



Capítulo III

SISTEMAS ABIERTOS (ARQUITECTURA)

1.- DESCRIPCION GENERAL

Para la implantación de un sistema abierto de control, como resultado de la integración de uno existente con tecnología de entorno distribuido, deberá tener en cuenta los elementos del sistema en servicio actual (mecánicas, eléctricas, software, protocolos, hardware, etc..).

Este acercamiento es una estrategia de negocio e ingeniería para escoger, adoptando especificaciones y normas por cuerpos de normas de industria o de facto normas (conjunto por el lugar de mercado) para un sistema, eligiendo interfases (funcional y físicos), productos, prácticas y herramientas.

Elegido éste, deberá estar basado sobre ejecución, costes, aceptación de industria, ciclo de vida largo, mantenimiento mínimo, y poder actualizar su potencial de forma totalmente autónoma.

El Sistema deberá tener una total transparencia de Tecnología para una rápida actualización del mismo.

En resumen, un sistema abierto debe implementar bastante bien las especificaciones para interfases, servicios, soportando formatos abiertos para habilitar de manera apropiada la utilización de componentes ingeniados a través de un rango ancho de sistemas con cambios mínimos, para inter-operar con otros componentes sobre sistemas locales o remotos, interactuando con usuarios en un estilo que facilite portabilidad. Un sistema abierto se caracteriza por lo siguiente:

- Buena definición, no propietario usando ampliamente interfases/protocolos, y
- Uso de normas que se desarrollan/adoptan a cuerpos de normas industriales conocidas, y
- Definición de todos los aspectos de sistemas de interfases para facilitar capacidades nuevas del sistema o adicionales para un amplio rango de aplicaciones, y
- Previsiones explícitas para expansión o actualización mediante la incorporación de elementos de ejecución adicionales con el mínimo impacto sobre el sistema

En definitiva, los Sistemas Abiertos son, efectivamente, una metodología (procesos), un entorno flexible, cohesivo que será beneficioso para la competitividad entre proveedores de equipos y software. Tres componentes interdependientes son las que se requieren para crear



un verdadero entorno de sistemas abiertos: productos basados en normas/tecnología; una infraestructura de desarrollo abierta; y una directiva de gerencia.

Toda Empresa debería estar apta para decidir qué producto es conforme a su perfil de normas abiertas, y NO un producto abierto. Normalizar sobre interfases y no sobre productos. De ahí el estudio del presente trabajo: UCI como interfaz que cumple la norma para el sistema abierto de Sevillana.

Compendio de Conceptos Clave:

- Un Sistema Abierto es una estrategia de negocio para conseguir beneficios económicos.
- Un Sistema Abierto se consigue aplicando disciplinadamente una gerencia de ingeniería para abrir normas de interfases comerciales para dirigir una arquitectura de sistemas, construir perfiles para el sistema reflejando requisitos de ejecución, y agenciando el uso de productos para la construcción del sistema conforme a los perfiles a conseguir.
- Producto comercial volátil para poder estar efectivamente dirigiendo con una disciplinada ingeniería de sistemas abiertos.

2.- BENEFICIOS DE LA ARQUITECTURA DE SISTEMAS ABIERTOS

- **Inter-operabilidad:** la habilidad de que dos o más sistemas de ordenadores (ya sean parecidos o desiguales) y su software puedan intercambiar información de uso.
- **Compatibilidad:** la habilidad general de dos aplicaciones para coordinar con otra en su operación, diseñadas originalmente para hacerse así.
- **Portabilidad:** facilidad con la que un componente de aplicación o equipo pueda estar entorno al equipo o software, transfiriéndose desde una a otro.
- **Reusabilidad:** es la habilidad al reuso de aplicaciones de software/equipo en generación con otra aplicación.
- **Mantenibilidad:** calidades que mejoran la habilidad para mantener la aplicación.
- **Independencia del vendedor:** permitiendo diferentes vendedores para competencia con ofertas de equipos y software
- **Mejora de productividad para el usuario:** conseguido mediante la consistencia, estableciendo y prediciendo los servicios a pesar de la plataforma de aplicación.



