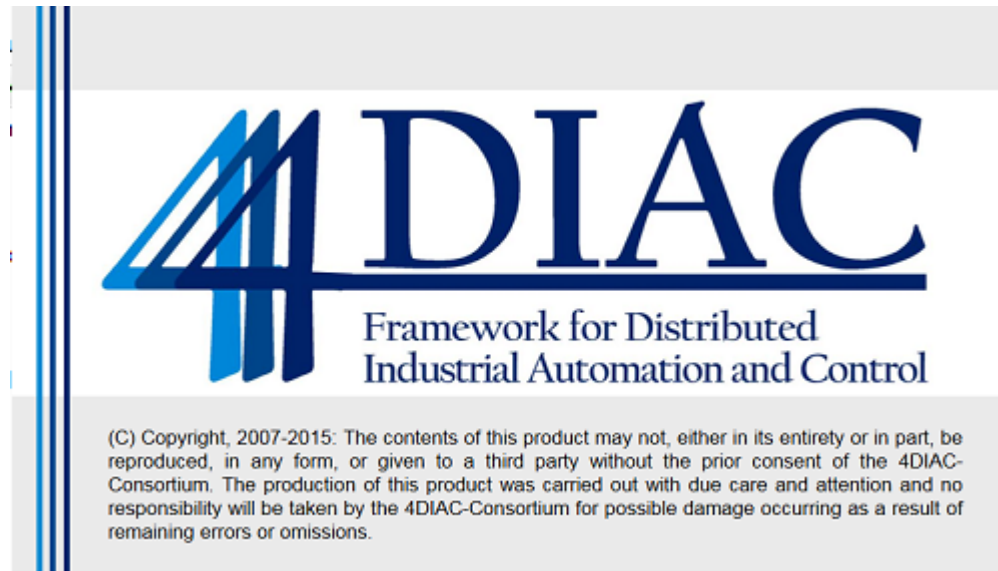
FB_PIDR

- Bloque funcional con algoritmo de control



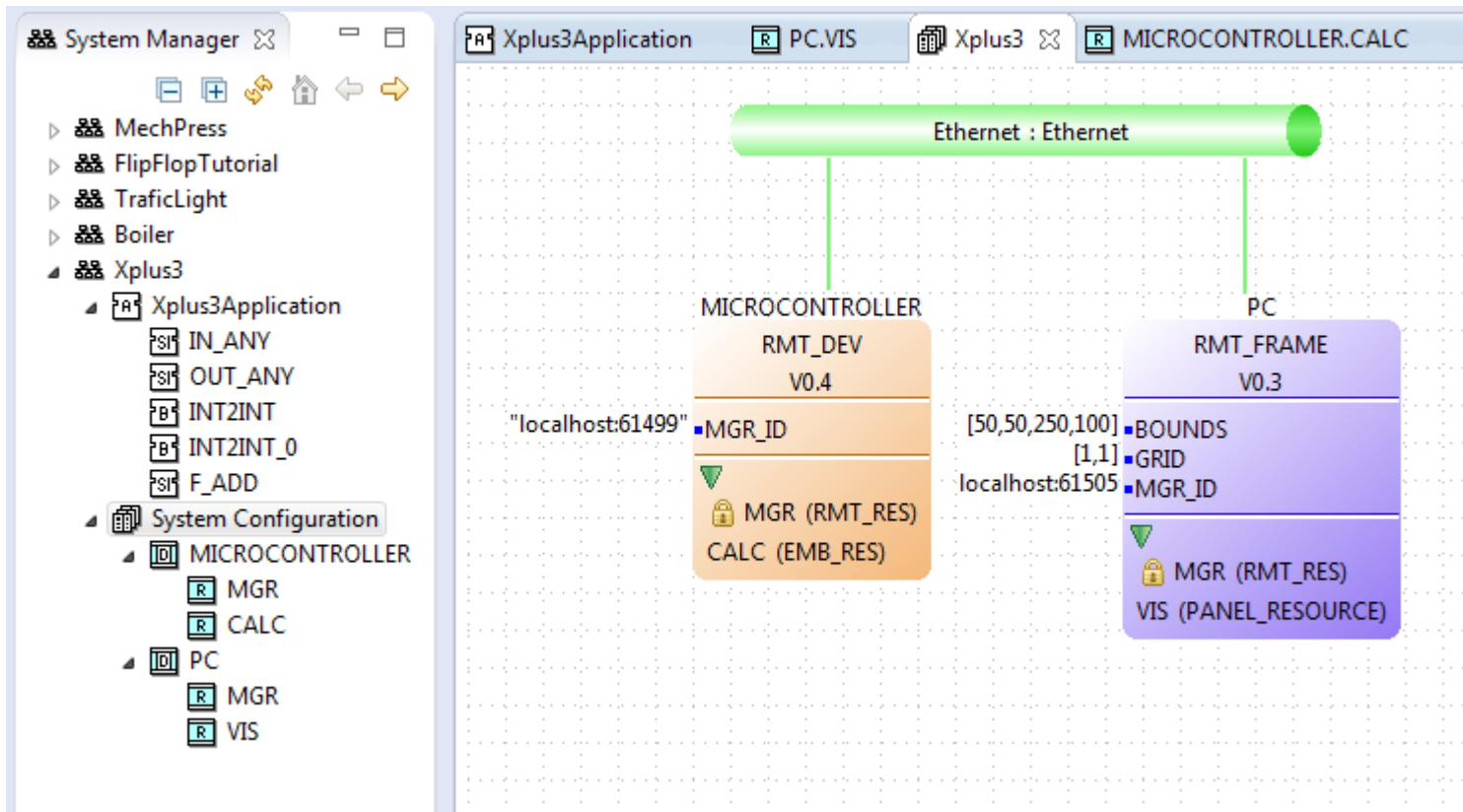
4-DIAC

- Entorno de desarrollo de bloques funcionales con clases C++ y Java



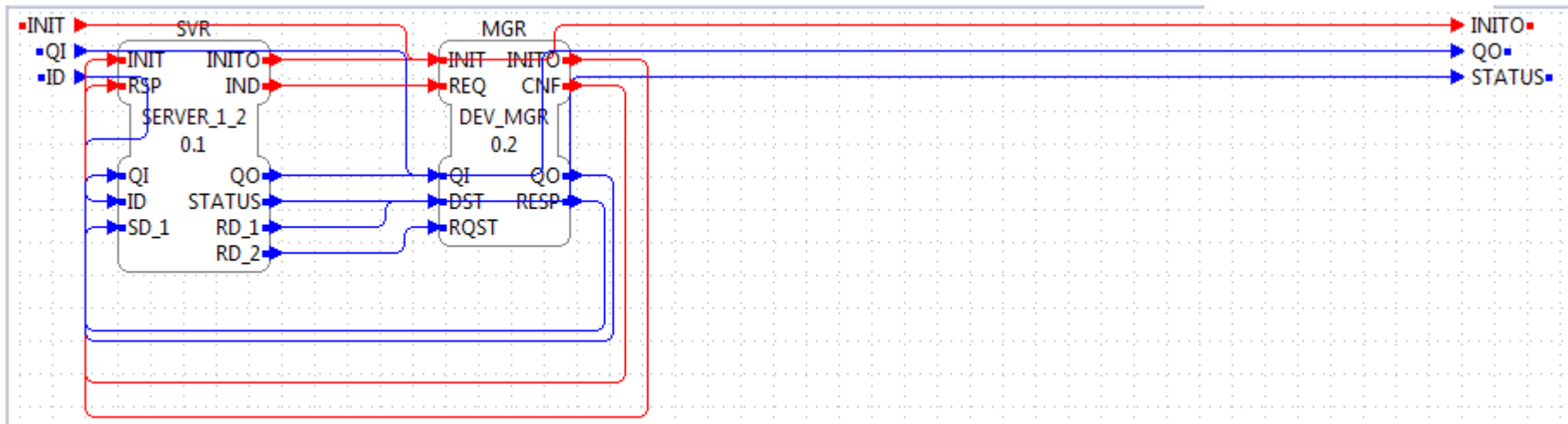
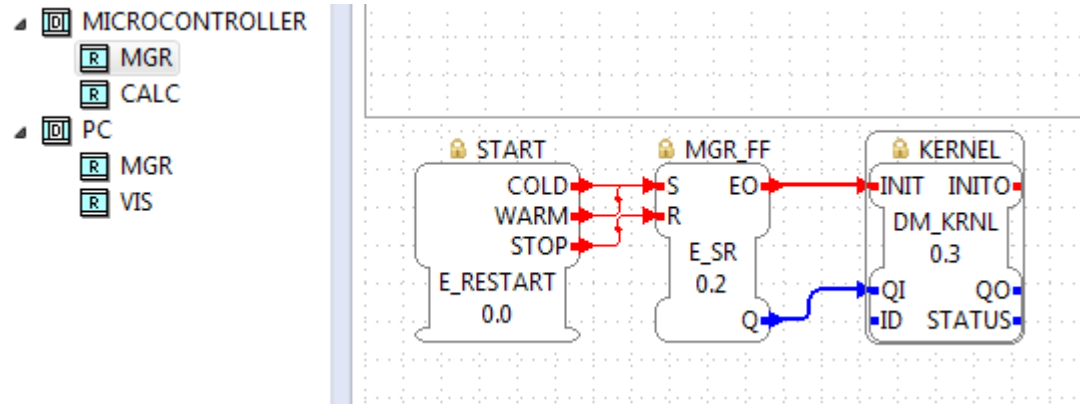
4-DIAC

