



Software Group

# WebSphere z/OS Strategy

Franck Bonsergent  
Mgr zSeries Technical Sales team  
IBM Certified

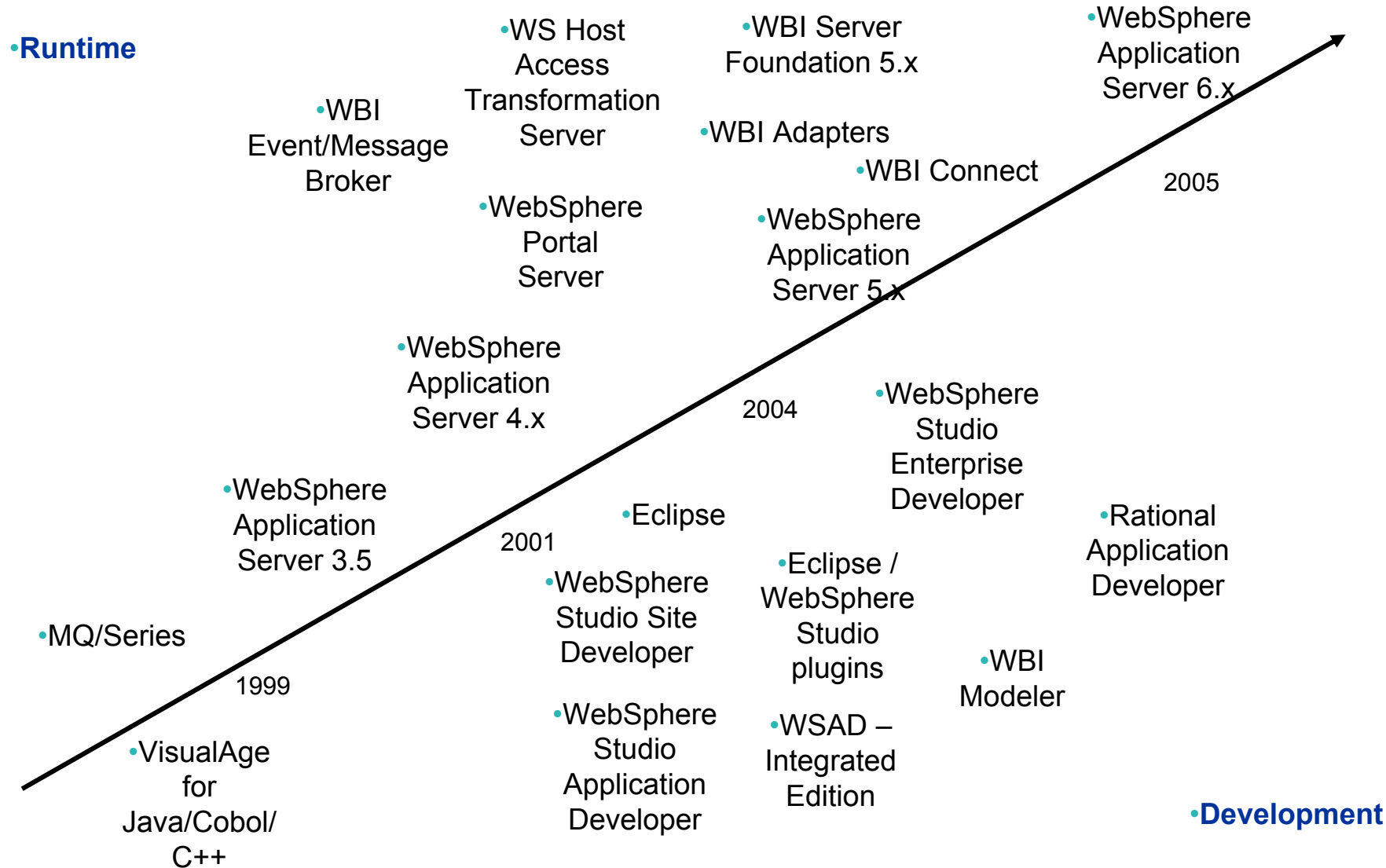
# Agenda

- WebSphere – Definition
- How the WebSphere offering evolved over time?
- Business needs
- Next Step: SOA
- Why and When WebSphere on zSeries?
- How we can help you with Business Integration Architecture Workshops