- **Costes reducidos del ciclo de vida:** duplicación mediante reducción, consiguiendo funcionalidad para reunir los requisitos de vendedores diferentes (requisitos de configuración).
- **Mejora de inserción de Tecnología.**
- **Escalabilidad:** la habilidad de configurar aplicaciones que permitan operar sobre plataformas ranging desde micros a mainframes.

En definitiva, los sistemas abiertos proveen beneficios económicos, oportunidades de inserción de tecnologías, independencia de firmas comerciales proveyéndose de portabilidad, escalabilidad, interoperabilidad,...



3.- DDM CON TECNOLOGIA SIEMENS-EMPROS

3.1- OBJETO

Se consideró, por la Dirección de Desarrollo que el Despacho de Maniobra, así como los Centros Provinciales de Maniobras, habían prestado servicio desde aproximadamente más de quince años, periodo legal de amortización establecida para los equipos de telecontrol dentro del Marco Legal Estable.

La vida útil de este tipo de equipos habían sido ampliamente superada, tal que se tienen problemas de repuestos, así como de una nula capacidad de ampliación de funcionalidad a las necesidades funcionales del usuario del momento actual.

Todo lo anterior llevó a la aprobación de la renovación del Despacho de Maniobras, en primer lugar, y, posteriormente, a la sustitución y/o renovación de los Puestos Provinciales de Maniobras.

Se decide implantar en el DDM la nueva Tecnología de la firma EMPROS basada en la nueva Tecnología de Sistemas Abiertos de Control.

Una vez probada dicha renovación del DDM, se pasaría por ir expandiendo dicha tecnología a los nueve Centros Provinciales de Maniobras Desarrollando las pertinentes interfases necesarias para dicho fin (motivo de este proyecto).

3.2- PROCESADORES RISC- multiproceso y caché-

3.2.1- INTRODUCCION

A lo largo de la historia de la industria de los ordenadores, la tendencia mayormente adoptada para conseguir un aumento de prestaciones, ha sido el incremento de la complejidad de las instrucciones. Es lo que se ha denominado “computación con conjuntos de instrucciones complejas” o CISC (Complex Instruction Set Computing).

Sin embargo, la tendencia actual, se esfuerza en conseguir procesadores con conjuntos de instrucciones de complejidad reducida o RISC (Reduced Instruction Set Computing). La idea es que un conjunto de instrucciones poco complejas son simples, y por tanto de más rápida ejecución, lo que permite crear un código más “aerodinámico”.



Tanto la tecnología CISC como RISC son acreditadas a IBM, aunque sus antecesores bien pueden ser John vonNeumman (inventor del primer programa de ordenador almacenado, y que promovía la velocidad inherente a conjuntos de instrucciones reducidas), Maurice Wilkes (padre de la microprogramación y de muchos conceptos de los diseños RISC), y Seymour Cray (primeros supercomputadores, empleando principios RISC).

En 1975, IBM inició el desarrollo de un controlador para un sistema de conmutación telefónica, que aunque fue abandonado, sirvió como punto de partida para el desarrollo de una CPU con tecnología ECL, corazón del sistema 801, precursor del IBM PC RT.

Los inicios de la tecnología RISC también surgen en el ambiente académico, ya que en 1980, la Universidad de Berkeley (California), el Dr. David A. Patterson inició un proyecto denominado RISC I, que obtuvo resultados en tan solo 19 meses, seguido por RISC II, SOAR (Smalltalk on a RISC) y SPUR (Symbolic Processing on a RISC). El resultado directo, además de la educación en la ingeniería y los fundamentos del diseño de microprocesadores, fue la creación de una máquina que fuese capaz de mayores velocidades de ejecución a menores velocidades de reloj y que requiriese menores esfuerzos de diseño.

Casi simultáneamente, en la Universidad de Stanford, el Dr. John Hennessy inició también un proyecto de implementación RISC, denominado MIPS, seguido por el sistema MIPS-XMP, enfocados hacia el proceso simbólico, demostrando las capacidades de velocidad de la arquitectura RISC.

Ambos profesores se vieron envueltos rápidamente, en proyectos de productos comerciales, y en concreto, Hennessy fue uno de los fundadores de MIPS Computer Systems, mientras Patterson actuaba de asesor durante el desarrollo del primer SPARC.

