



# Introduction to WebSphere and WebSphere XD on z/OS

## **System z** Software Summit

*Snehal S. Antani*

*antani@us.ibm.com*

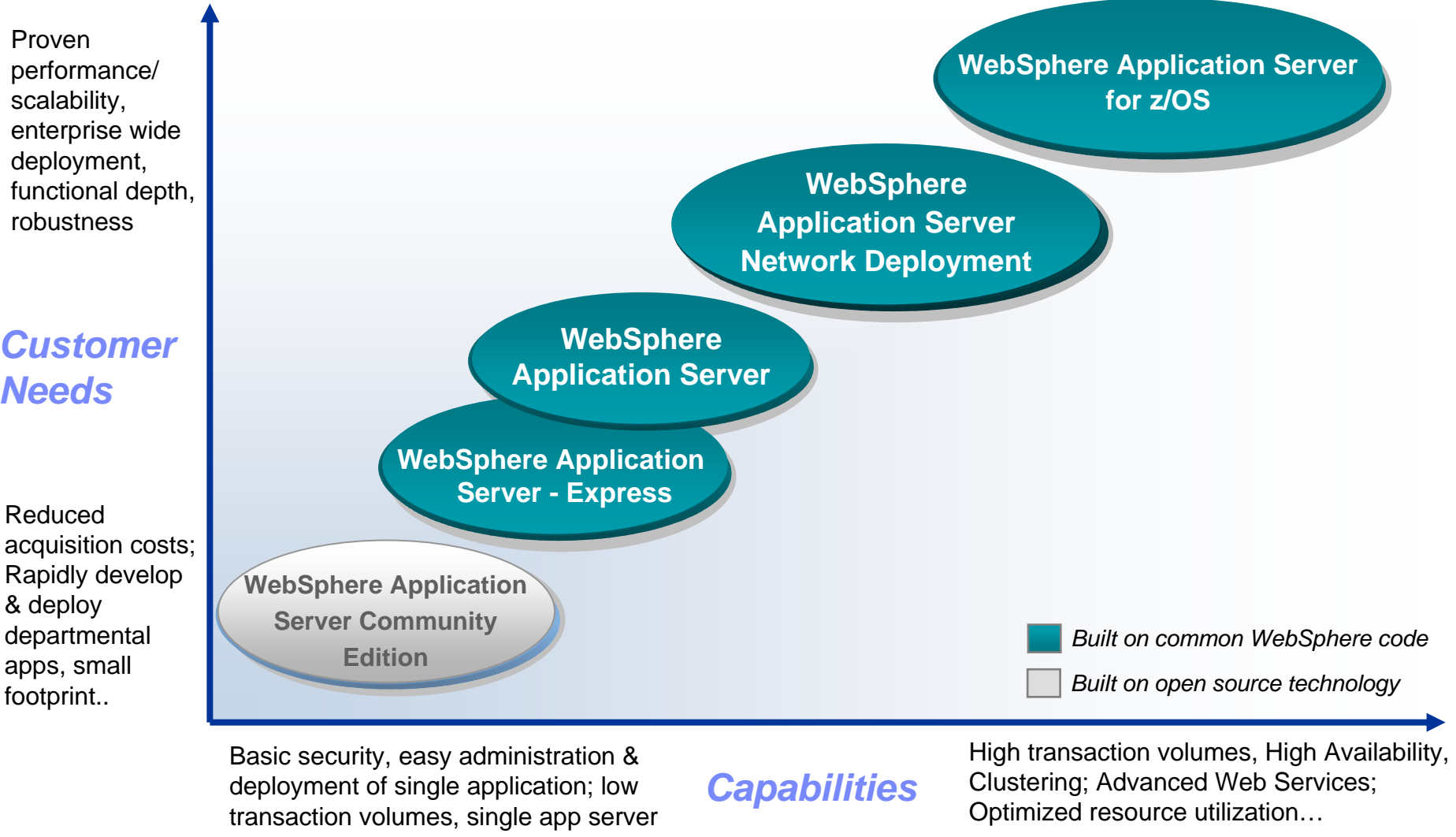


# Topics of Interest

- Eliminate server sprawl by consolidating and simplifying using WebSphere Application Server for z/OS
- Centralize and reuse enterprise applications for batch access and online-transaction access using WebSphere Extended Deployment for z/OS®
- Support Web Services that enable core assets, such as CICS transactions, to be reused as a service with the CICS Transaction Server