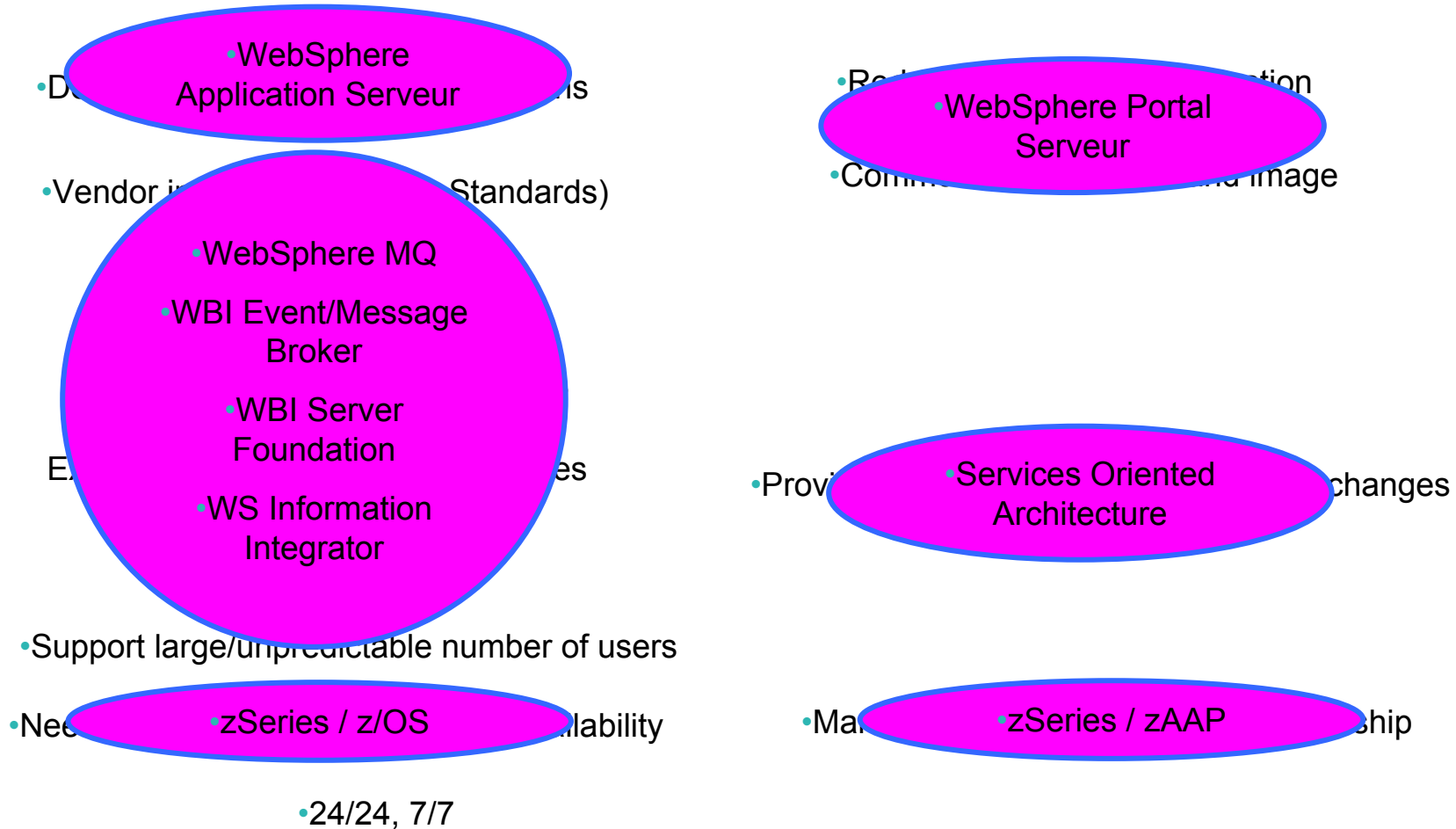
## WebSphere - Definition

- **WebSphere is:**
  - Middleware software solutions
  - A product family – A brand
  - Fully based on open standards
    - J2EE, XML, Web Services
  - Based on 2 major products:
    - WebSphere Application Server
    - WebSphere MQ
  - Available on all major platforms
    - Windows
    - Unix (AIX, Solaris, HP)
    - Linux Intel, Linux for z/Series
    - OS/400
    - z/OS
  - Same product features on all platforms
  - Same development tools for all products
- **WebSphere is not:**
  - Application solutions (ERP)
    - Except some Industry solutions
  - A unique product
  - A proprietary technology

# How the WebSphere offering evolved over time?



# Business Needs

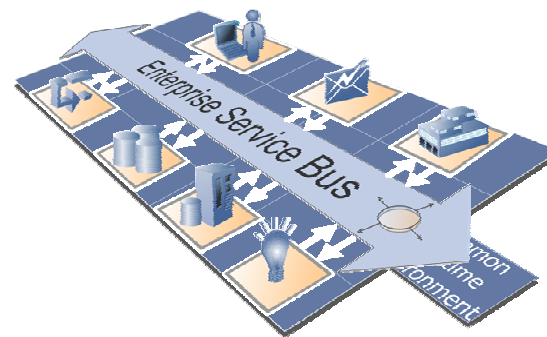


# Next step: SOA

# What is SOA?

## Service Oriented Architecture (SOA)

- A Service Oriented Architecture enables flexible connectivity of applications or resources by
  - ▶ Representing every application or resource as a service with a standardized interface enabling them to exchange Structured information (messages, documents, 'business objects')
  - ▶ mediating the message exchange through a service integration bus
  - ▶ providing on-ramps to the bus for legacy application environments



