

# ARQUITECTURAS HW Y SW PARA SISTEMAS DE TIEMPO REAL

Fernando Sevillano  
[fernando.sevillano@wonderware.es](mailto:fernando.sevillano@wonderware.es)



Marta Beltrán  
[marta.beltran@urjc.es](mailto:marta.beltran@urjc.es)



# AGENDA

2

## ANTES DEL “COFFEE BREAK”

- “Arquitecturas HW y SW para sistemas de Tiempo Real”. Marta Beltrán (Universidad Rey Juan Carlos) y Fernando Sevillano (Wonderware Spain).

## DESPUES DEL “COFFEE BREAK”

- “EMASAGRA: Un caso práctico de implantación de Tecnología Wonderware”. Ramón Carlos Valor (EMASAGRA).

# CONTENIDOS

3

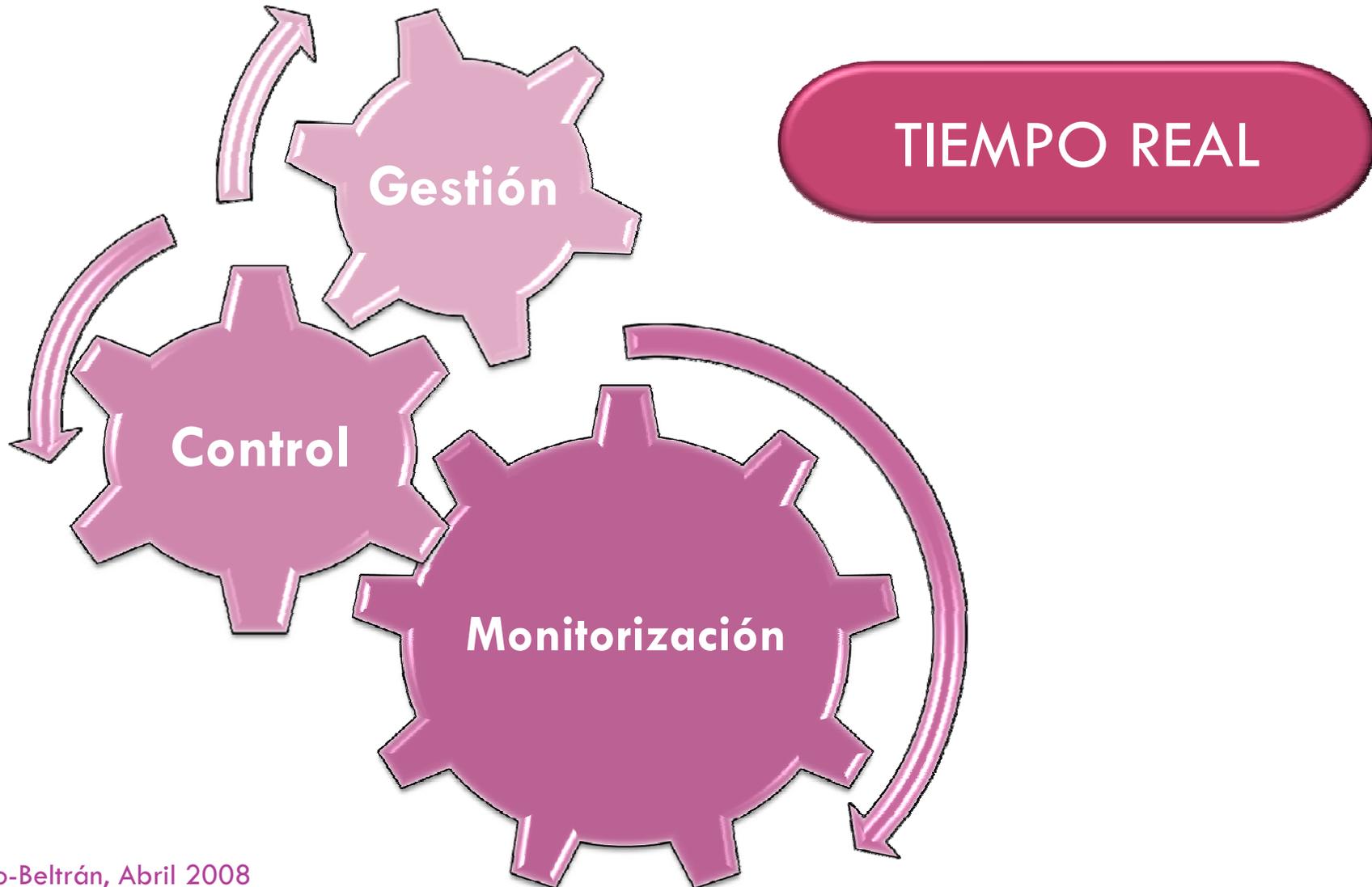
- 1) Introducción.
- 2) Conceptos Básicos de Tiempo Real.
- 3) Arquitecturas HW para Tiempo Real.
- 4) Arquitecturas SW para Tiempo Real.
- 5) El problema de la integración de aplicaciones.
- 6) Solución tecnológica propuesta por Wonderware.
- 7) Conclusiones.

# 1. INTRODUCCIÓN

- El concepto de Tiempo Real surge constantemente en el ciclo integral del agua.
  - ▣ Seguimiento de la evolución de caudales y del nivel de los depósitos, regulación de la apertura de compuertas.
  - ▣ Gestión del suministro, detección de averías e incidencias en la red.
  - ▣ Monitorización de la calidad del agua, detección de contaminantes.
  - ▣ Facturación.

# 1. INTRODUCCIÓN

5



## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

6

- “Proceso en el que el usuario no percibe el tiempo empleado en su ejecución”.
- “Tipo de procesamiento en cual una transacción es ejecutada y procesada sin espera alguna”.

(definición para usuarios “comunes”)

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

7

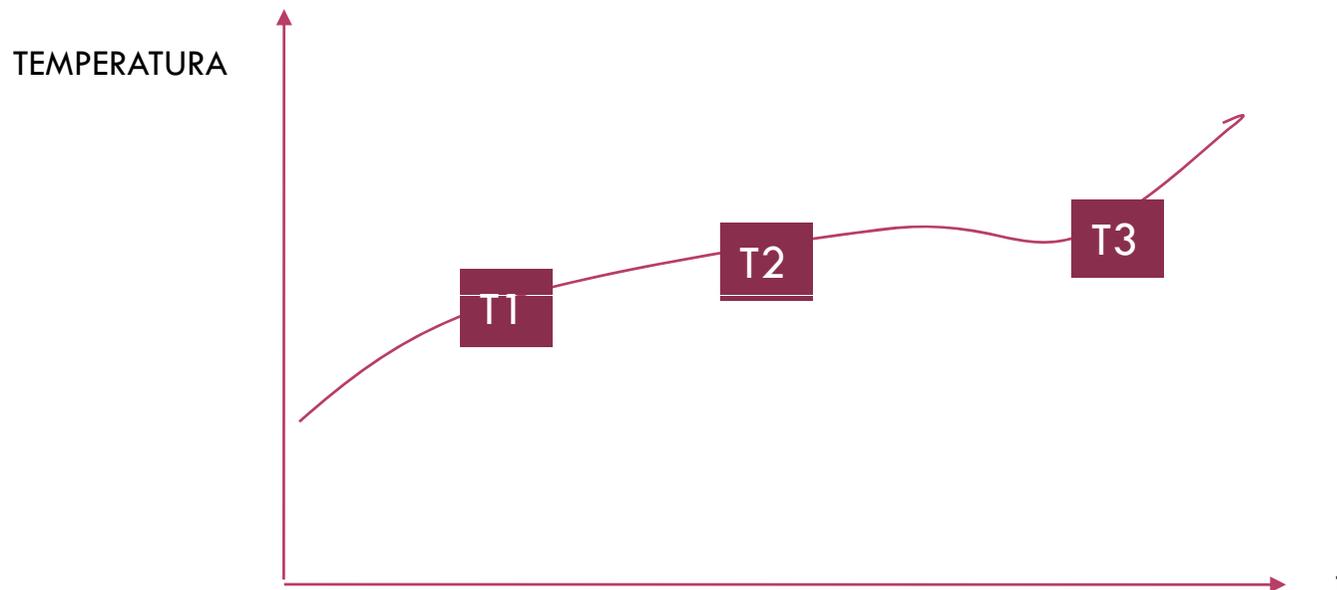
- “Rápida transmisión y proceso de datos orientados a eventos y transacciones a medida que se producen, en contraposición a almacenarse y retransmitirse o procesarse por lotes”.