# Grow as needs evolve, leveraging core skills and assets



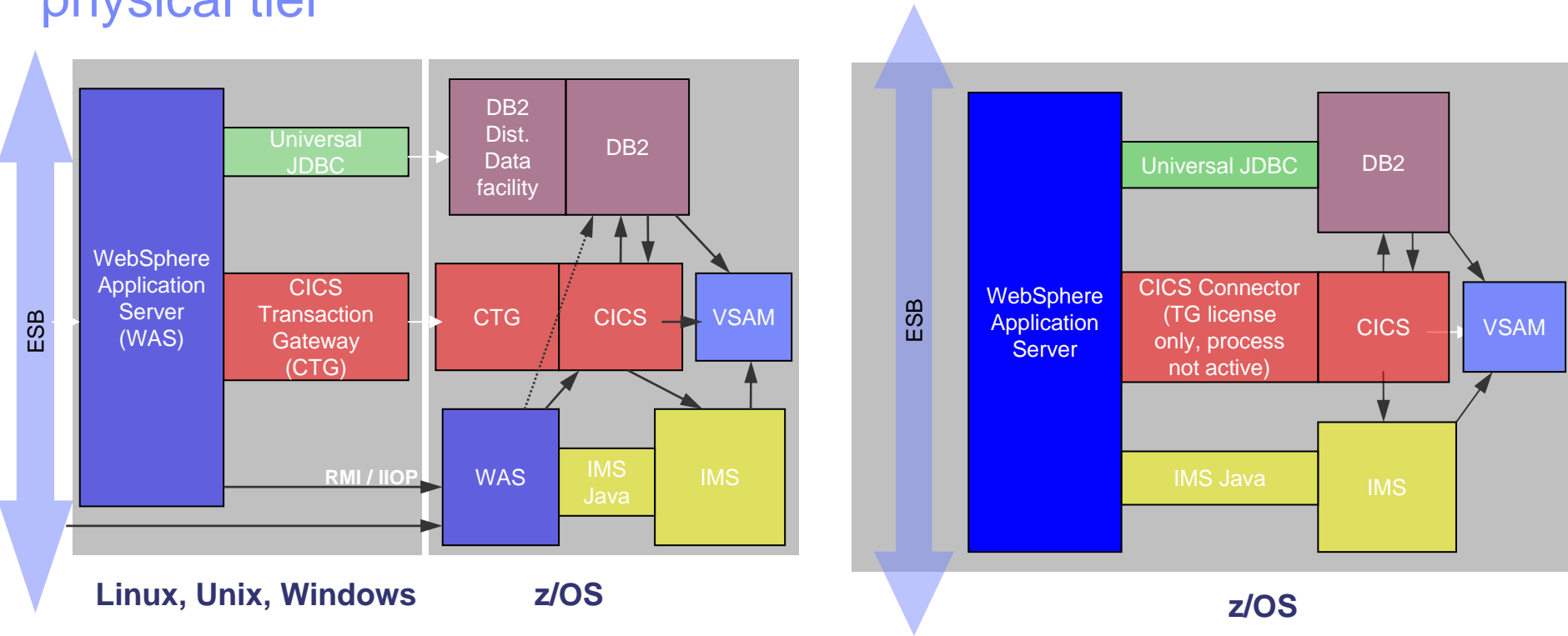
# Technical Business Problems



Customers are driven to the mainframe for their middleware deployments because they are encountering:

- Server sprawl, resulting in a complicated IT infrastructure
- Inability to effectively track costs associated with distributed server environments
- Workloads not meeting, or in danger of missing, service level agreements
- Spiraling costs associated with maintaining and managing existing IT environment
- Potential security loopholes with mission critical business transactions
- Significant investments in existing core business environments

# Enterprise data access with multiple physical tiers vs a single physical tier



Moving from multiple physical tiers to a single tier provides many benefits, including tighter security, enhanced management, reduced costs, and performance gains.