Por otro lado, durante las pasadas décadas, el multiproceso, como medida de incrementar drásticamente las prestaciones de los sistemas a un coste razonable, se ha visto reducido al ámbito de los computadores de “alto nivel”, en parte debido a los bajos niveles de integración del silicio, y a la falta de software que facilitase la ejecución paralela de las aplicaciones.

Las ventajas de los procesadores RISC, especialmente las ligadas a los SISTEMAS ABIERTOS, los hacen plataformas ideales para explotar los puntos fuertes de los sistemas multiprocesadores.

3.2.2- TECNOLOGIA RISC

La comparación que antes hemos realizado entre CISC y RISC es algo simple, ya que no es sólo una cuestión de diferencias en el conjunto de instrucciones, puesto que es fundamental resaltar el mejor uso y aprovechamiento de los recursos del silicio, es decir,



menor tiempo de diseño y empleo número de transistores, lo que redundará en menor número de errores de diseño y menor tiempo de ejecución para instrucciones individuales.

Las características comunes a todos los procesadores RISC, fuente de sus capacidades de altas prestaciones, son:

1. **Modelo de conjunto de instrucciones Load/Store** (Cargar/Almacenar). Sólo las instrucciones Load/Store acceden a memoria; las demás operaciones en un RISC, tienen lugar en su gran conjunto de registros. Ello simplifica el direccionamiento y acorta los tiempos de los ciclos de la CPU, además facilita la gestión de los fallos de páginas (page faults) en entornos de memoria virtual. Además, permite un elevado nivel de concurrencia a consecuencia de la independencia de las operaciones de Load/Store de la ejecución del resto de las instrucciones.
2. **Arquitectura no destructiva de tres direcciones.** Los procesadores CISC destruyen la información que existe en alguno de los registros, como consecuencia de la ejecución normal de instrucciones; esto es debido a su arquitectura de dos direcciones, por la cual el resultado de una operación sobrescribe uno de los registros que contenía a los operandos. Por el contrario, las instrucciones RISC, con tres direcciones, contienen los campos de los dos operandos y de su resultado. Por tanto, tanto los operandos origen como el destino, son mantenidos en los registros tras haber sido completada la operación. Esta arquitectura “no destructiva” permite a los compiladores organizar las instrucciones de modo que mantengan llenos los conductos (pipelines) del chip, y por tanto reutilizar los operandos optimizando la concurrencia.
3. **Instrucciones simples, de formato fijo, con pocos modos de direccionamiento.** Las instrucciones simples reducen de manera muy significativa el esfuerzo para su decodificación, y favorecen su ejecución en pipelines. Las instrucciones de longitud fija, con formatos fijos, implican que los campos de códigos de operación (opcodes) y de los operandos están siempre codificados en las mismas posiciones, permitiendo el acceso a los registros al mismo tiempo que se está decodificando el código de operación. Todas las instrucciones tienen una longitud equivalente a una palabra y están alineadas en la memoria en límites de palabra (word boundaries), ya que no pueden ser repartidas en pedazos que puedan estar en diferentes páginas.
4. **Ausencia de microcódigo.** El microcódigo no se presta a la ejecución en ciclos únicos, ya que requiere que el hardware sea dedicado a su interpretación dinámica. La programación en microcódigo no hace que el software sea más rápido que el programado con un conjunto de instrucciones simples. Todas las funciones y el control, en los procesadores RISC, están “cableados” (hardwired), para lograr una máxima velocidad y eficiencia.
5. **Ejecución en conductos (pipelined).** Las instrucciones simples, de formato fijo y ciclo único permiten que las diferentes etapas de los ciclos de operación (búsqueda o



fecth, decodificación, ejecución, y escritura del resultado o result write-back) para instrucciones múltiples, se puedan realizar simultáneamente, de un modo más simple y eficaz.

6. **Ejecución en ciclos únicos (single-cycle).** El resultado directo de los conjuntos de instrucciones que ofrecen los procesadores RISC, es que cada instrucción puede ser ejecutada en un único ciclo de la CPU. Esto invalida la creencia de que las microinstrucciones en microcódigo, creadas para ser ejecutadas en un solo ciclo de procesador, son más rápidas que las instrucciones del lenguaje ensamblador. Ya que el caché está construido partiendo de la misma tecnología que el almacenamiento de control del microprograma, una única instrucción puede ser ejecutada a la misma velocidad que una microinstrucción. La ejecución en ciclos únicos también simplifica la gestión de las interrupciones y los conductos (pipelines).