(definición para servidores, entornos de comunicación, etc)

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

8

- “Un sistema de tiempo real es aquel capaz de procesar una muestra de señal antes de que ingrese al sistema la siguiente muestra”.



(definición para procesamiento digital de señales)

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

9

- “Un sistema de tiempo real es aquel en el que la **corrección de los resultados** no depende sólo de la **corrección de los cálculos** realizados para producirlos, sino también del **instante** en el que éstos están disponibles”.

(definición más aceptada, Donald Gillies”)

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

### □ **En resumen:**

- **Tiempo real no es sinónimo de inmediatez, baja latencia o rapidez de funcionamiento:**
  - No es la velocidad de la respuesta del sistema la que lo convierte en un sistema de tiempo real.
  - El objetivo en sistemas de tiempo real es asegurarse de que la latencia es la adecuada para resolver el problema al cual el sistema está dedicado.
    - Y esto pueden ser días, horas, segundos o microsegundos dependiendo del sistema.
      - No es lo mismo monitorizar el nivel de un depósito, que controlar la apertura de una compuerta, detectar una avería en la red o generar una factura.

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

- **Tipos de sistema de tiempo real:**
  - ▣ **Sistemas Hard Real Time:** Cuando la latencia de un proceso del sistema sobrepasa su cota máxima se produce un error en el sistema.
  - ▣ **Sistemas Soft Real Time:** Cuando la latencia de un proceso del sistema sobrepasa su cota máxima dentro de unos límites establecidos sin provocar un error en el sistema.
  - ▣ **Sistemas de Misión Crítica:** Cuando la latencia de un proceso del sistema sobrepasa su cota máxima puede llevar a la pérdida de vidas o a catástrofes similares.

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

12

### □ Características de un sistema de tiempo real:

Determinismo

Fiabilidad y  
seguridad

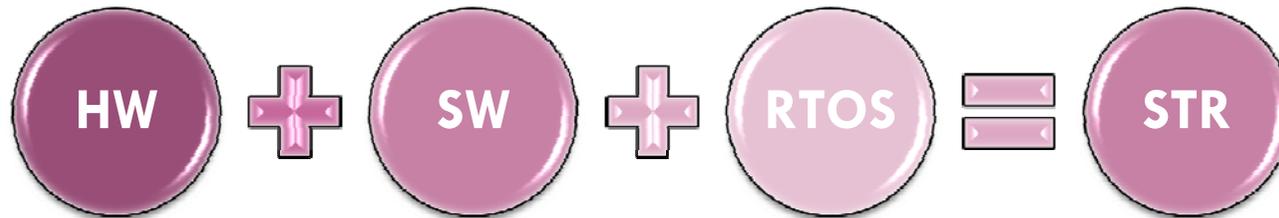
Tolerancia a  
fallos

Gran  
complejidad

Simultaneidad  
en la ejecución  
de tareas

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

13



### □ HW que no introduzca aleatoriedad.

- Procesador, memoria, disco duro: el sistema debe ser lo más predecible posible (sin caché, sin paginación, sin predicción de saltos, etc).
- Todos estos factores añaden una aleatoriedad que hace que sea difícil demostrar que el sistema es viable, es decir, que cumple con los plazos.

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

14

- SW específico.
  - ▣ Optimizado para cumplir con los plazos y para no introducir aleatoriedad.
  - ▣ Mínima utilización de recursos
- RTOS: Real Time Operating System.
  - ▣ Planificador con asignación de prioridades a los procesos.
  - ▣ Especial importancia del tratamiento de interrupciones.

	Latencia/Jitter
SO estándar	0.1 ms-100 ms
Linux	1 ms
IEEE 1003.1 d Linux	10-100 $\mu$ s
Real Time Linux	1-10 $\mu$ s
RTOS kernels	1-10 $\mu$ s

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

15

El STR debe garantizar que todas las tareas que se ejecutan, finalizan dentro del plazo especificado (con el margen correspondiente si es que lo hay).



Independientemente de si el plazo es de días, horas o microsegundos.