# What is SOA?

## Why Service Oriented Architecture?

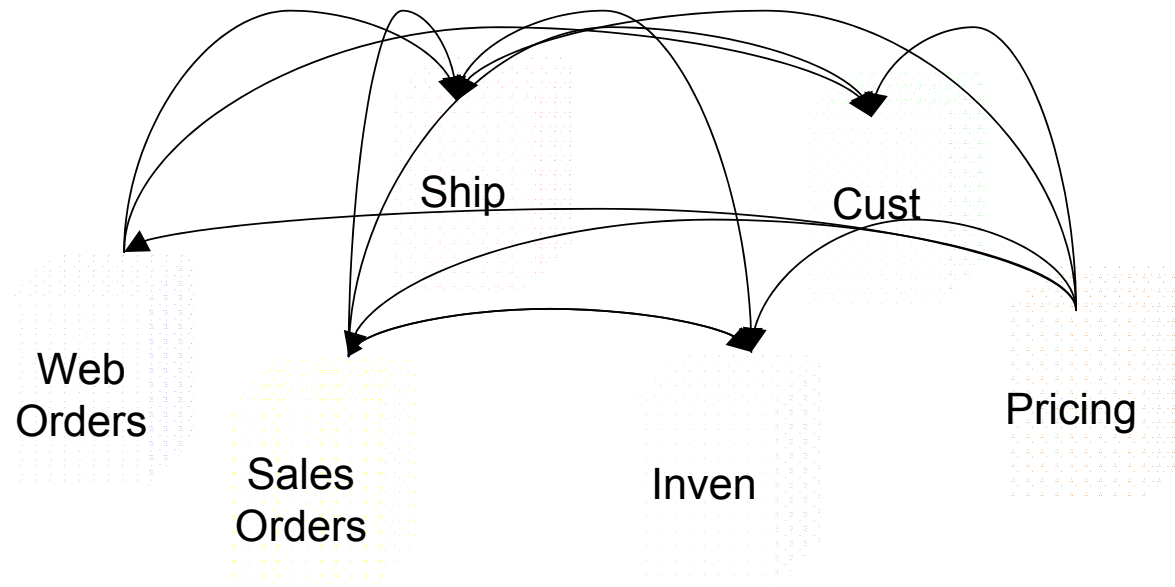
- Enables flexible *connectivity* of applications or resources.
- Promotes encapsulation which enables *re-use*: each aspect of your business is captured and implemented in only one place, so changing it is straightforward.
- Requires the use of *implementation independent interfaces* which means that heterogeneous systems can be integrated, and changes to one lead to fewer negative effects.
- Enabler for *process modeling* and automation. With clearly defined interfaces between all business systems, you can start modeling (and changing) the business process captured by them at a level above individual systems.





# What is SOA

## Component-based Architecture is not Enough

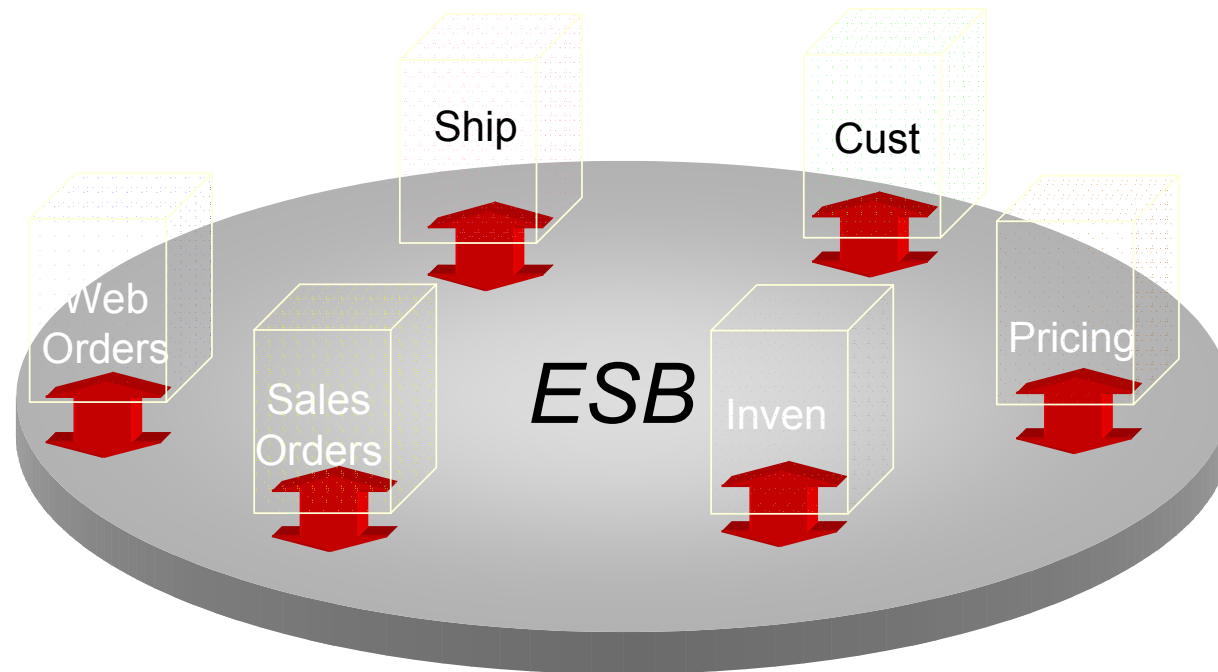


Services defined as units of business logic, but ...