# Simplify management and security with single tier

*Using WAS for z/OS enables more flexible and rich deployment configurations*

## Management benefits

- Fewer components to manage
  - ✓ Hardware servers
  - ✓ Network Infrastructure
  - ✓ Operating Systems
- zSeries Platform Scalability
  - ✓ Vertical (Up to 16 CPU per image)
  - ✓ Horizontal Scalability (Parallel Sysplex)
- Smaller Physical Footprint and resource utilization
- Single end to end management interface / console for all products within the architecture
  - ✓ WebSphere cross platform administrative interface
  - ✓ MVS Console, System Monitor Facility

## Security benefits

- Single security authority for authentication and authorization
- Simplified Compliance Certification and Analysis (SOX, HIPAA, etc)
- Tight integration of WebSphere and RACF via SAF
- End user authentication for authorization and auditing throughout the architecture
- No network between tiers
  - ✓ Reduced Opportunity for intercepting transmissions (
  - ✓ No need to encrypt data or tunnel between tiers



# Add speed and resiliency with single tier

*The mainframe is an optimized integration environment with benefits!*

## High availability benefits

- Fewer tiers to HA enable and manage
- Proven Parallel Sysplex HA enablement
- Parallel Sysplex exploitation by all program products
- Simplified recovery procedures
- Faster restart times
- Operating system service (ARM) for system restart

## Performance benefits

- No network time
- No product specific network protocol construction / deconstruction
  - ✓ DRDA (DB2)
  - ✓ CICS Transaction gateway (CICS)
  - ✓ RMI / IIOP (IMS & J2EE)
- Reduced CPU utilization
- Hi speed access to cache without network
  - ✓ Parallel Sysplex infrastructure infrastructure for DB2, CICS, IMS, MQ
  - ✓ Guaranteed integrity and currency of data (Pessimistic data access)
- Improved two phase commit performance

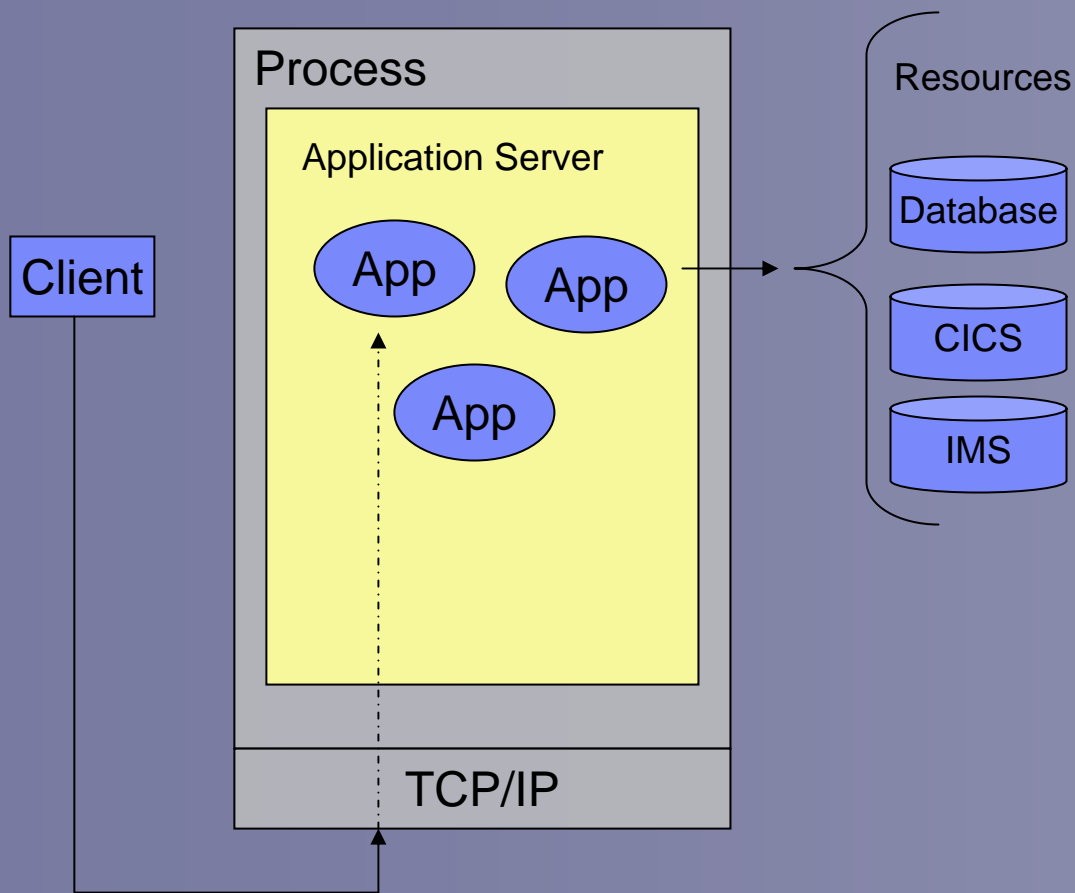
# WebSphere for z/OS Architecture

- Why is WebSphere for z/OS
  - Highly available?
  - Scalable?
- WebSphere for z/OS Topologies