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

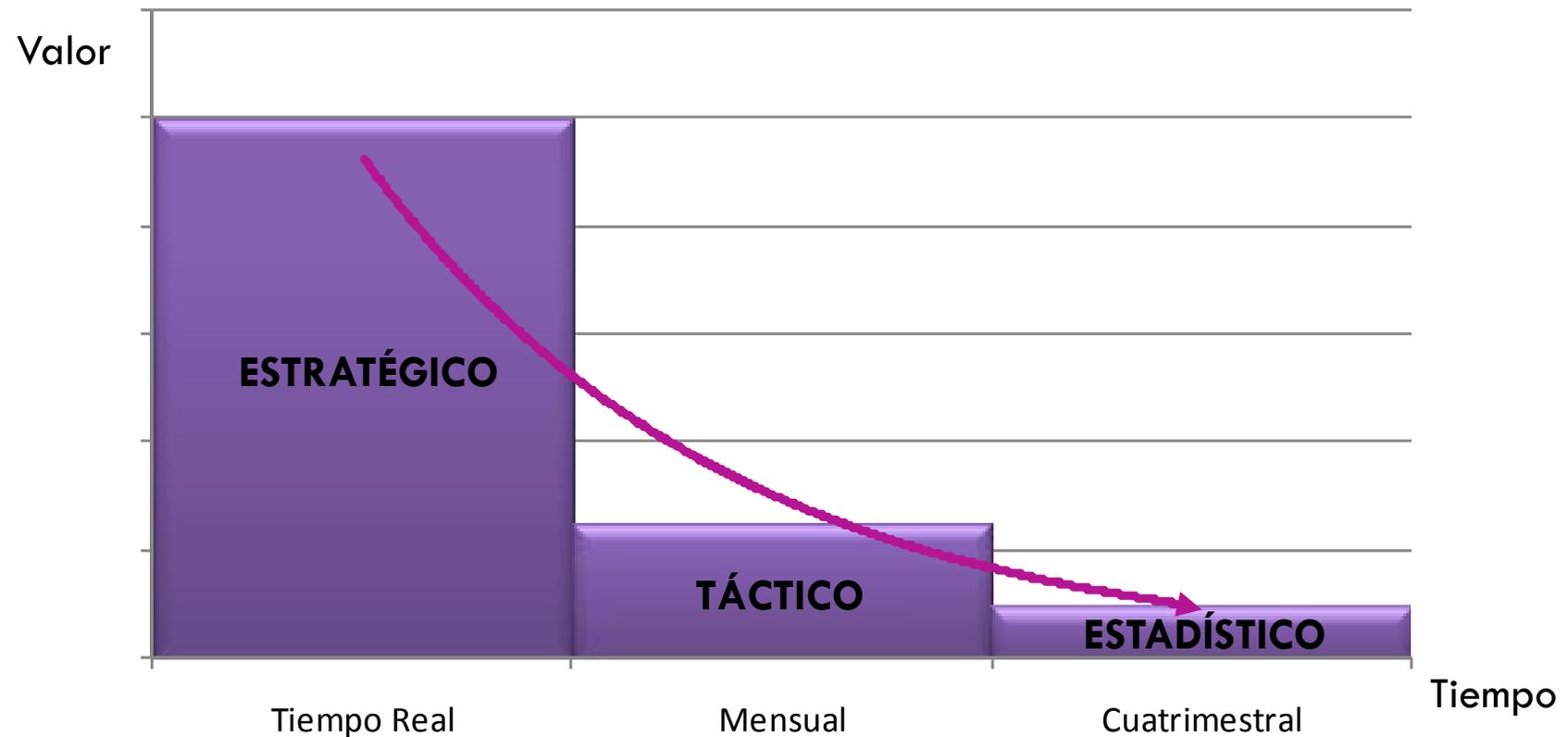
16

- **Tiempo Real aplicado a la gestión global de la empresa:**
  - **Zero Latency Entreprise (ZLE)**
    - “Que la información llegue en un tiempo mínimo a la persona responsable”.
  - ZLE implica que la empresa ha minimizado la latencia de sus operaciones de manera que los eventos relacionados con cualquier factor que afecte a la compañía inmediatamente desencadenan las acciones oportunas de las personas responsables.

## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

17

### VALOR ESTRATÉGICO DE LA INFORMACIÓN



## 2. CONCEPTOS BÁSICOS DE TIEMPO REAL

18

### □ **Beneficios de ZLE:**

- Brindar una vista en tiempo real del negocio.
  - Información necesaria para reaccionar ante problemas cuanto antes.
- Disponer de aplicaciones de gestión de negocio en tiempo real (Data Mining, CRM, Score-Card, Business Intelligence)
- Detectar y eliminar potenciales peligros para el sistema (amenazas de seguridad externas e internas).
- Incrementar la productividad y la eficiencia de los procesos de negocio.

### 3. ARQUITECTURAS HW PARA TIEMPO REAL

19

## ALTERNATIVAS ACTUALES

**HW COMERCIAL**

**DESARROLLOS A  
MEDIDA**

**ARQUITECTURAS  
PARALELAS**

Procesadores  
de propósito  
general

HW  
específico

ASIC

FPGA

Memoria  
compartida

Memoria  
distribuida

### 3. ARQUITECTURAS HW PARA TIEMPO REAL

20

#### □ Procesadores de propósito general



# 3. ARQUITECTURAS HW PARA TIEMPO REAL

21

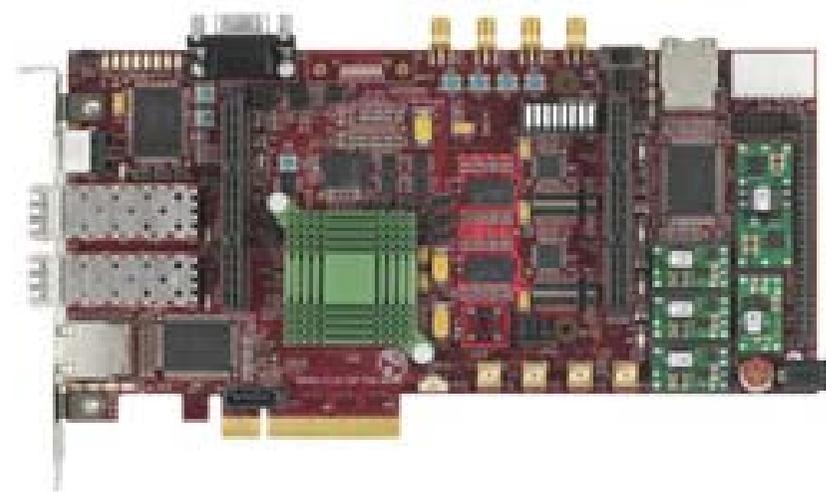
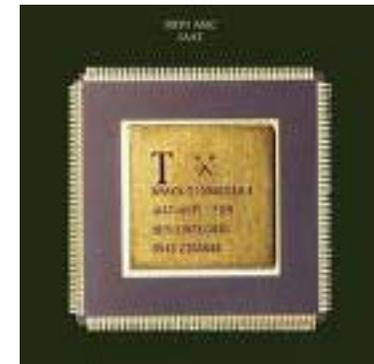
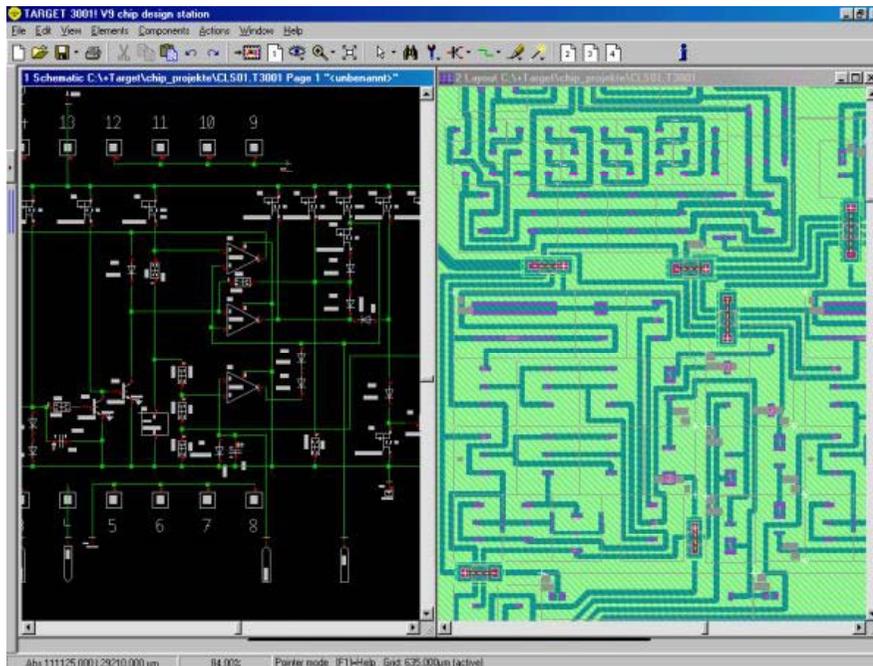
## □ HW específico



# 3. ARQUITECTURAS HW PARA TIEMPO REAL

22

## □ Desarrollos a medida (ASIC y FPGA)



### 3. ARQUITECTURAS HW PARA TIEMPO REAL

23

- **Arquitecturas paralelas (memoria compartida y distribuida)**



## 4. ARQUITECTURAS SW PARA TIEMPO REAL

- El SW programado para sistemas de tiempo real está casi siempre descompuesto en tareas que se ejecutan concurrentemente.
- Al igual que ocurre con el HW de un sistema de tiempo real, la ejecución de estas tareas debe ser determinista.
  - ▣ Por eso deben evitarse ciertos lenguajes de programación, herramientas y prácticas en el desarrollo de aplicaciones de tiempo real.
- Actualmente existen multitud de herramientas y tecnologías que son adecuadas para programar SW en tiempo real.
  - ▣ POSIX, RTJava o ADA son sólo algunos ejemplos.