3.2.3- CONCEPTOS DE MULTIPROCESO

La industria informática, ha tenido siempre un objetivo primordial, repetido a lo largo de toda su cadena (fabricantes de semiconductores, fabricantes de sistemas y usuarios): la búsqueda de la velocidad. Para alcanzar este objetivo se han invertido ingentes cantidades de recursos, hasta alcanzar los límites físicos del silicio.

Obviamente, la velocidad va ligada a las prestaciones, y por lo general, la primera ha sido la principal medida para decidirse por un sistema u otro. Sin embargo, por muy evidente que parezca, y dados los límites físicos de los semiconductores, las prestaciones pueden no estar forzosamente ligadas a la velocidad. Hoy es posible construir sistemas, que aún teniendo procesadores más “lentos” que otros, ofrezcan unas prestaciones significativamente superiores. Son los sistemas multiprocesador, que como su nombre indica, incorporan varios procesadores para llevar a cabo las mismas funciones.

No es un concepto nuevo, ya que los “minicomputadores” construidos por compañías como NCR, Sequent y Stratus, ya empleaban varios nodos de proceso como alternativas económicas a otros productos de otras compañías. Sin embargo, aquellos sistemas aún duplicaban recursos caros del sistema, como memoria y dispositivos de entrada/salida, y por tanto, confinaban a los sistemas multiprocesador al mundo de los sistemas de alto nivel.

Ahora, y en gran medida gracias a los procesadores de arquitectura RISC, el soporte multiprocesador es una solución integrada y fácilmente disponible en estaciones de trabajo de sobremesa, que resuelve, a través de hardware VLSI, los complejos problemas de compartición de recursos (memoria compartida) de aquellas primeras máquinas.



Evidentemente, estas mejoras en el hardware, para ser funcionales, requieren importantes desarrollos en el software, y de hecho, muchos sistemas operativos admiten extensiones multiproceso (Match, SCO, Solaris, System V, etc.), que proporcionan paralelismo “en bruto” (asignando múltiples tareas a múltiples procesadores) a nivel del sistema operativo.

Los sistemas multiprocesadores pueden ser clasificados con mayor propiedad como sistemas MIMD (Multiple Instruction, Multiple Data): múltiples secuencias de instrucciones operan, simultáneamente, sobre múltiples secuencia de datos. Ello implica que son máquinas con múltiples y autónomos nodos de proceso, cada uno de los cuales opera sobre su propio conjunto de datos. Todos los nodos de proceso, cada uno de los cuales opera sobre su propio conjunto de datos. Todos los nodos son idénticos en funciones, por lo que cada uno puede operar en cualquier tarea o porción de la misma.

El sistema en que la memoria está conectada a los nodos de proceso establece el primer nivel de distinción entre diferentes sistemas multiprocesador:

1. Multiprocesadores de memorias distribuida.
2. Multiprocesadores de memoria compartida.

Es evidente, que los sistemas actuales tienden al uso de arquitecturas de memoria compartida, fundamentalmente por razones de costes, a pesar del problema de la contienda por el bus. Los tres fuentes fundamentalmente responsables de dicha disputa son la memoria (cada CPU debe usar el bus para acceder a la memoria principal), la comunicación (el bus es usado por los “bus masters” para la comunicación y coordinación), y la latencia de la memoria (el subsistema de memoria mantiene al bus durante las transferencias de datos, y en función de la velocidad a la que la memoria puede responder a las peticiones, puede llegar a ser un factor muy significativo).

3.2.4- LOS SISTEMAS DE MEMORIA CACHE Y EL MULTIPROCESO

Los sistemas de memoria multinivel (caché) son un esfuerzo para evitar el número de peticiones realizadas por cada CPU al bus. Los caches son pequeñas y rápidas (y por tanto caras) memorias, que hacen de tampon (buffer) entre la CPU y la memoria externa, para mantener los datos y/o instrucciones. Se basan en el principio de la "localidad", lo que significa que, dada la fundamental naturaleza secuencial de los programas, los siguientes datos o instrucciones requeridas, estarán localizadas inmediatamente a continuación de las actuales.