- Flow of control – bound into service logic
- Transformation of data formats bound into service logic
- Tight coupling between services makes them fragile

# What is SOA

## Move IT Logic out of Services

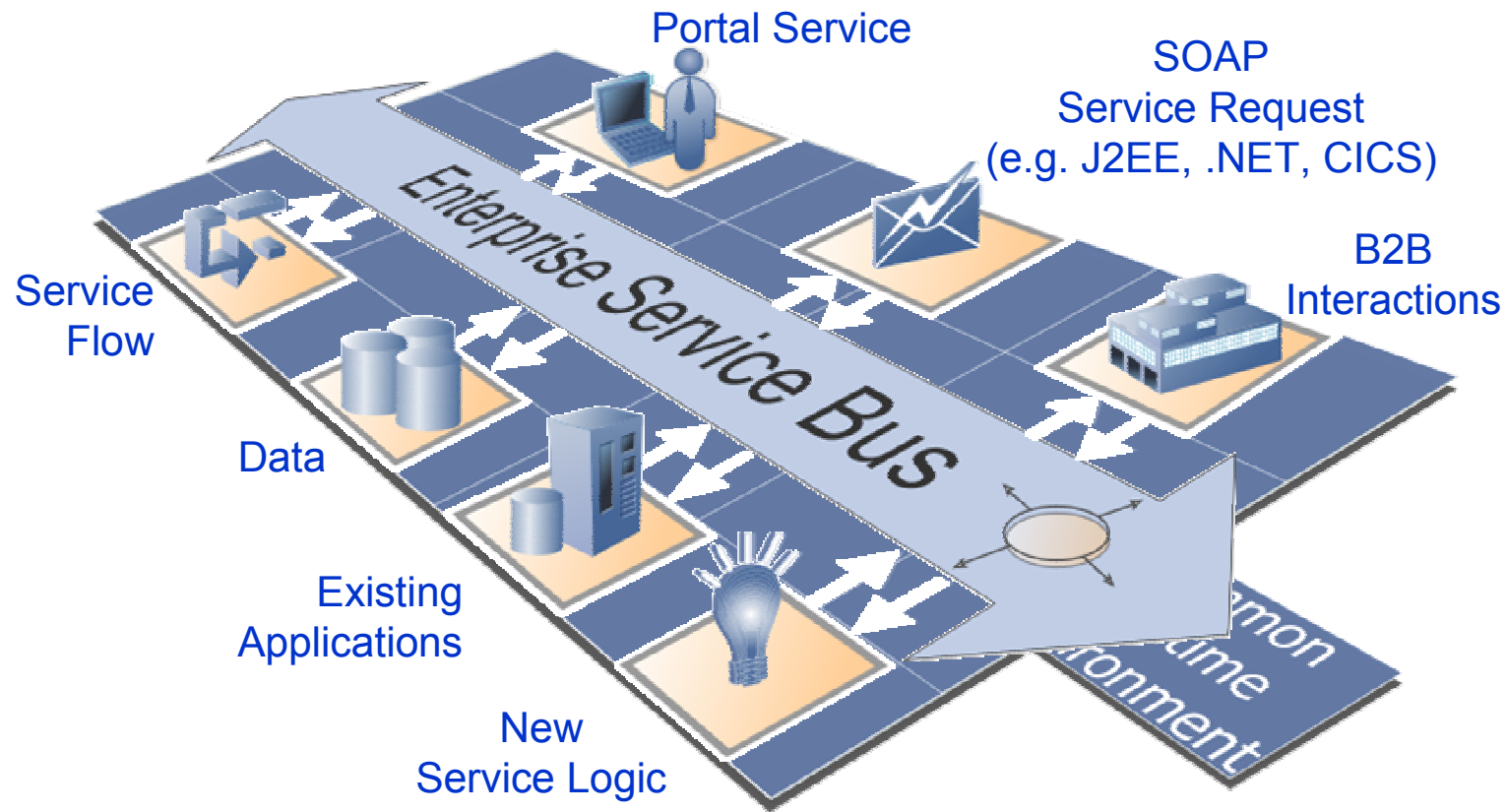


Services defined as units of business logic separated from:

- Flow of control and routing
- Data transformation and protocol transformation

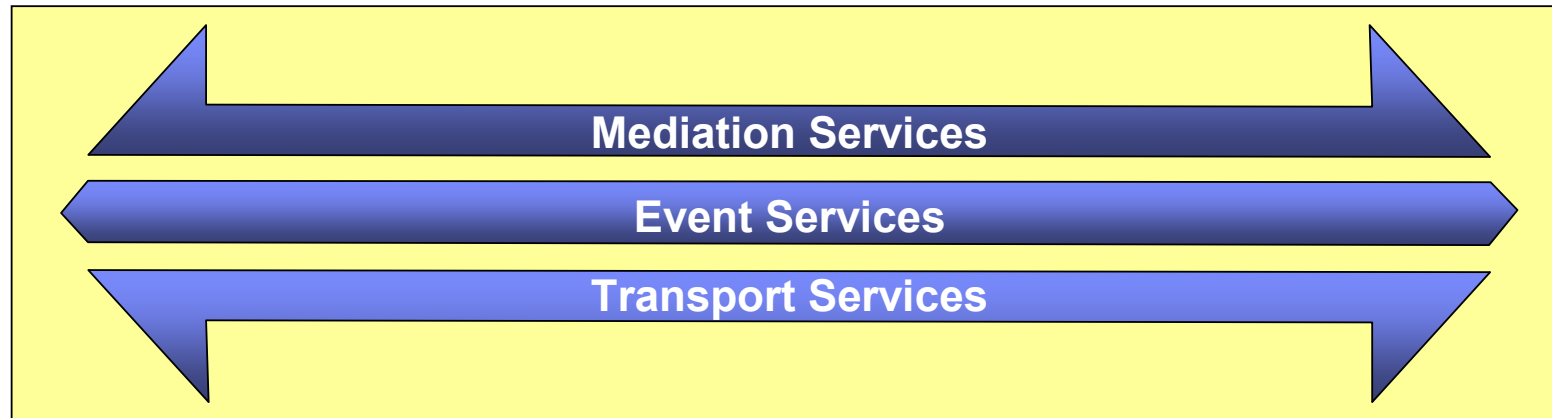
# What is ESB?