# 4. ARQUITECTURAS SW PARA TIEMPO REAL

25

## □ Decisiones importantes:

### □ Arquitectura del SW.

- Módulos que componen esa arquitectura (descomposición en tareas).
- Comunicación entre módulos.

### □ Diseño de los módulos.

- Funcional.
- Implementación.

### □ Mecanismos de tolerancia a fallos.

## 4. ARQUITECTURAS SW PARA TIEMPO REAL

- **Mejores prácticas en el diseño de SW para tiempo real:**
  - Aprovechar la experiencia de otros diseñadores pero sin reutilizar su código a no ser que estemos seguros de que el problema es exactamente el mismo.
  - Estudiar con detalle el HW y el SO del sistema.
  - Basar la selección de tecnología y herramientas en un análisis exhaustivo de las necesidades técnicas del sistema.
    - Utilizar estándares siempre que sea posible.
  - Invertir muchos esfuerzos en los programas de test (siempre worst-case).
  - Utilizar simulación siempre que sea posible ya que la verificación formal suele ser complicada.

# 5. EL PROBLEMA DE LA INTEGRACIÓN DE APLICACIONES

- En la mayor parte de sistemas actuales conviven aplicaciones de diferentes fabricantes, programadas con diferentes lenguajes, que se ejecutan sobre diferentes plataformas y sistemas operativos, manejando diferentes formatos de datos y que funcionan sobre diferentes motores de base de datos.
- Por lo tanto es necesario realizar grandes esfuerzos para integrar todas estas aplicaciones.
  - EAI (Enterprise Application Integration).
- Especialmente cuando existen requisitos de tiempo real.

# 6. SOLUCIÓN TECNOLÓGICA PROPUESTA POR WONDERWARE

28

## □ Una Plataforma Tecnológica.

- Conectable con sistemas que incorporan Software para Tiempo Real.
  - PLC, dispositivos de campo, RTU.
- Desplegable en Arquitecturas Hardware flexibles y escalables.
  - Del monopuesto a arquitecturas distribuidas.
  - Aprovechando al máximo las capacidades del Hardware instalado. Redundancia, Balanceo de carga.
- Que facilita la Integración Horizontal (a nivel campo) y la la Integración Vertical (ZLE).