Los datos contenidos en la memoria caché se organizan en bloques denominados líneas. Las líneas son cargadas en el caché como copias exactas de los datos situados en la



memoria externa. Para referenciar a los datos de la memoria caché, se emplean marcas (tags) que identifican a cada línea. Las marcas o tags emplean una porción de la dirección física de los datos, para compararla con la dirección física solicitada por la CPU. Cuando existe una coincidencia exacta de la dirección y de otros cualificadores (estado, privilegio, contexto, etc.), se dice que ha tenido lugar un acierto (hit) de caché; en caso contrario, tiene lugar un fallo (miss) del caché, y en este caso, los datos han de ser recuperados desde la memoria.

El empleo de memoria caché se ha popularizado, como medida para acelerar el tiempo de acceso a la memoria principal, incluso en los sistemas monoprocesadores, evitando así, según se incrementa la velocidad de los propios monoprocesadores, aumentar la velocidad de dicha memoria, y por tanto encarecer el sistema.

La forma en que la memoria es actualizada por los caches locales puede tener un gran impacto en las prestaciones de un sistema multiprocesador. Básicamente hay dos métodos:

1. Escritura continua (write-through)
2. Copia posterior (copy-back)

3.2.5- RISC frente a CISC

Existen varios mitos que contraponen las ventajas de la tecnología RISC frente a la CISC, que es importante descalificar:

- Los procesadores RISC ofrecen peor soporte para los lenguajes de alto nivel o HLL (High Level Language) que los CISC. Esta creencia se argumenta en que un conjunto de instrucciones de "alto nivel" (CISC) es mejor soporte para lenguajes de alto nivel. Sin embargo, la característica fundamental de los lenguajes de alto nivel, implica que el programador sólo interacciona con el ordenador a través del propio lenguaje de alto nivel (programación, depuración, mensajes del sistema, etc.), por lo que todos los problemas a "bajo nivel", deben de ser transparentes y desconocidos para él. Por ello, son de nulas consecuencias para el programador y los lenguajes de alto nivel, como se implementan las funciones, en función del tipo de CPU.
- Es más complicado escribir compiladores RISC que CISC. Dado que los procesadores CISC tienen un mayor número de instrucciones y modos de direccionamiento, existen por tanto más formas de hacer la misma tarea, lo que puede confundir tanto al compilador como al que escribe. Por ello, subjetivamente es posible escoger una forma de hacerlo poco adecuada, por el tipo de instrucciones o por el tiempo de ejecución que requieren. En cambio, en un procesador RISC, hay menos opciones, por lo que el compilador es más simple, aunque se genere,



habitualmente, un 20-30% más de código; a cambio, se consigue un incremento de la velocidad de hasta un 500%.

- Un programa es más rápido cuanto más pequeño. La velocidad a la que un programa puede ser ejecutado no depende en absoluto de su tamaño, sino del tiempo de ejecución de cada una de sus instrucciones. Dado que las instrucciones RISC son más rápidas, y admiten mejor pipelines, puede haber mayor paralelismo y simultaneidad en la ejecución de pequeñas secciones de código. Dichas secciones de código pueden ser ejecutadas en una fracción del tiempo que requiere una sola instrucción CISC.

3.2.6- RESUMIENDO

No es un hecho meramente académico, sino puramente comercial y económico. La "era RISC" ha alcanzado a todos los fabricantes de semiconductores: AMD, Intel, MIPS, Motorola, ROSS, ...; y todos ellos son productos usados por fabricantes de ordenadores y estaciones de trabajo: Apple, DEC, HP, IBM, SUN, etc. Y sus correspondientes clónicos.

El tiempo de diseño de estos productos se reduce sensiblemente, lo que disminuye su coste final, y por tanto, se incrementan sus expectativas, al poder llegar al mercado en un tiempo más adecuado, y con menos posibilidades de errores.

Además, son globalmente más eficaces, de menores dimensiones y más bajo consumo, ofreciendo siempre claras ventajas técnicas frente a los más avanzados CISC.

Actualmente, las estaciones de trabajo RISC multiprocesadoras de mayor éxito, se basan en diferentes versiones de la tecnología SPARC: superSPARC e HyperSPARC.