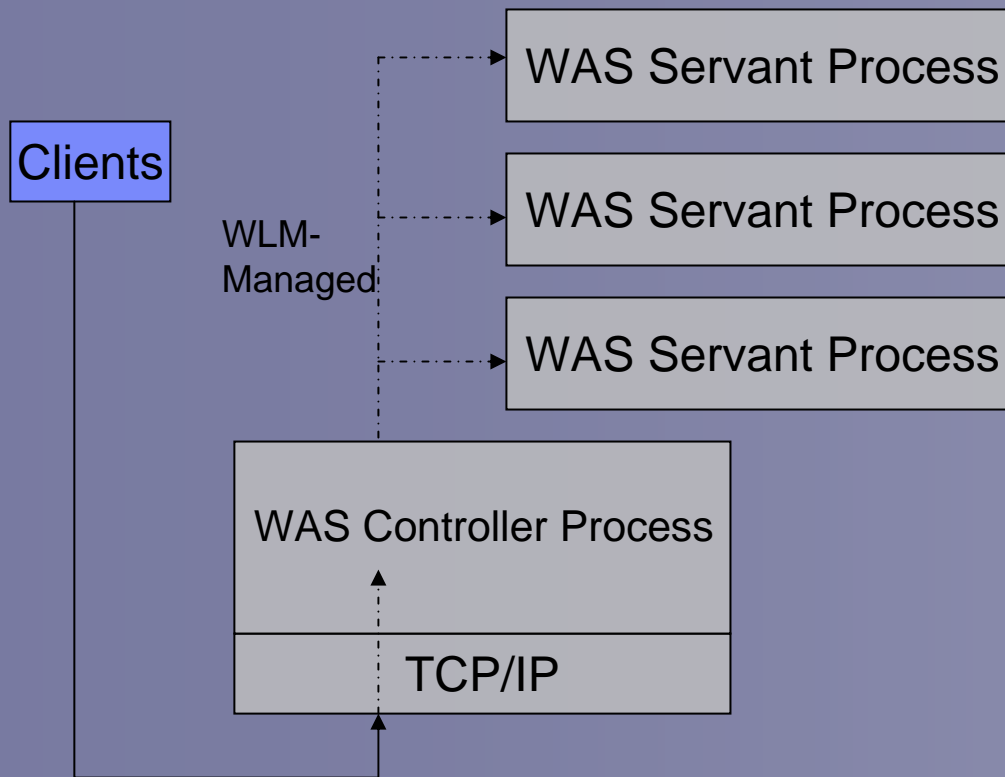
# Typical Middleware architecture



- Typical application server runs as a single process
- Fatal exceptions that stem from Applications can terminate the entire server process.
- In order to scale, we would start multiple server processes.
  - ▶ Many aspects of these processes could reside as singletons within the entire server environment.
  - ▶ Allowing duplicate pieces of those types of components to run is not an efficient use of resources.
- Termination of a Server process leads to unavailability of the application unless the application is clustered.