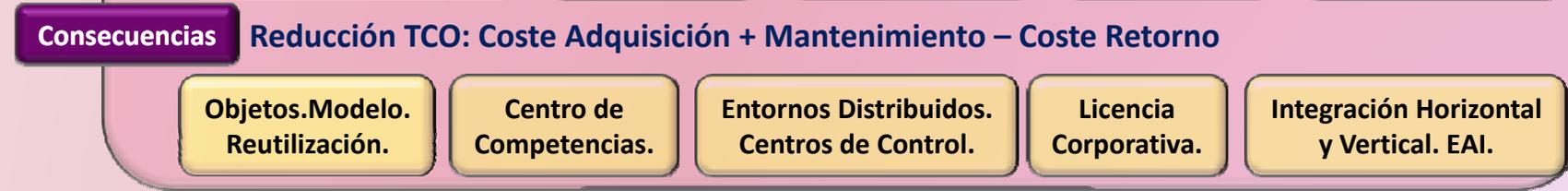


**Plataforma Sistemas Transaccionales: ERP**

Mensajería XML. MIMOSA

**Control Costes / Desviaciones / Mejora**  
*Tiempo Real. ZLE.*

Información Instalación

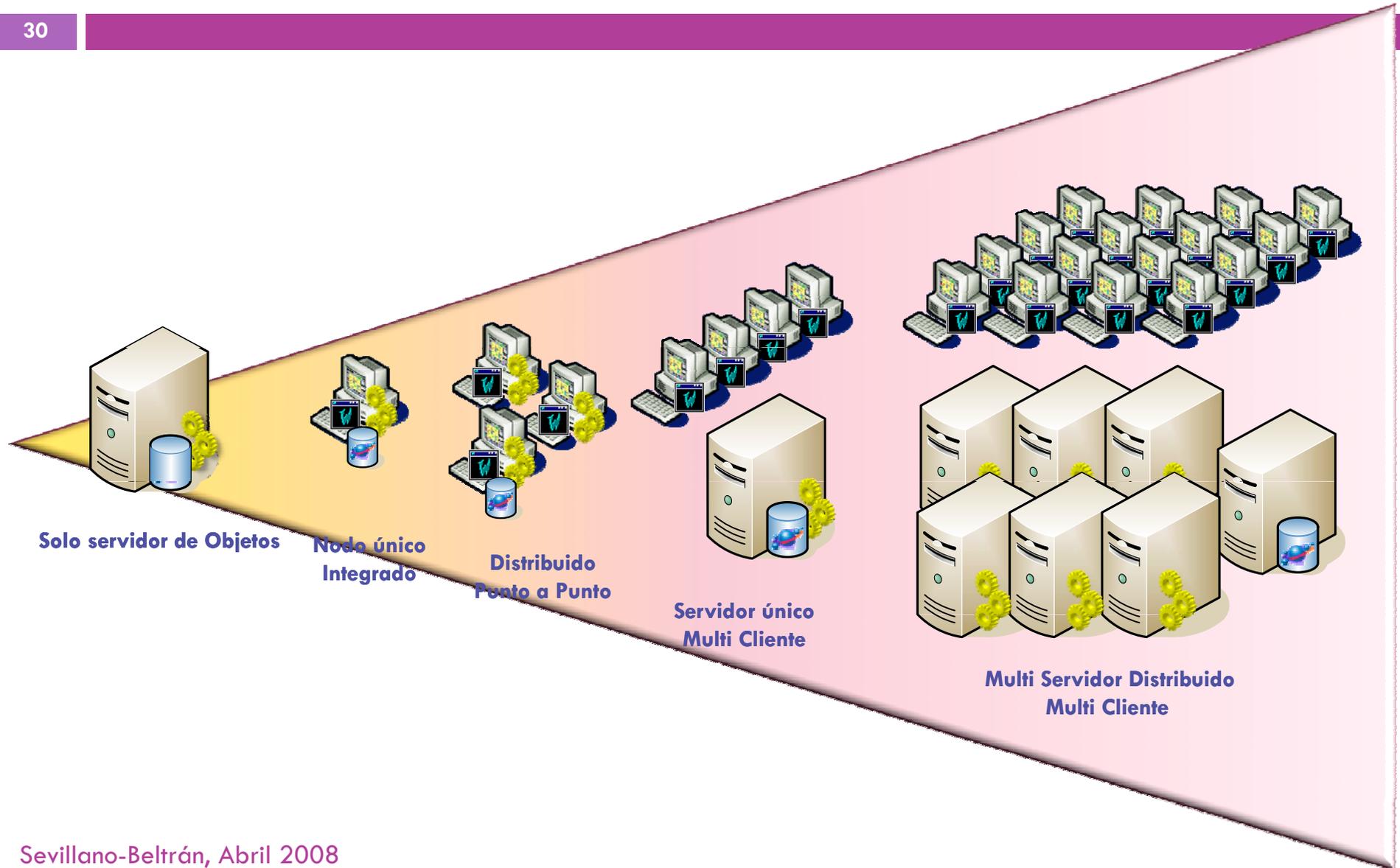


**Plataforma Sistemas Tiempo Real**

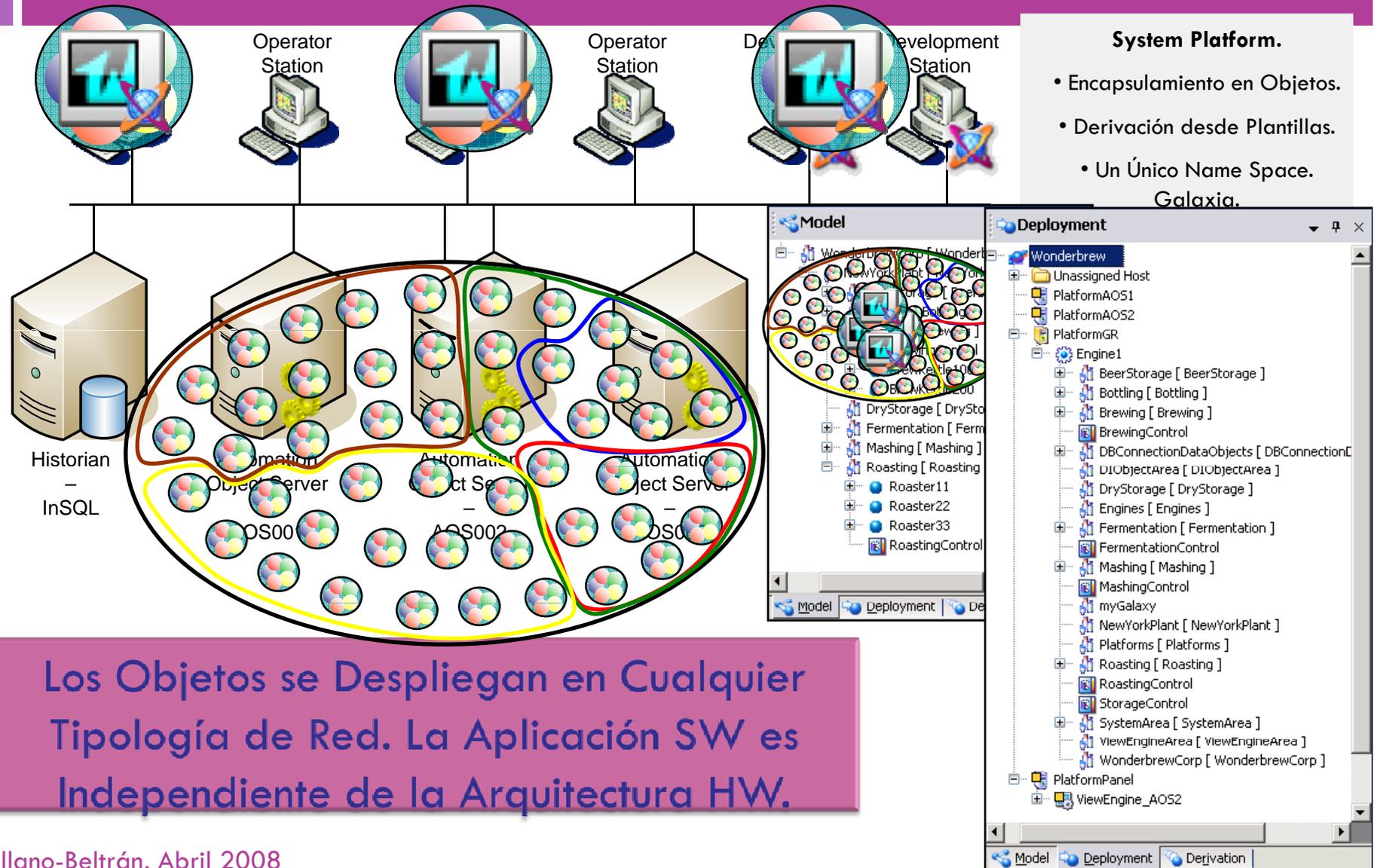


# 6. SOLUCIÓN TECNOLÓGICA PROPUESTA POR WONDERWARE

30

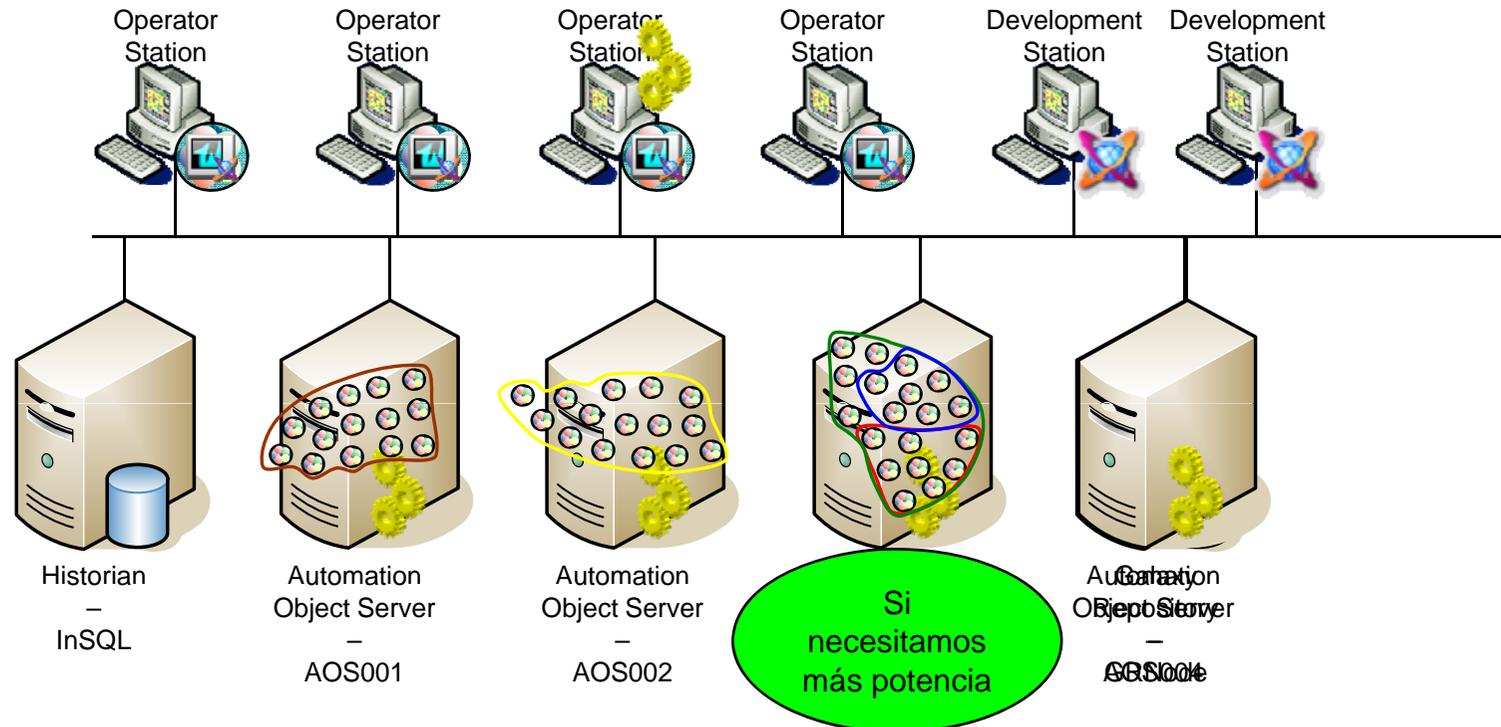


# 6. SOLUCIÓN TECNOLÓGICA PROPUESTA POR WONDERWARE



# 6. SOLUCIÓN TECNOLÓGICA PROPUESTA POR WONDERWARE

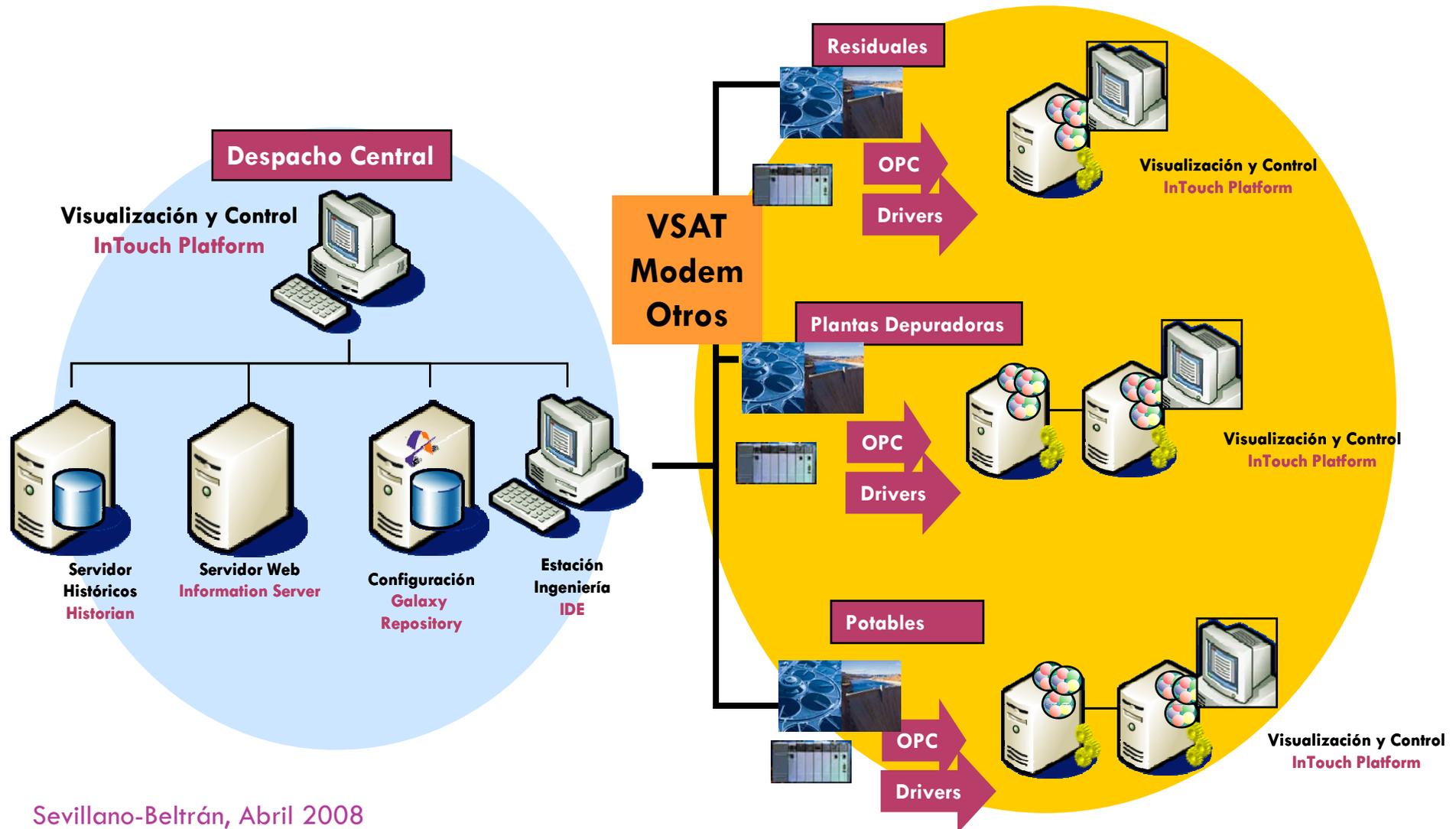
32



- Redistribución de la Carga en Tiempo de Ejecución. Alta Disponibilidad.
- Ampliación de Servidores para dotar de más Potencia al Sistema. Ampliación. Escalabilidad
- Ampliación de Puestos Clientes sin necesidad de “tocar” la Aplicación Principal. Reducción TCO.
- Los Clientes leen los objetos distribuidos entre Servidores. Sin definir dirección. Flexibilidad.

# 6. SOLUCIÓN TECNOLÓGICA PROPUESTA POR WONDERWARE

33



# 7. CONCLUSIONES

- Un sistema de tiempo real debe garantizar que todas las tareas que se ejecutan en él finalizan dentro del plazo especificado, sea este el que sea.
  - ▣ Gestión, control y monitorización a bajo y alto nivel.
- Este tipo de sistemas suele ser tremendamente complejo, y deben ser deterministas, fiables, seguros, tolerantes a fallos y capaces de ejecutar varias tareas simultáneamente.

# 7. CONCLUSIONES

- Existen multitud de arquitecturas HW y SW que permiten implementar sistemas de tiempo real hoy en día.
  - Problemas de integración.
  - Muy importante también el SO.
- Lo importante es conocer todas las alternativas, y escoger las que menos aleatoriedad introduzcan en el sistema.
- Wonderware propone una plataforma tecnológica que permite integrar sistemas e información en multitud de entornos diferentes con necesidades de tiempo real.