Está claro que el futuro pertenece a los RISC y a los sistemas multiprocesador, a no ser que la física y la electrónica logren superar las barreras tecnológicas para incrementar muy por encima de las cotas actuales, las velocidades y prestaciones de una única CPU.

También cabe esperar, y por qué no, la pronta aparición de otras tecnologías que compitan con CISC y RISC.



4. - DESCRIPCION DEL SISTEMA DE CONTROL DE LA RED DE CSE .

4.1 – INFORMACION GENERAL.

4.1.1 – Dominios del Sistema de Control.

CSE cuenta en la actualidad con tres sistemas independientes y con finalidades distintas para poder llevar a cabo las tres funciones de operación, mantenimiento y respaldo de seguridad. Cada sistema se identifica por un código de “dominio” que usaremos en adelante para su identificación.

De esta forma, tenemos el sistema o dominio “bor”, usado para la operación en tiempo real de la red de CSE. Sobre este sistema también se realizan las operaciones de modificación de la base de datos en tiempo real.

En paralelo con aquél y con la posibilidad de recibir datos de las remotas mediante la conmutación física manual de los modems de comunicaciones, está el dominio “ddg”. Su objetivo principal es el de ser alternativa al “bor” ante fallos prolongados de su redundancia (fallo doble) o para casos como el del proceso de actualización que se describe en este documento, para el que se requiere la parada completa de la aplicación en todos los servidores. Gracias a este respaldo, las funciones mínimas de supervisión y control de la red quedan garantizadas, aunque se trata de un sistema no redundante y por tanto susceptible de fallo simple. Por ello el proceso de actualización de una configuración simple del sistema “bor” debe llevarse a cabo en el menos tiempo posible, siendo este uno de los principales objetivos en la planificación del proceso y realizándose la misma operación previamente sobre el “ddg”.

Por último, tenemos el dominio “cse”, usado por el equipo de Sistemas de Control para las pruebas de integración de actualizaciones de código, compilaciones del sistema, configuración de la base de datos y prueba del nuevo código hasta donde lo permite el hecho de no disponer de toda la información en tiempo real del estado de la red.

4.1.2 – Identificación de los servidores del sistema.

Cada dominio esta compuesto por un conjunto de servidores con funciones claramente diferenciadas. Unicamente el dominio “bor” cumple el requisito de redundancia; el impacto de un fallo simple debe ser mínimo, siendo este mínimo dependiente del tipo de servidor que falle. Nos centraremos en él para una descripción breve de los servidores y sus funciones.



a) ADM: Administrador Principal (ADM1)

En él residen la base de datos maestra, la base de datos operacional estática (sin datos de tiempo real) y los datos históricos. Las operaciones de mantenimiento de la base de datos se realizan sobre él mismo.

A excepción del administrador secundario, el resto de servidores cuentan únicamente con un subconjunto de la base de datos operacional (más la información dinámica). Es indispensable que uno de los dos administradores estén en servicio para el funcionamiento de los puestos de operación. El proceso de conmutación entre administradores tienen una duración de 5 minutos durante el cual los puestos de operación quedan fuera de servicio. También es indispensable para el proceso de arranque de las aplicaciones en otros servidores (conocido como INITSOS).

b) ADM:Administrador Secundario (ADM2)

Cuenta con una copia actualizada de la base de datos operacional estática, datos históricos y la versión maestra de todas las aplicaciones. Sobre él se realizan las operaciones de mantenimiento del código para la posterior actualización de los servidores afectados. Puede asumir las funciones de administrador principal que afectan a la operación del sistema ante fallo de este último. El proceso de actualización que se describe se desarrolla fundamentalmente sobre el administrador secundario.

c) COM: Procesadores de Datos (COM1 y COM2)

Además de la porción necesaria de la base de datos operacional estática, es el maestro de datos dinámicos del estado de la red, encargándose del procesamiento de los mismos y de su distribución al resto de los servidores. Este procesamiento consiste en la verificación de límites, generación de alarmas, señalización, encaminados a la supervisión; gestión de los mandos y AGC; y archivo temporal de datos entre otras funciones.