The connectivity layer within SOA



# What is ESB?

## ESB Communications Services Services Inherent to ESB



### Transport Services

- ✓ Assured delivery
- ✓ Secure delivery
- ✓ Transactional delivery
- ✓ Manageable delivery
- ✓ Delivery replay
- ✓ Modifiable qualities of transport.

### Event Services

- ✓ Event detection
- ✓ Event triggering
- ✓ Event distribution
- ✓ Complex Event Processing (CEP).

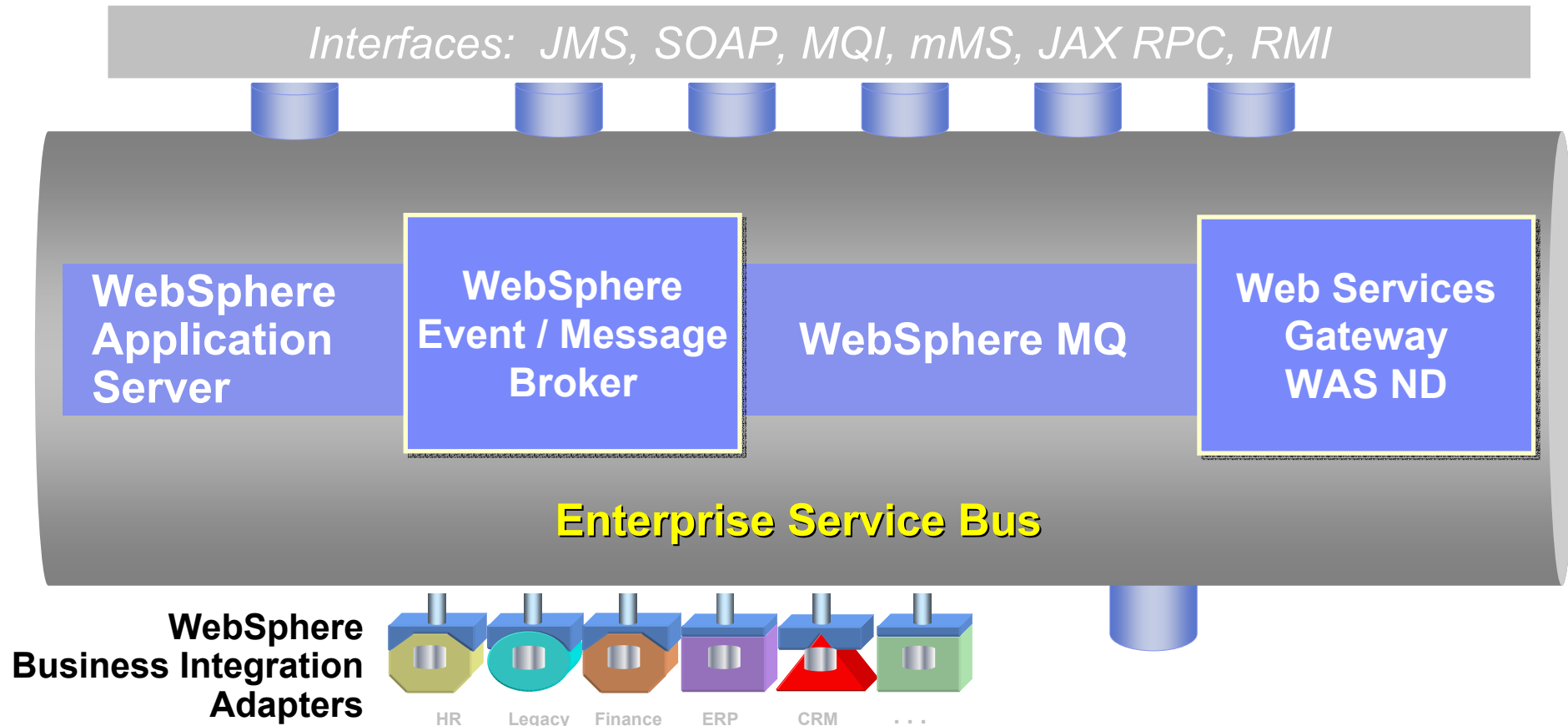
### Mediation Services

- ✓ Routing
- ✓ Transport switching
- ✓ Programming model switching
- ✓ Transformation & content augmentation
- ✓ Customized communications.