# WebSphere for z/OS Server Architecture

## Single WAS z/OS Application Server

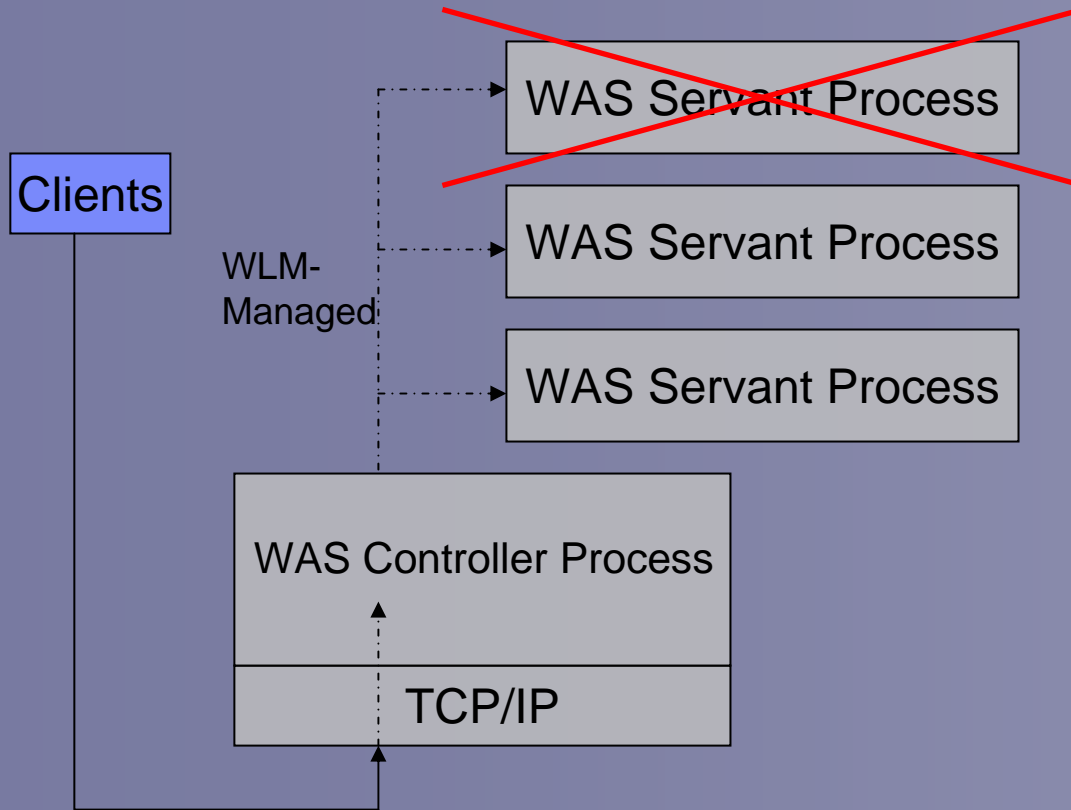


A WAS z/OS application server is comprised of multiple processes.

- ▶ Only authorized, robust system code is executed within the Controller process.
- ▶ Application code is executed within the Servant Process.
  - Theoretically, Servant processes are clones of each other.
  - They can be terminated or created without interrupting the overall end-user experience
  - Servant processes can be dynamically created or destroyed by Work-load Management. The decisions can be based on user-defined service policies.

# Why is WAS z/OS Reliable?

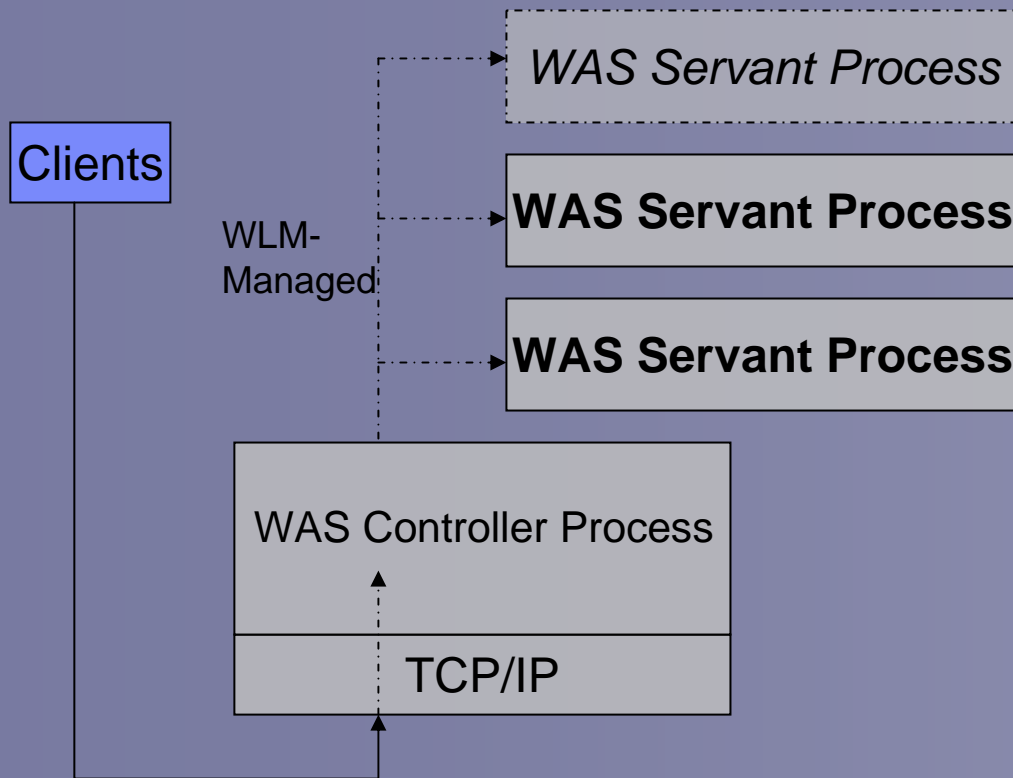
## Single WAS z/OS Application Server



- If a servant process is terminated, the application can still be served by any of the other servants within the Application Server