- System Configuration con los dispositivos locales y remotos



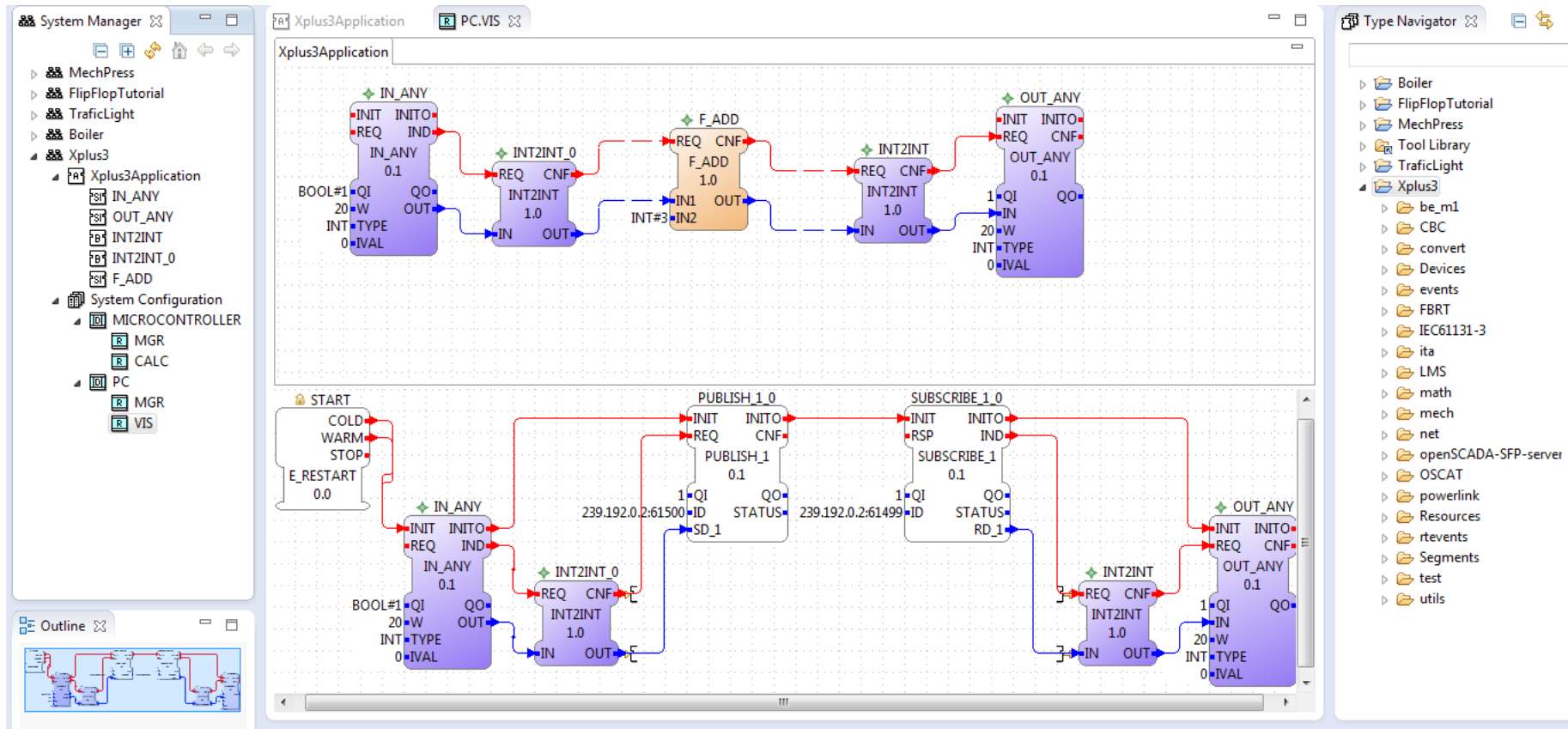
Recurso MGR

- Implementación según el estándar IEC 61499



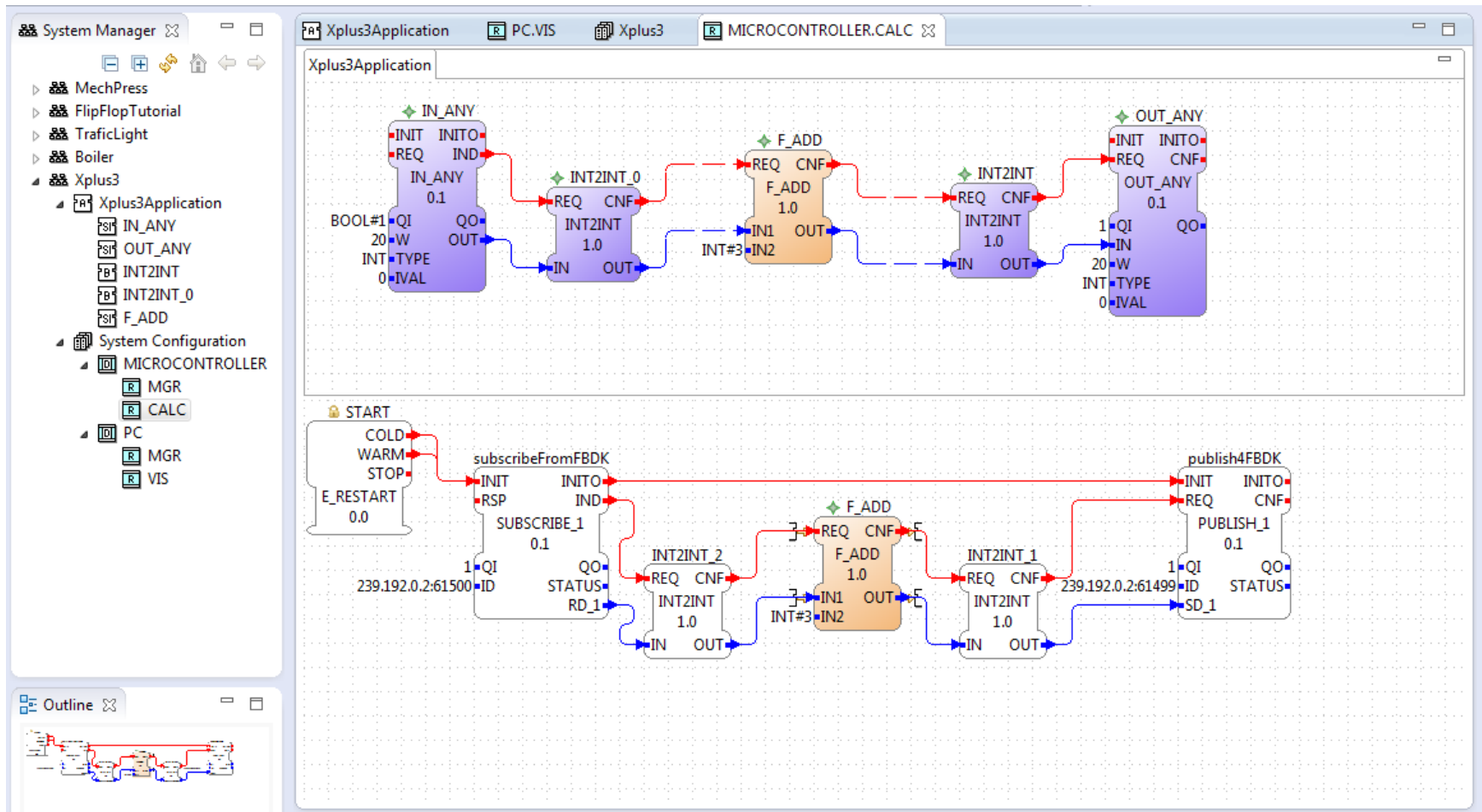
4-DIAC

- Simultaneidad en el acceso a:
 - la FBN de la aplicación
 - La FBN del recurso incluyendo los SIFB de comunicaciones



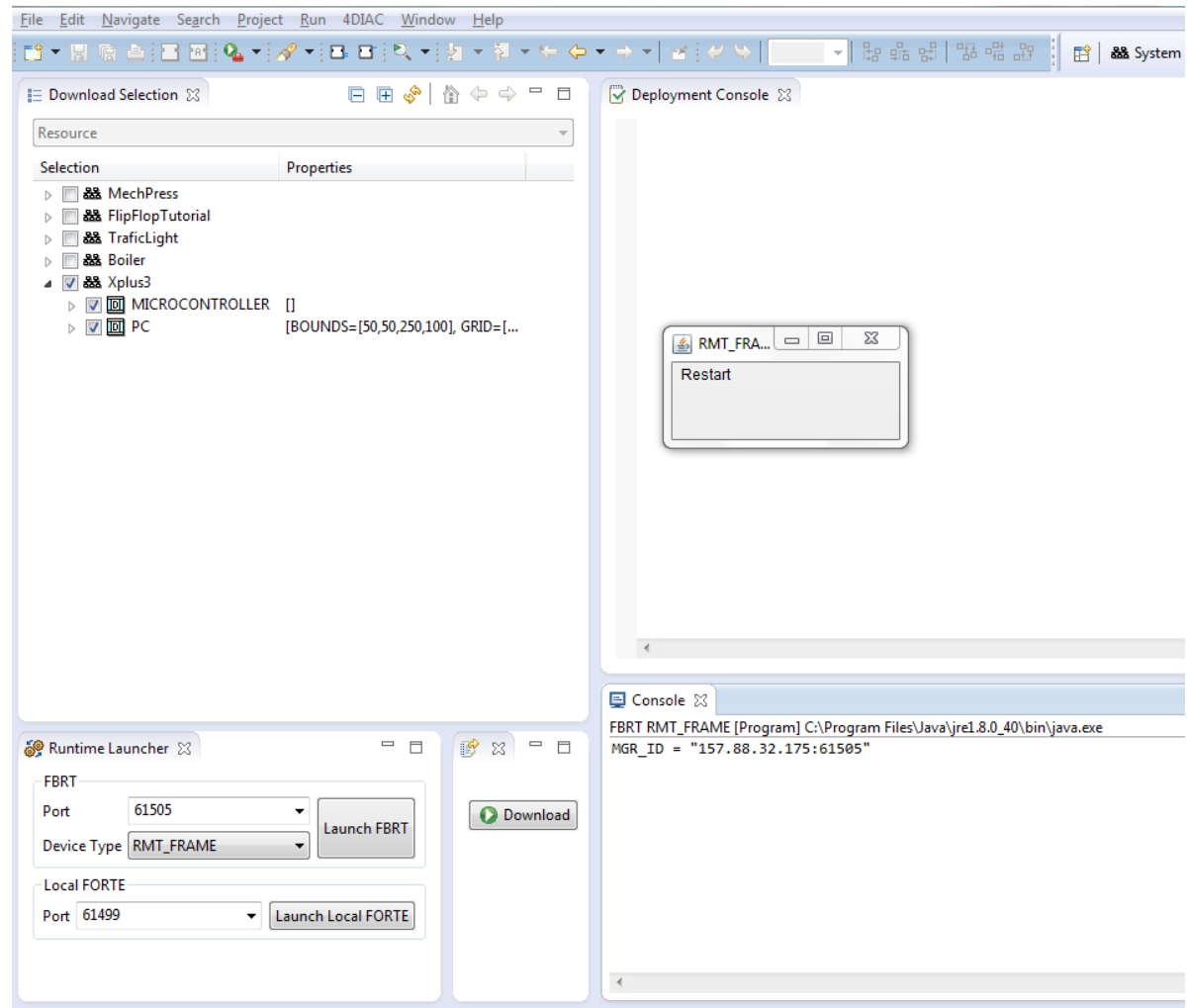
4-DIAC

- Simultaneidad en el acceso a:
 - la FBN de la aplicación
 - La FBN del recurso incluyendo los SIFB de comunicaciones



Deployment / Launch

- Runtime nativo FORTE
- FBRT para dispositivos que incorporen bloques de interfaz de usuario
 - Microcontroller con FORTE
 - PC con FBRT



Deployment / Download

The screenshot displays a software interface with several panels:

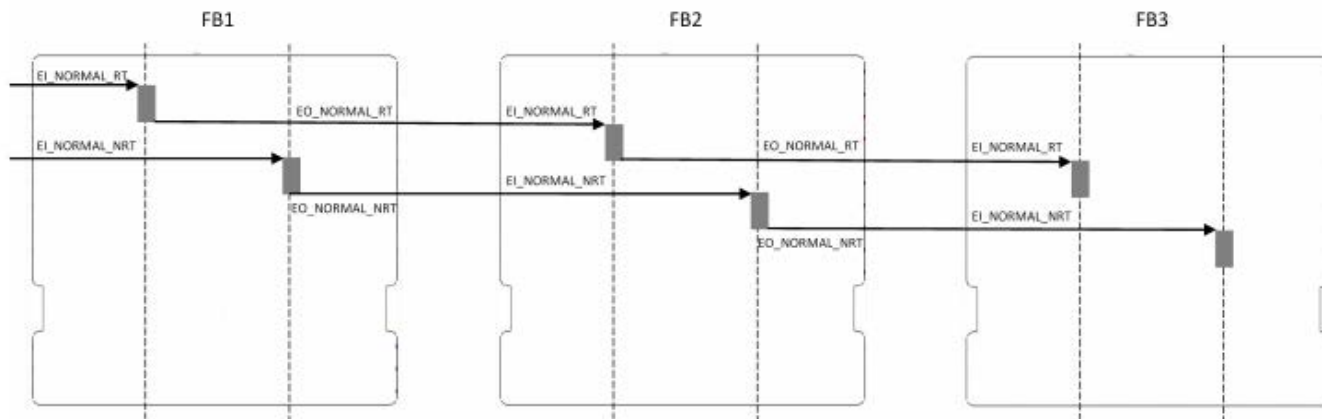
- Download Selection:** A tree view showing resources. The 'Xplus3' resource is selected, with sub-resources 'MICROCONTROLLER' and 'PC' also selected. The 'PC' resource has properties: [BOUNDS=[50,50,250,100], GRID=[...].
- Deployment Console:** A window showing XML-like output from a deployment process. The output includes request and response tags for 'CREATE' and 'WRITE' actions, with IDs and names like 'CALC', 'F_ADD', and 'F_ADD.IN2'. Comments indicate the source IP is 127.0.0.1:61499.
- Runtime Launcher:** A panel with two sections: 'FBRT' and 'Local FORTE'. The 'FBRT' section has a 'Port' dropdown set to 61505 and a 'Device Type' dropdown set to 'RMT_FRAME'. The 'Local FORTE' section has a 'Port' dropdown set to 61499. There are 'Launch FBRT' and 'Launch Local FORTE' buttons, and a 'Download' button.
- Console:** A window at the bottom showing the command prompt output: 'FBRT RMT_FRAME [Program] C:\Program Files\Java\jre1.8.0_40\bin\java.exe' and 'MGR_ID = "157.88.32.175:61505"'. A small 'RMT_FRAME' dialog box is also visible, showing a 'Restart' button and a list with items 4 and 7.

COSME

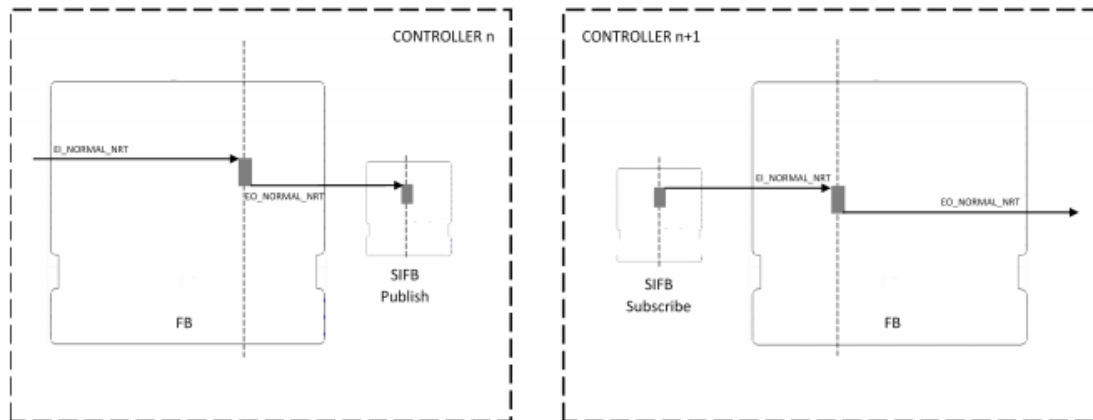
- COntrol System and Modelling Environment
- Universidad de Zaragoza
- No implementa IEC61499 en su totalidad
- Tiempo real inter-controlador
- Implantado en TUROMAS (máquina herramienta para corte, almacenamiento y manipulación de vidrio)
- IDE Domiciano

Ejecución en tiempo real en COSME

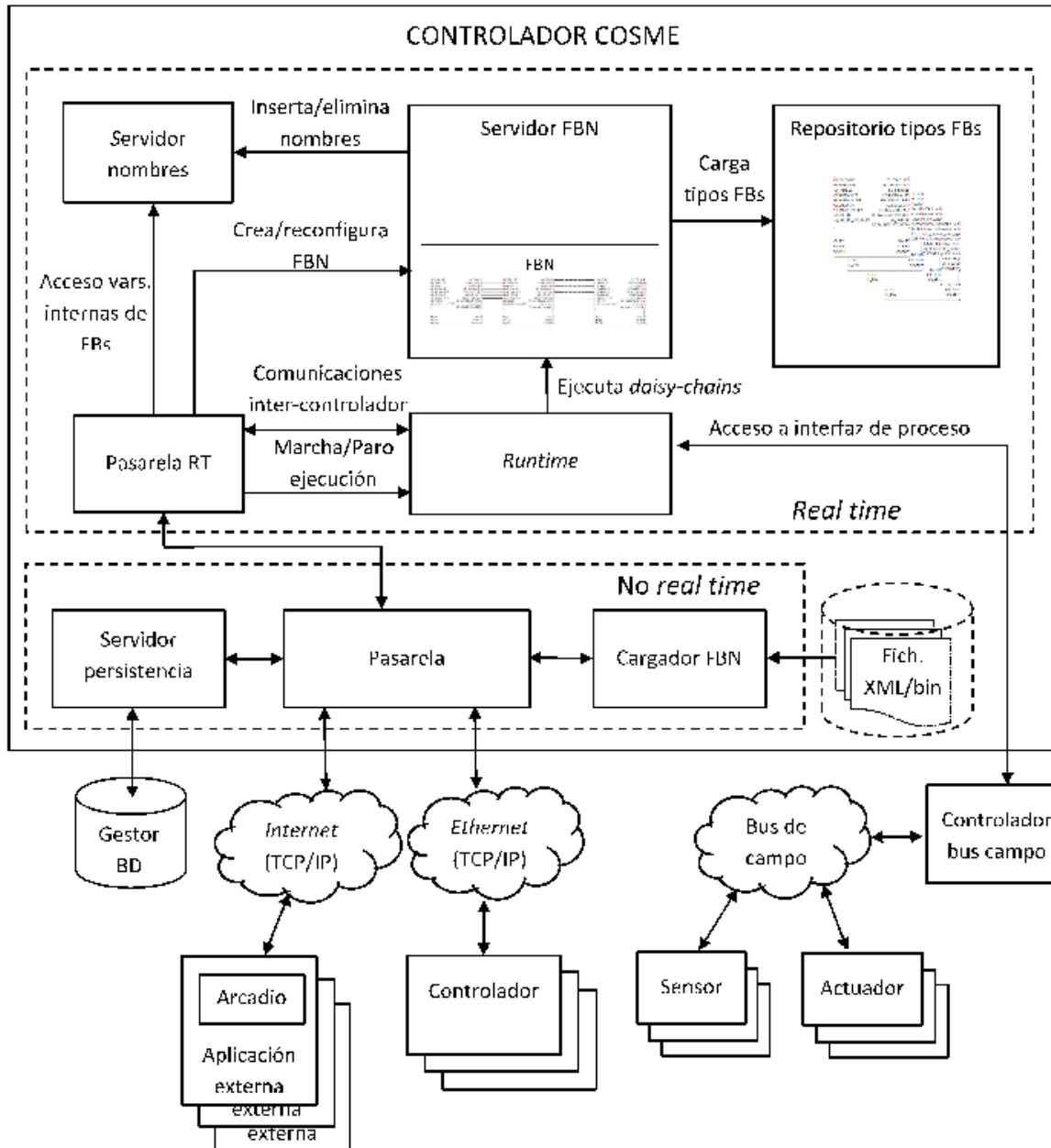
- Basado en ejecución por eventos daisy-chain intra-controlador planificada con deadlines
 - Versiones RT_Linux: RTAI, PREEMPT_RT