# 10 RAZONES ESTRATÉGICAS.

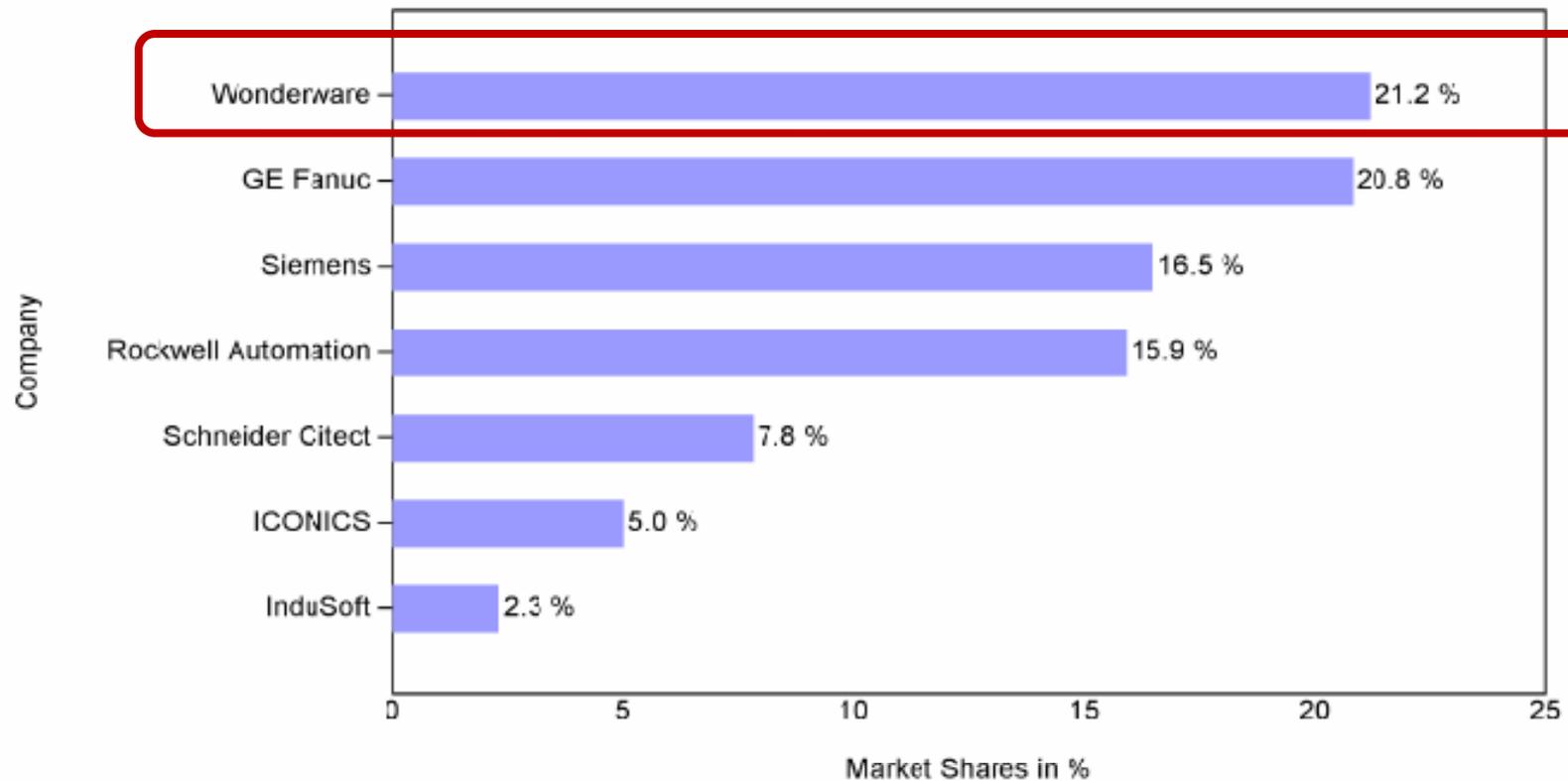
36



# 1. Solución HMI (Human Machine Interface) Líder del Mercado.

37

2005 = \$ 651.0 Million



Other = 10.5 %

# 2. Proveedor de Software Estándar. Independiente fabricantes Hardware.

38

Testimonio de la capacidad  
del mejor software de automatización  
e información industrial del mundo

100.000+

Cada día, el 30% de las plantas del mundo confían en el software de Wonderware para tomar decisiones de planta mejores y de forma más rápida.

[wonderware.com/ad/100000+](http://wonderware.com/ad/100000+)

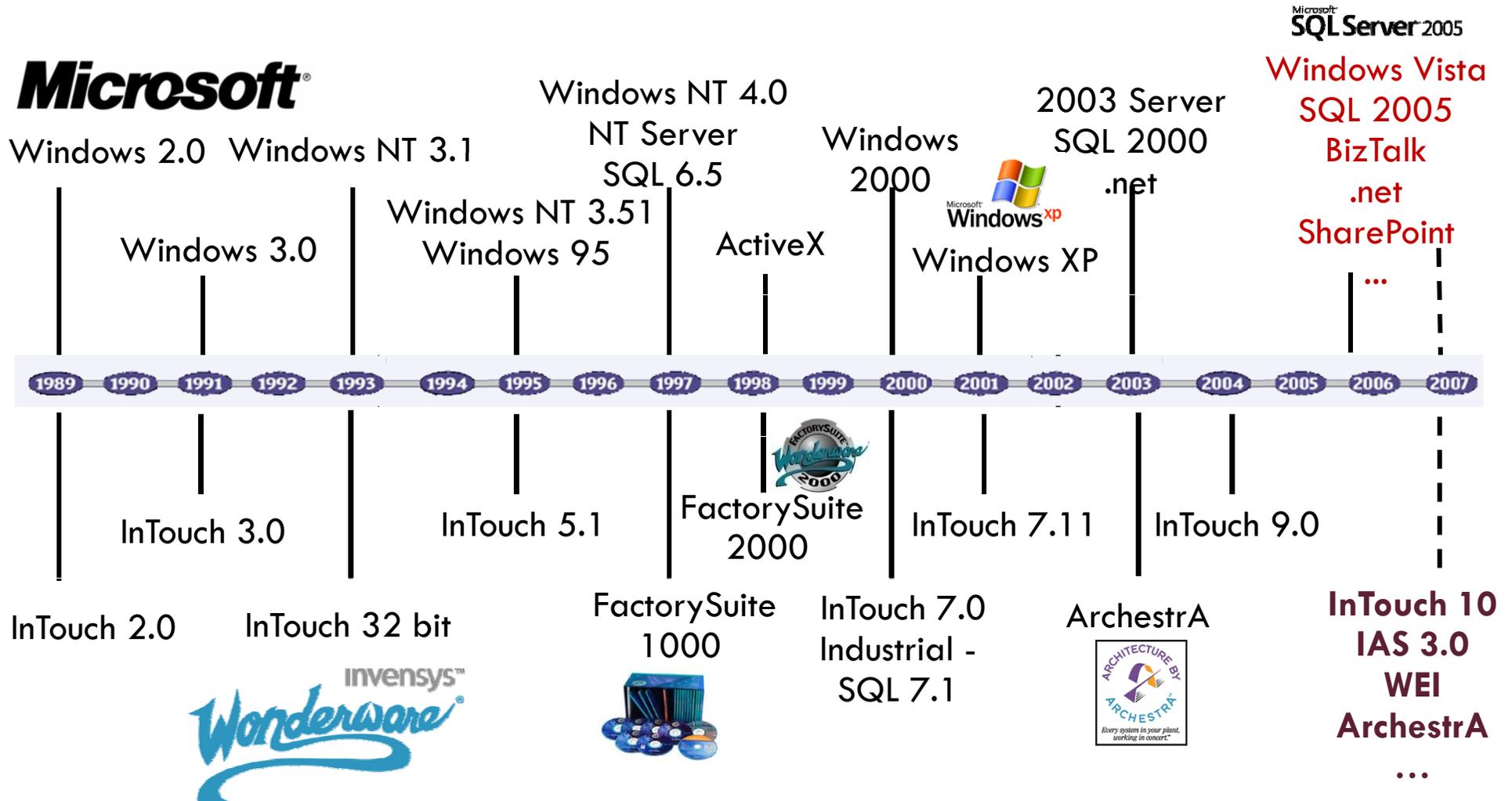
No puede ser que más de 100.000 plantas estén equivocadas. Cuando la necesidad es tener la información adecuada, en las manos apropiadas, en el tiempo oportuno para realizar las acciones correctas, más y más plantas en todo el mundo eligen el software de Wonderware. ¿Por qué? Por su facilidad de utilización sin igual, por su sencilla integración con el software y el hardware existente y por la inmejorable visualización de la información en tiempo real. Construidas sobre la innovadora arquitectura software de información y automatización industrial Archestra, las soluciones de software de Wonderware proporcionan significativos

aumentos de la productividad, rendimiento y calidad, todo por una pequeña inversión, entregando uno de los mejores ROI de la industria. Continuamente ponemos límites al *state-of-the-art* en tecnología, aún trabajando con su software de gestión y de planta existente, por lo que puede impulsar su competitividad. Y este es un testimonio que puede llevar hasta su banco.

**Wonderware**

Powering intelligent plant decisions in real time.

# 3. Inexistencia de Saltos Tecnológicos.



# 4. Arquitectura Software basada en .NET de Microsoft. Archestra.

40



**Capa de Informes**



**Capa de Ejecución**



**Capa de Desarrollo y  
Mantenimiento**

Microsoft  
**.net**



# 5. Modelo de Negocio. Certificación de Integradores.

41



CERTIFIED  
SYSTEM INTEGRATOR.



SYSTEM INTEGRATOR.