**Supporting yesterdays, today's... and tomorrow's standards.**

IBM customers are building the Enterprise Service Bus using existing WebSphere products

- **The ESB is emerging to support the acceptance of Service Oriented Architectures**





# What is ESB?

## ESB Implementation Options

- ESB is an architectural pattern incorporating the following approaches to integration
  - ▶ Message Oriented
  - ▶ Event Driven
  - ▶ Service Oriented
- Realize ESB implementation using one of the following WebSphere products as foundation
  - ▶ WebSphere MQ
  - ▶ WebSphere Message Broker/Event Broker
  - ▶ WebSphere Application Server V6
    - Web Services Gateway
- Advocate incremental enhancement of customers connectivity services
  - ▶ WBI Adapters (Access Services)
  - ▶ WBI Connect (Partner Services)
  - ▶ WBI – SF (Process Services)
  - ▶ Other appropriate services

# Next Step SOA

## SOA on zSeries

### ▪ **Services provider (server)**

- ▶ WAS, WBI Server Foundation (z/OS and zSeries Linux) for deployment of J2EE services
  - J2C connectors, JMS/MQ integration with CICS/IMS/DB2, WBI Adapters
- ▶ SOAP for CICS
- ▶ IMS SOAP Gateway
- ▶ DB2 UDB for z/OS SOAP support
- ▶ WBI Message Broker

### ▪ **Services consumer (client)**

- ▶ WAS, WBI-SF z/OS & zSeries Linux J2EE
- ▶ SOAP for CICS
- ▶ ~~IMS SOAP Gateway~~ - *(not yet – known re*
- ▶ DB2 UDB for z/OS – services calls from S
- ▶ WBI Message Broker

### ▪ **Development tools**

- ▶ Enterprise COBOL, PL/I – support for XML
- ▶ WebSphere Studio Enterprise Developer
- ▶ Rational tool integration with zSeries