- Se generaliza a la ejecución inter-controlador



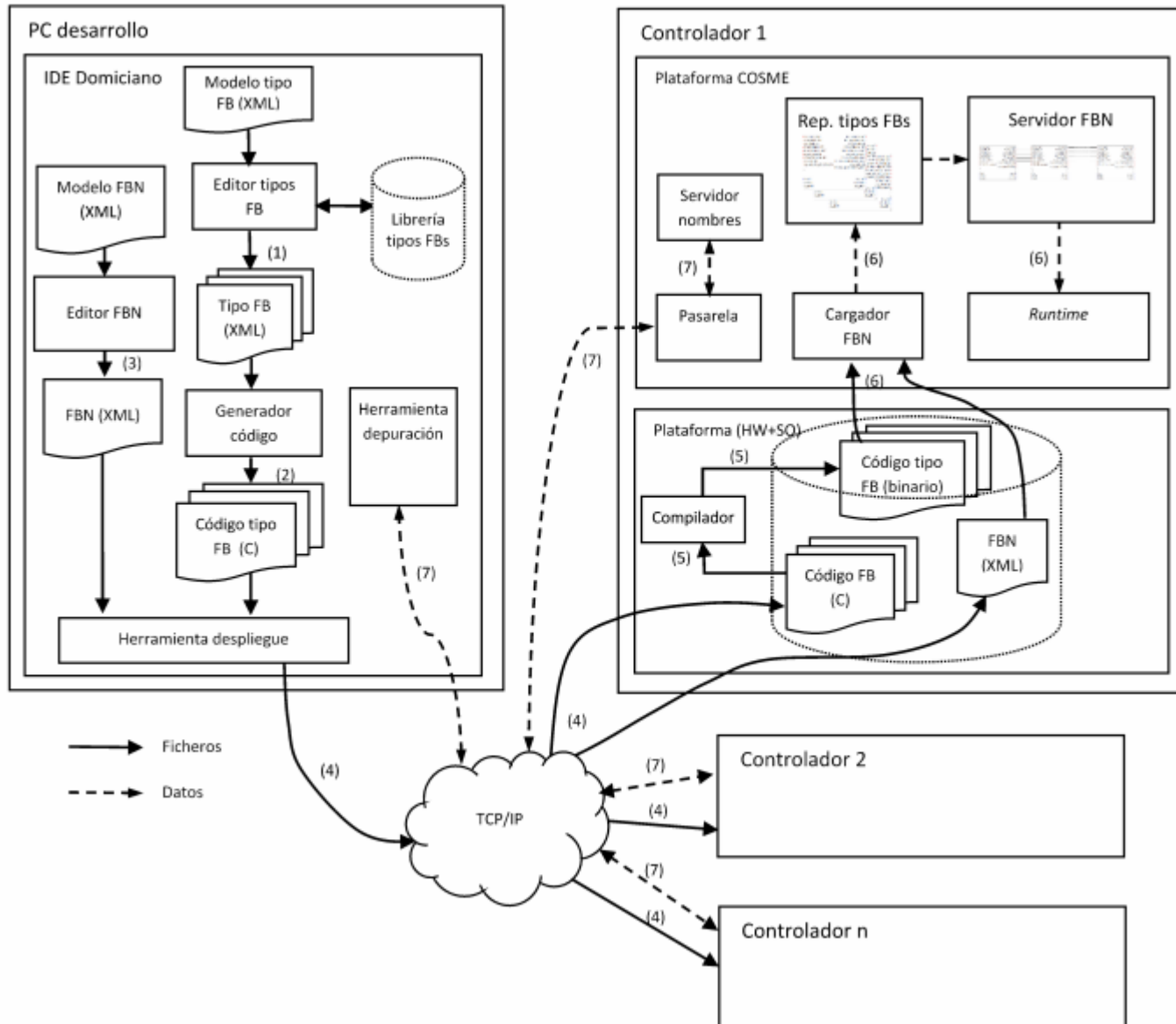
Arquitectura controlador COSME



IDE Domiciano de COSME

- Funcionalidades
 - Diseñar FBN con FB en repositorios y diseñados
 - Desplegar FBN sobre los controladores
 - Depurar el funcionamiento de FBN
- Implementado en Java
- Editores gráficos y de código C mediante bibliotecas con licencia GPL del IDE de Java Netbeans
- Despliegue con protocolos SAMBA o SCP
- Aplicaciones de ejecución remota de comandos Linux como plink

IDE Domiciano de COSME



IDE Domiciano de COSME

The screenshot displays the COSME Domiciano IDE interface. The main window shows a project configuration for 'Proyecto1' with various components and their connections. The components are organized into a hierarchy, with 'rscl' components being the primary focus. The components shown include:

- rsclLow_x2**: Inputs include 'input_struct_digital_i', 'd_i: _struct_d', and 'value: _jm'. Outputs include 'd_j: _struct_d'.
- rsclGearbox_x2**: Inputs include 'pos2: _struct_numeric_sign', 'motor: _struct_axi', and 'averia_zona: _struct_averia_zo'. Outputs include 'averia: _struct_averia_com' and 'cnc: _struct_ax'.
- rsclCnc_pid_x2**: Inputs include 'limit_switch_1: _struct_d', 'zero_slow_switch: _struct_e', 'limit_switch_0: _struct_d', 'zero_switch: _struct_d', 'averia_zona: _struct_averia_zo', and 'motor: _struct_axi'. Outputs include 'warning: _struct_averia_com', 'sto_out: _struct_di', 'averia: _struct_averia_com', and 'cncf: _struct_cnc_constr'.
- rsclGearbox_x**: A central gearbox component.
- rsclLow_x**, **rsclZero_x**, **rsclFcx0**, and **rsclFcx1**: Additional control and feedback components.
- rsclCnc_pid_x**: A central PID controller component.

The connections between these components are shown as lines with arrows, indicating the flow of data and control signals. The interface also includes a menu bar (File, Edit, View, Generate, Options, Help), a toolbar, and a status bar at the bottom with the text: 'CTRL + left mouse button - Connection with forwards: CAPS + left mouse button'.