El proceso de la conmutación es de 5 segundos a nivel de procesamiento, pero su efecto sobre los puestos de operación alcanza los 20 segundos. Funcionan indistintamente como principal o secundario.



d) NA: Análisis de Seguridad de la Red (NA01 y NA02)

En estos servidores funcionan las aplicaciones de análisis de la red, como el Estimador de Estados, Reparto de Carga, Análisis del N-1 en tiempo real, etc.

El fallo doble no afecta a las funciones de supervisión y control.

e) RTDS: Comunicación con las Remotas (RT01 y RT02)

Se comunica con las remotas para actualizar la información dinámica de los COM, realizando el procesamiento inicial de conversión a unidades de ingeniería de las medidas; gestiona los distintos protocolos de comunicaciones.

La carga de trabajo se reparte entre ambos por lo que ambos están siempre en activo, pudiendo asumir uno de ellos toda la carga ante fallo del otro. Es indispensable al menos uno de ellos para las funciones de supervisión y control de la Red.

f) UCS: Comunicación con otros sistemas (UC01 y UC02)

Gestiona los distintos protocolos de comunicaciones con otras empresas, para el envío de datos de CSE hacia el exterior así como actualización de la base de datos dinámica del COM en la porción de red para la que CSE no cuenta con remotas propias, así como red externa, necesaria fundamentalmente para las funciones de análisis seguridad.

g) MMI: Puestos de Operación

Desde estos servidores se realizan las funciones de operación. Cada centro cuenta con dos puestos a los que se suman los de mantenimiento. Reciben la información desde el COM y NA, enviando las órdenes de control al COM para su ejecución así como actividades de supervisión como aceptación de alarmas.

Existen dos tipos de servidores MMI, los locales, conectados a la misma red local (LAN) física que los descritos anteriormente, y los remotos, que trabajan sobre una red conectada con la anterior por medio de enlaces de telecomunicaciones y con unas características particulares en el diseño de sus aplicaciones tendentes a mejorar la velocidad de respuesta ante solicitudes del usuario; la velocidad del enlace es una restricción muy importante no solo para la respuesta del equipo, también para las tareas de mantenimiento del código, cuya actualización



UNIDAD CONCENTRADORA DE INFORMACIÓN

se realiza a través de la red; Evidentemente también afecta al tiempo necesario para el arranque de los mismos.

La siguiente tabla da una idea de lo anteriormente expuesto y además será usada posteriormente para justificar el orden de prioridad en la actualización de los puestos de operación. Refleja únicamente datos de la red A, siendo la B de 64 Kb normalmente:

PUERTO	CENTRO	IP	BAUDIOS
(5A)	Málaga	207	2 Mb
(5B)	Huelva	208	2 Mb
(5C)	Granada	209	2 Mb
(6A)	Badajoz	203	2 Mb
(6B)	Cádiz	205	2 Mb
(6C)	Córdoba	204	2 Mb
(7A)			
(7B)			
(7C)			
(8A)	Jaén	210	2 Mb
(8B)	Almería	211	2 Mb
(8C)	Calonge	202	2 Mb

PUERTO	CENTRO	IP	BAUDIOS
(3A)	Málaga	207	1920 K
(3B)	Huelva	208	64K
(3C)	Granada	209	64K
(4A)	Badajoz	203	64K
(4B)	Cádiz	205	64K
(4C)	Córdoba	204	64K
(7A)			
(7B)			
(7C)			
(8A)	Jaen	210	64k
(8B)	Almeria	211	64k
(8C)	Calonge	202	1920k



4.2 – ARQUITECTURA INFORMÁTICA EN EL DDM.

La Arquitectura informática aplicada fue diseñada según las especificaciones desarrolladas por los expertos en sistemas de control de CSE, basada en la Tecnología de Sistemas Abiertos. Todo el desarrollo de aplicaciones fue adjudicado en concurso a la firma americana EMPROS, la cual fue posteriormente comprada por la multinacional SIEMENS.

En la gráfica de detalle de máquinas, del Sistema SIEMENS-EMPROS, veremos que las mismas están todas interconectadas mediante una red ethernet dual, como habíamos dicho en apartado anterior. Para que la visualización sea más correcta, en detalle, se mostrará por zonas limitadas por su ubicación:

4.2.1 – Despacho de Operación.