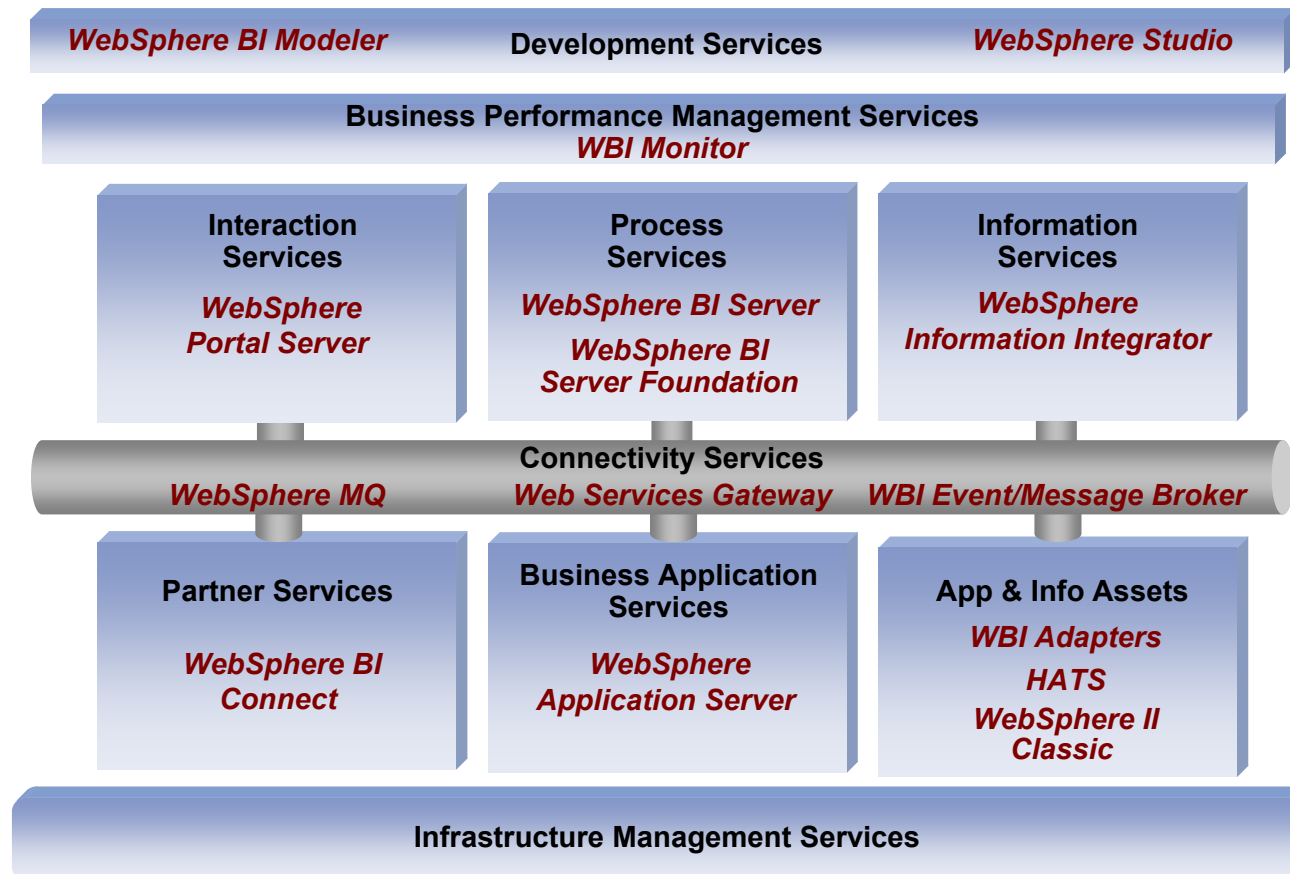
IBM Software Group





# Architecture Référence Business Integration

## IBM WebSphere Software Offerings



*Augmenter la qualité de services pour répondre aux besoins métier*

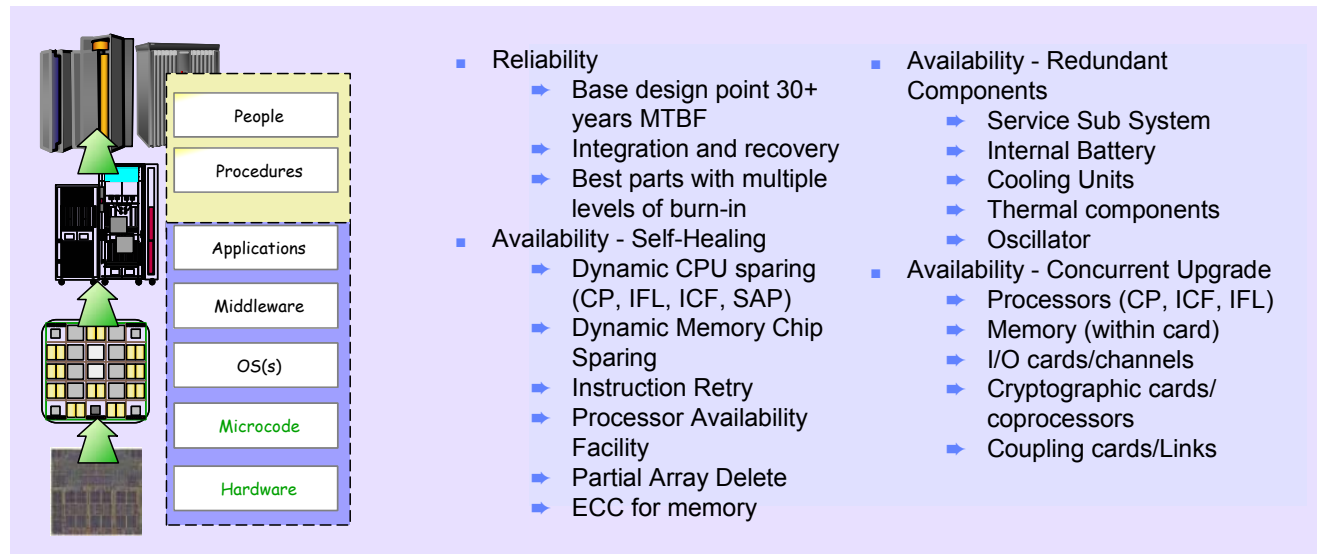
# When WebSphere on zSeries?

## zSeries difference

zSeries Value - Why zSeries?

# The zSeries Difference

## Single system Hardware reliability and availability



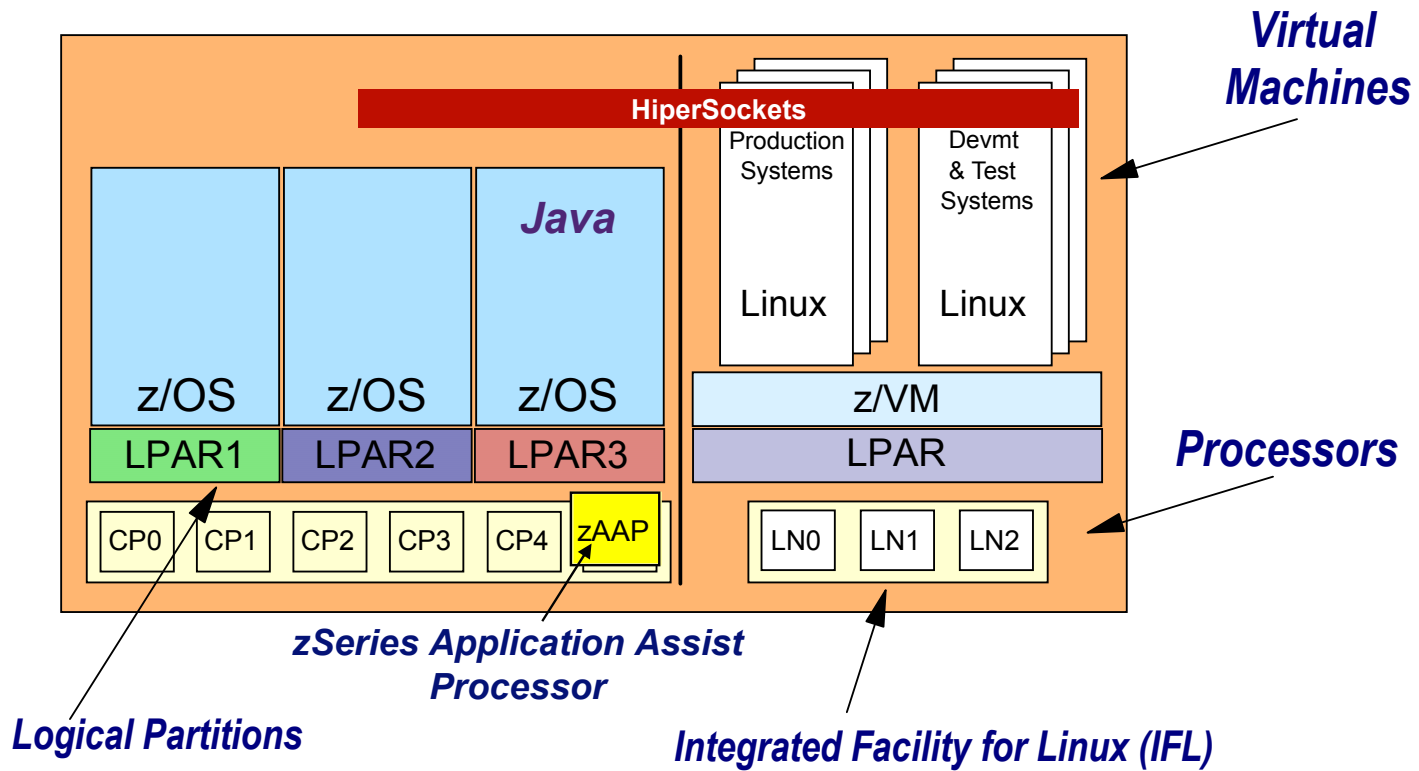
- zSeries philosophy: The more errors that can be prevented at the lowest level — at the hardware and microcode levels — the less impact on applications and operators and end users.