The Messages window at the bottom shows the following text:

```
Reading names list from file '/usr/cosme/apps/Proyecto1/configs/platea/blas_nombres.xml'...  
'/usr/cosme/apps/Proyecto1/configs/platea/blas_nombres.xml': 754 names.
```

Reconfiguración en IEC 61499

- La adaptabilidad y la reconfiguración son dos de los requisitos principales de la arquitectura IEC 61499
- Cambios en los productos o los recursos

Reconfiguración de aplicaciones de control IEC 61499

- Cambios en la funcionalidad del programa de control (Ej, en la secuencia de una máquina)
- Ampliación de funcionalidad al programa de control añadiendo componentes de software o recableando los existentes
- Reemplazar componente de software (Ej, mejora de algoritmos)
- Cambios de parámetros en los componentes de software (Ej, ajuste de un algoritmo de control a condiciones cambiantes del entorno)
- Relocalización de un componente software en otro dispositivo de control (por caída o sobrecarga de un dispositivo)
- Eliminación de un componente software (Ej, producto que deja de producirse)

Demandas de la aplicación

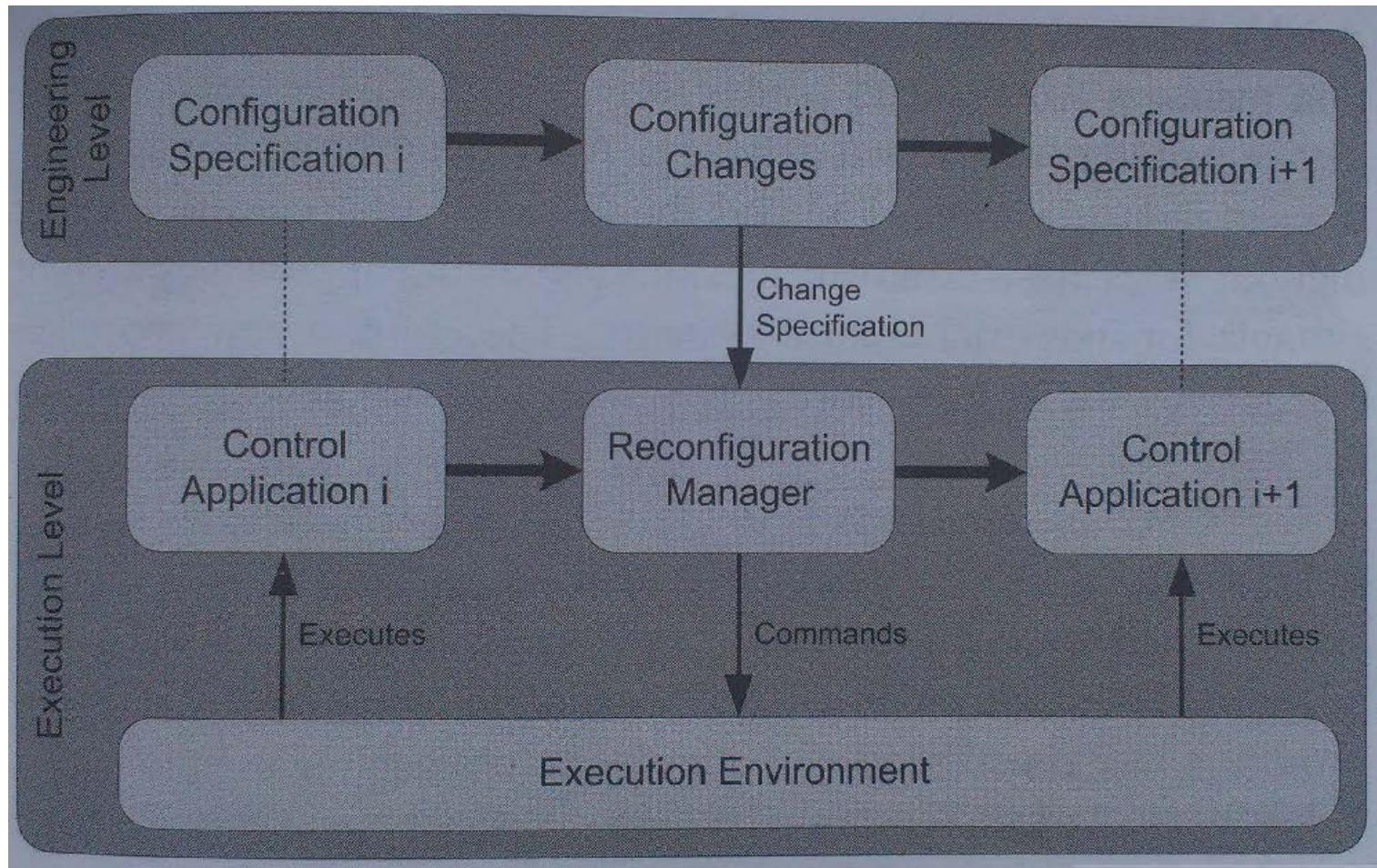
- Objetivo: Cambio del sistema de control ejecutando la reconfiguración
- Demandas específicas de la aplicación:
 - Cómo y cuándo se transfiere la información entre los componentes software
 - En qué estado de la aplicación se puede llevar a cabo la reconfiguración

Modelo de reconfiguración y Especificación de Configuración

- Describe el procedimiento del proceso de reconfiguración
- Una aplicación de control se define por su especificación de configuración
- El proceso de reconfiguración toma una aplicación de control existente y la transforma en otra.
- El modelo de sistema IEC 61499 se corresponde con la especificación de configuración
- El diseño del control modela los cambios en la configuración durante el proceso de ingeniería
- Combinación de la especificación de configuración original y los cambios de configuración resultan en una nueva especificación de configuración

Gestor de reconfiguración

- Los cambios de configuración se ejecutan en el gestor de reconfiguración en tiempo de ejecución
- El gestor de reconfiguración interactúa con el entorno de ejecución del sistema de control a través de comandos especiales o servicios de reconfiguración
- Cuando el gestor de reconfiguración finaliza su ejecución, la aplicación de control se ha transformado en una nueva
- La nueva aplicación de control está a cargo del control del sistema a partir de ese momento



Decisiones del Gestor de reconfiguración

- Cuándo aplicar los cambios: considerar los estados de la aplicación y los estados de los FBs.
- Cómo aplicar los cambios: considerar las propiedades específicas de la aplicación
- En qué secuencia aplicar los cambios: mantener el sistema en su conjunto consistente en todo momento