# 6. Comunicación Universal.

42

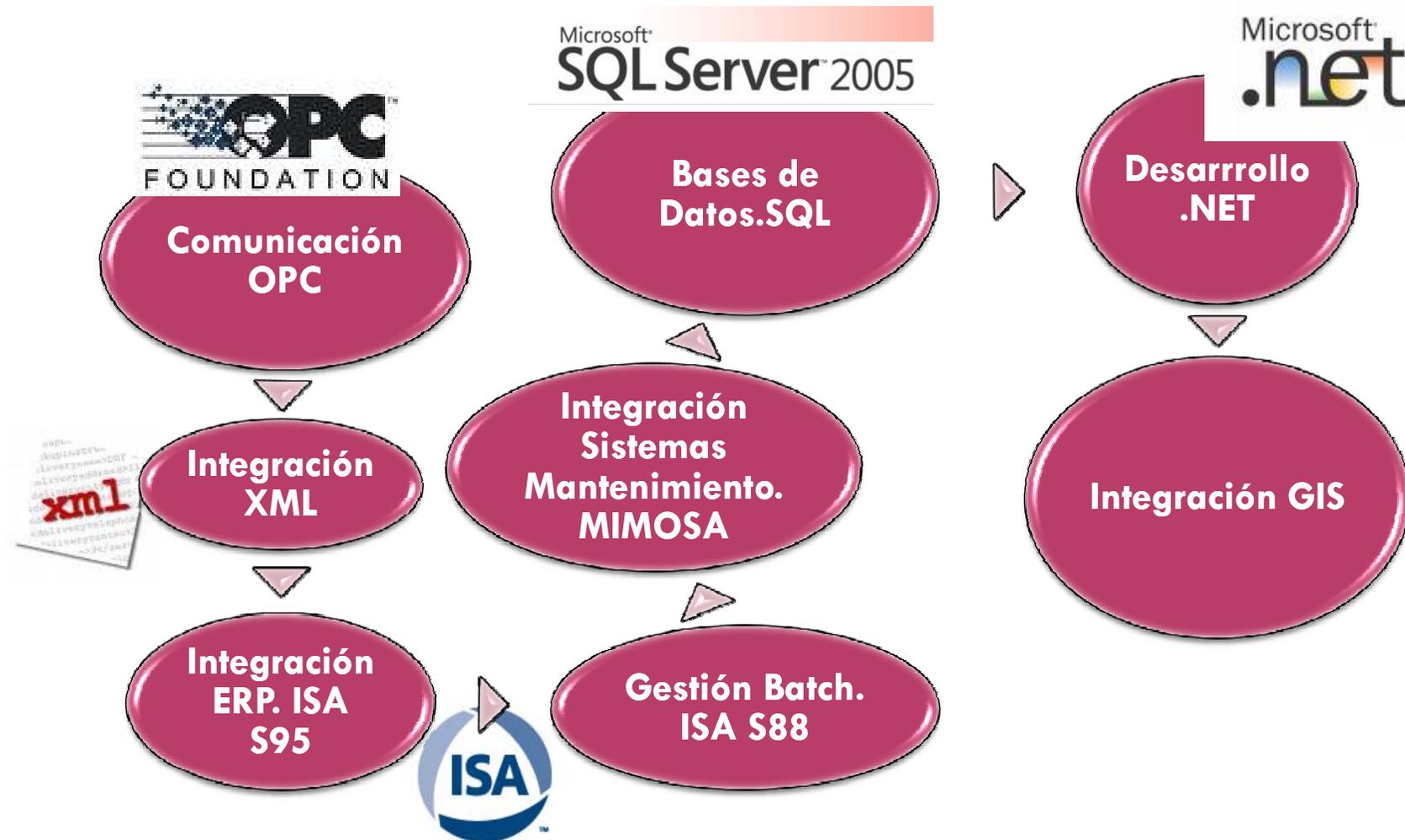
## INTEGRACIÓN VERTICAL

### INTEGRACIÓN HORIZONTAL

**300 Drivers para comunicar  
con Dispositivos de Planta  
Utilidades para Integrar otros  
SCADA**

**Software  
Estándar  
para  
abordar  
proyectos de  
Integración  
de Sistemas  
GIS y ERP**

# 7. Soluciones basadas en Estándares o Estándares de Facto.



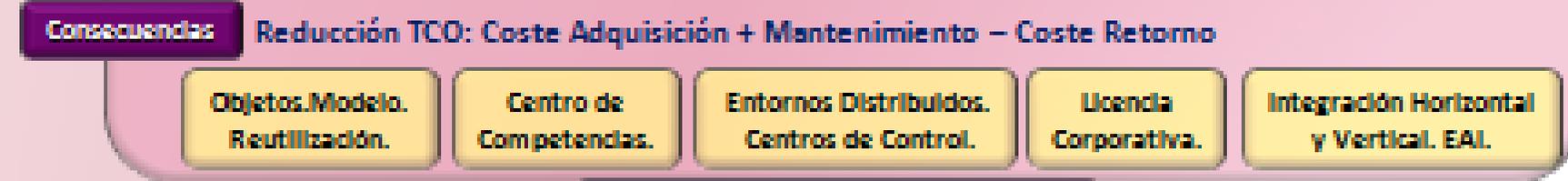
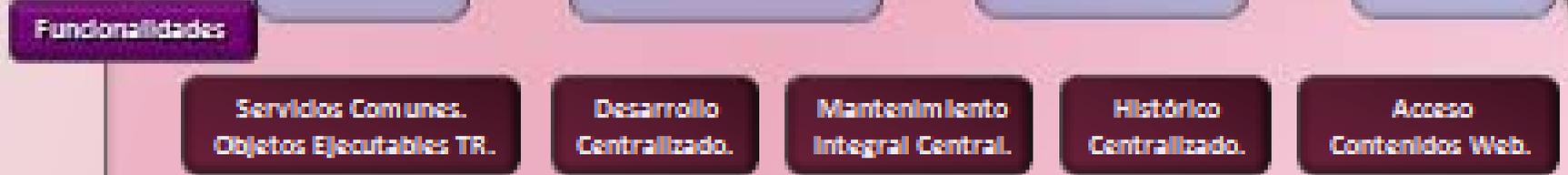


**Plataforma Sistemas Transaccionales: ERP**

Mensajería XML MIMOSA

**Control Costes / Desviaciones / Mejora  
Tiempo Real. ZLE.**

Información Instalación



**Plataforma Sistemas Tiempo Real**



# 9. Funcionalidades incluidas en la Plataforma.

45

## Redundancia

Comunicaciones  
Aplicaciones  
Históricos  
Visualización

## Gestión Aplicación Única

Gestión Entornos  
Distribuidos.  
Integración  
Aplicaciones

## Escalabilidad

Señales e Históricos  
Servidores y Clientes  
Incremento Funcionalidades

Reducción TCO.  
(Total Cost of  
Ownership)

# 10. Wonderware apuesta por el Sector Aguas.

46



❑ En Mayo de 2008, Wonderware lanza al Mercado, su Primer Vertical dirigido al SECTOR AGUAS.

- ❑ Objetos Gráficos y Lógicos Predefinidos.
- ❑ Más drivers de comunicación.
- ❑ Problemática de envío de datos asíncrono. (RTU).

# Un pensamiento...

47

The logo for Wonderware Spain. The word "Wonderware" is written in a large, teal, cursive script font. A thick, teal, curved line starts under the "w" and sweeps under the "e" in "Wonderware". Below this line, the word "Spain" is written in a bold, dark grey, sans-serif font.

**Wonderware**<sup>®</sup>  
**Spain**

# MUCHAS GRACIAS

Fernando Sevillano  
[fernando.sevillano@wonderware.es](mailto:fernando.sevillano@wonderware.es)



Marta Beltrán  
[marta.beltran@urjc.es](mailto:marta.beltran@urjc.es)