pSeries has many of the hardware availability features to lead the industry in comparative functionality with other high-end UNIX platforms. But pSeries is not same as zSeries. For example, spare processors are not standard, there are no dual instruction processor execution, and no storage protect keys on pSeries.

For more on z990 availability - <http://www.research.ibm.com/journal/rd/483/fair.pdf>

# The zSeries Difference

## Sample IBM eServer zSeries™ Configuration

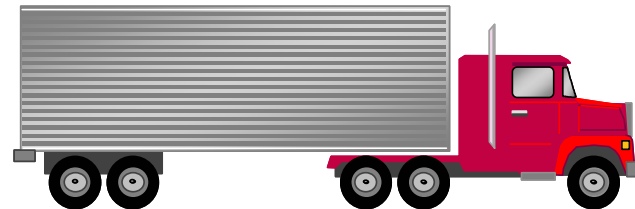


# The zSeries Difference

## Define the race - then pick the vehicle

What the environment and the characteristics tell us

- Typical industry benchmarks for Linux and UNIX tend to measure the performance of a single server running a single application and results tend to be highly dependent upon processor speed
- Stand alone processors run at higher speeds than mainframes, hence they look better in most typical industry benchmarks
- Mainframes distinguish themselves through outstanding capacity, usually not measured by typical industry benchmarks
- The work performed by multiple stand alone servers is a good candidate for consolidation when:
  - The servers are lightly to moderately loaded
  - The servers do not peak concurrently



*A race car goes faster than a trailer truck, but if the contest is which one can move 100 refrigerators across the country fastest, bet on the truck.*

# The zSeries Difference

## Workload types for zSeries & Unix

- zSeries: workload characteristics that perform especially well
  - ▶ High data movement
  - ▶ Task switching between different workload types
  - ▶ High volume, unpredictable workload
  - ▶ High availability requirement
  - ▶ Application modernization integrated with current zSeries applications
  - ▶ New J2EE applications with new expanded functionality and flexibility
  - ▶ Stringent Service Level Agreement requirements
  - ▶ Consolidation of many lowly utilized servers / cost reduction requirements
  - ▶ Requirement for large volume of SSL
  - ▶ Major ISVs, selected mid-market application support
  
- UNIX workload characteristics that perform especially well
  - ▶ CPU intensive
  - ▶ Minimize switching between workloads to not flush cache
  - ▶ Availability, Service Level Objectives more demanding than Windows but not as demanding as for z/OS parallel sysplex
    - Can tolerate outages for OS or DBMS changes in a cluster
    - Can tolerate outages for scheduled DBMS maintenance activities
    - Can tolerate outages for moving DBMS node to backup if node fails
  - ▶ More choices of ISV software

How we can help you with

**Business**

**Integration**

**Architecture**

**Workshops**

# BIAW

- 2 days workshop with your IT Architects, zSeries specialists & other subject matter experts
- Objective of the workshop:
  - Customer describes the actual system and the needed/possible evolutions of this system
  - IBM shares its vision of the reference architecture and technologies that supports this architecture.
  - IBM and Customer teams work together on possible scenario/architecture/technologies that can match the customer's objectives
  - All the work produced during the workshop is documented and shared with the customer
- The workshop must be prepared
  - Scope definition
  - Any information that is valuable to setup the workshop, have the right skills in the room, ...
- The workshop is FREE



*Thank You*