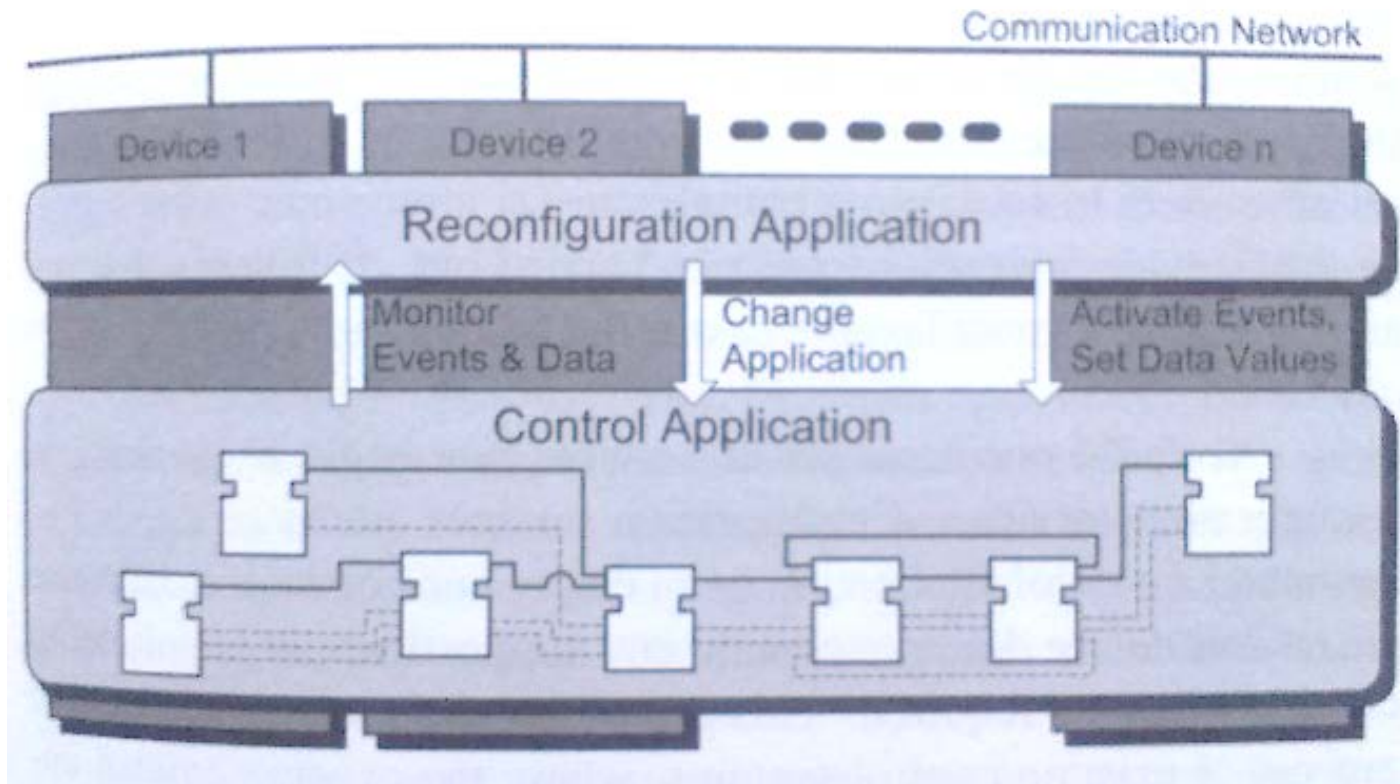
Aplicación de Reconfiguración (RCA)

- Plantear el Gestor de Reconfiguración como una aplicación especial programada por el usuario
- RCA: aplicación programada para cada proceso de reconfiguración
- Cada proceso de reconfiguración tiene un gestor de reconfiguración especializado con las particularidades de cada caso (Log)
- La RCA monitoriza las conexiones de eventos y datos de la aplicación de control bajo reconfiguración

RCA

- Invoca servicios de reconfiguración para
 - Cambiar la aplicación
 - Activar eventos en la aplicación de control
 - Cambiar valores y conexiones de datos
- Dado que la aplicación bajo reconfiguración puede ser distribuida, la RCA debe ser también una aplicación distribuida

RCA



Programación de RCAs

- Basada en componentes para ciertas tareas de reconfiguración
- Reutilizables en diferentes RCAs
- Ej.: Reemplazar FB
- Programar RCAs consiste en definir la temporización y secuencia del proceso de reconfiguración utilizando los componentes adecuados

Servicios de configuración y MFBs

- Los necesarios para descargar una aplicación de control a un dispositivo de control vacío
- Comandos Create y Write para cada tipo de elemento
- Se utilizan Bloques Funcionales de Gestión (MFB) al efecto
- Se generalizan a la reconfiguración cuando en el dispositivo ya reside una aplicación de control, con comandos y MFB inversos

Servicios de reconfiguración

- Grupo de servicios que pueden modificar todos los elementos de la aplicación de control IEC 61499
 - Recursos
 - FBs
 - Conexiones de eventos
 - Conexiones de datos
 - Parámetros
- Independientes de la aplicación que están reconfigurando
- La secuencia de servicios de reconfiguración constituye la RCA

Consistencia de la aplicación de control

- La aplicación de control permanece consistente si en ningún momento usa datos inválidos y no se pierden señales de proceso que afecten a la aplicación de control
- El diseño de la reconfiguración debe tener en cuenta la consistencia, y no está asegurada de antemano para todos los casos y técnicas
- Se deben tener en cuenta los estados de los FBs

Entorno de ejecución para las aplicaciones de reconfiguración

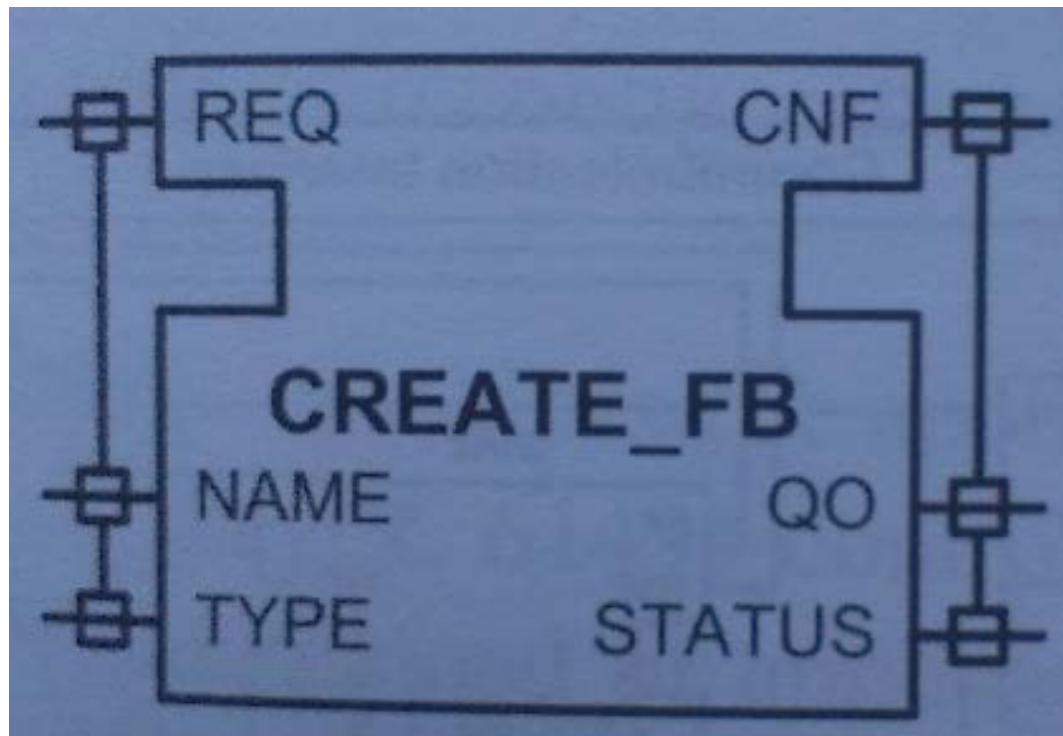
- IEC 61499 puede usarse para programar RCAs
- Las tareas de reconfiguración pueden proveerse en la forma de librerías de FBs estandarizados
- Los mismos mecanismos que se usan en los dispositivos para ejecutar las aplicaciones de control pueden usarse para ejecutar RCAs
- Las RCAs pueden diseñarse de forma complementaria a las aplicaciones de control

RFBs

- (Reconfiguration Function Blocks)
- Permiten optimizar el tiempo de ejecución de la RCA
- Encapsulan cada servicio de reconfiguración en FBs individuales, que son SIFBs que toman los datos en un formato más adecuado al entorno de ejecución
- Interactúan con el entorno de ejecución subyacente como lo hacen los DEV_MGR

RFBs

- Ej.: Bloque funcional CREATE_FB



Modelo de gestión

- Incluye el ciclo de vida completo de un elemento desde la creación, pasando por la ejecución, hasta el borrado
- Los componentes de gestión proveen la funcionalidad de los servicios de reconfiguración
- Hay dos modelos de gestión:
 - Modelo de Gestión Compartida
 - Modelo de Gestión Distribuida

Modelo de Gestión Compartida

- Un componente por dispositivo maneja todos los recursos y las FBNs localizadas en esos recursos
- Existe un único punto de acceso para RCAs
- Tareas de gestión diferentes se mezclan y se debe cuidar la consistencia: Tiempos de bloqueo que prolongan el tiempo de ejecución

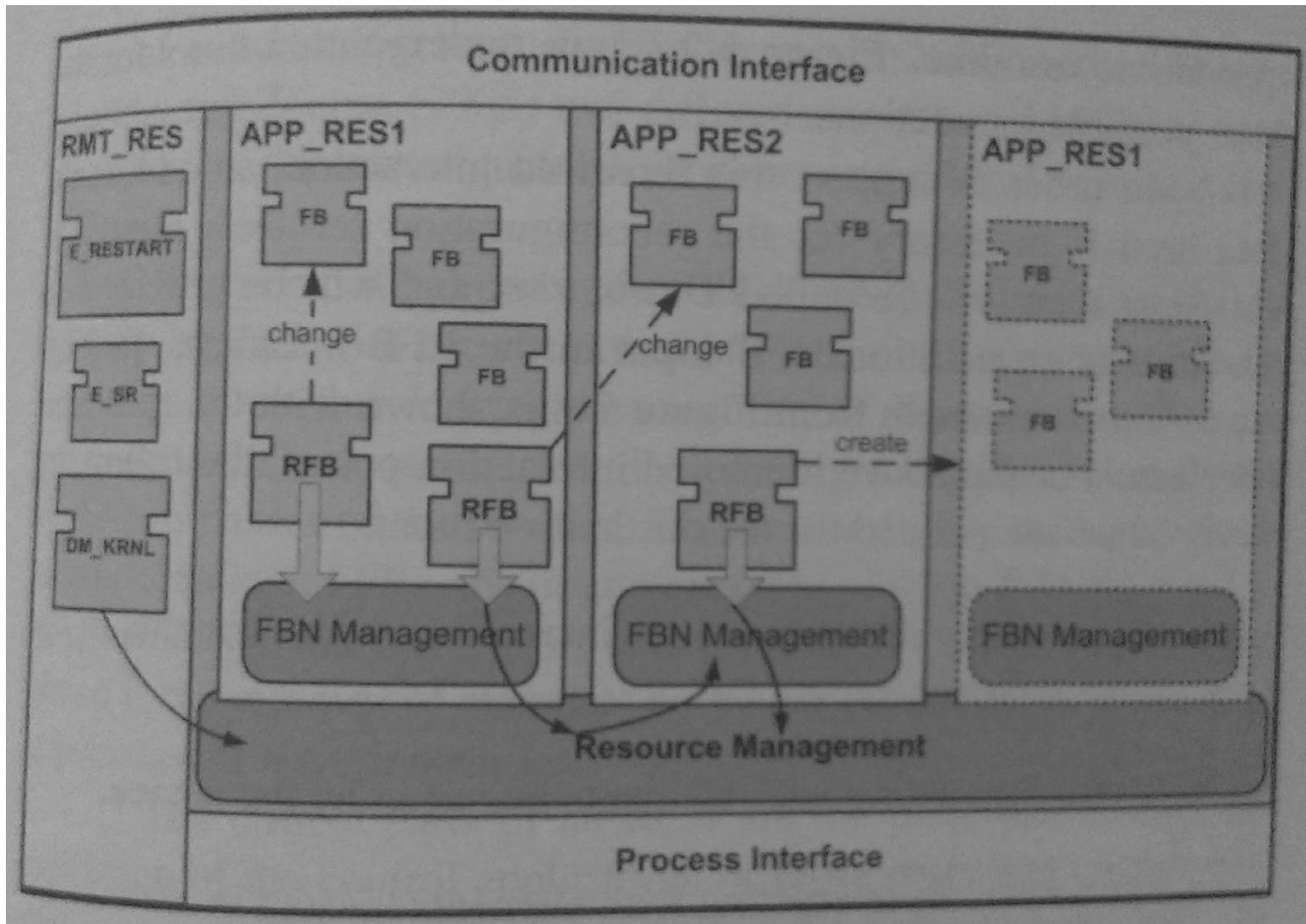
Modelo de Gestión Distribuida

- Cada recurso tiene un componente para gestionar la FBN dentro de él.
- Varias reconfiguraciones pueden ejecutarse en paralelo
- Cada componente de gestión necesita menos funcionalidad

Modelo de Gestión Distribuida

- El componente de gestión decide si ejecuta la petición o la pasa al componente de gestión del dispositivo, que tiene acceso a todos los recursos del dispositivo
- Se añade un dato de entrada a los RFBs para el campo Destino (DST)

Modelo de Gestión Distribuida



Ejecución de las aplicaciones de reconfiguración

- El proceso de reconfiguración tiene lugar durante la ejecución de la RCA
- La RCA actúa directamente con las aplicaciones de control con restricciones de tiempo real
- La RCA debe cumplir restricciones de tiempo real para perturbar lo menos posible las aplicaciones de control

Ejecución de las aplicaciones de reconfiguración

- Aproximación estructurada y sistemática a las RCA, con componentes reutilizables
- Agrupación de tareas de reconfiguración
- Dos partes en los procesos de reconfiguración:
 - Partes no críticas
 - Partes críticas

Partes no críticas de una RCA

- Partes que no interactúan directamente con la ejecución de la aplicación de control bajo reconfiguración
- Dejan la ejecución de la aplicación de control sin cambios
- Fases del proceso de reconfiguración:
 - setup: preparación
 - shutdown: limpieza

Partes no críticas de una RCA

- Fase de setup:
 - Creación de FBs
 - Interconexión de FBs
 - Escritura de parámetros
 - ...
- Fase de shutdown:
 - Borrado de elementos que no continúan conectados en la aplicación de control (FBs, conexiones, ...)

Partes críticas de una RCA

- Partes que modifican la ejecución de la aplicación de control
- Partes que cambian elementos accedidos durante la ejecución de la aplicación de control
- La modificación de la ejecución de la aplicación de control solo es posible añadiendo o eliminando un conexión de evento o cambiando el estado de ejecución de un FB
- Las modificaciones solo son críticas si cambian elementos que forman parte de la aplicación de control

Partes críticas de una RCA

- Tipo especial de cambios en parámetros y estados: fase de conmutación o switch, en el paso de la vieja a la nueva aplicación de control

Estructura general para la secuencia de una RCA

- Fase setup: Prepara la aplicación de control para su reconfiguración
- Fase de ejecución: Conmuta la ejecución a las partes nuevas creadas
- Fase shutdown: Limpia y se queda solo con las partes que permanecen de la aplicación original de control

Conclusiones

- Restricciones de tiempo real integradas en la operación de reconfiguración
- Es posible la reconfiguración dinámica en casos reales, todas las veces que haga falta en forma de aplicaciones de control intermedias entre la vieja y la nueva
- Complejidad de diseño de las RCAs: con interfaz de reconfiguración externa, 32 FBs para cambiar un FB por 3 FBs en la aplicación de control, aparte de la secuencia y temporización
- Desarrollo de herramientas de soporte que ayuden a manejar la complejidad del diseño de la reconfiguración
 - Ej.: Servicio de reconfiguración para reemplazar un FB por otro