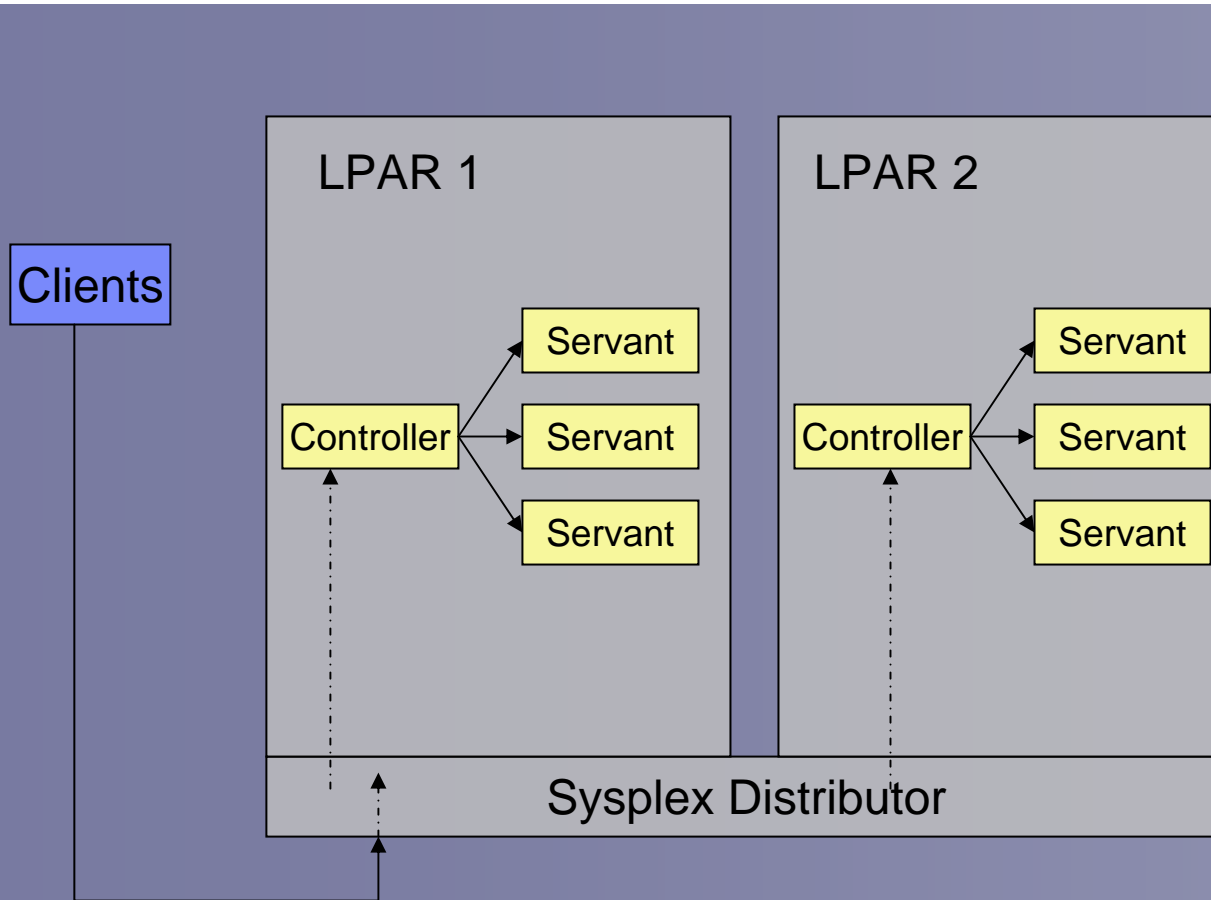
# Why is WAS z/OS Scalable?

## Single WAS z/OS Application Server



- Within a WAS z/OS Application Server, one can define a minimum and maximum number of servant processes that are allowed.
- As workload increased, WLM can detect that the Servant Processes are not able to keep up with the work (per defined service policies). If the maximum number of servants has not been hit, WLM can start a new Servant process to handle the load.

# WAS z/OS Basic High-Availability Topology



- In order to provide a true high-availability server, the Controller processes would be clustered.
- If a controller process fails, the Sysplex Distributor would route work to the other controller process.

# WAS z/OS Topologies

## WAS z/OS Interactions with an EIS/DB

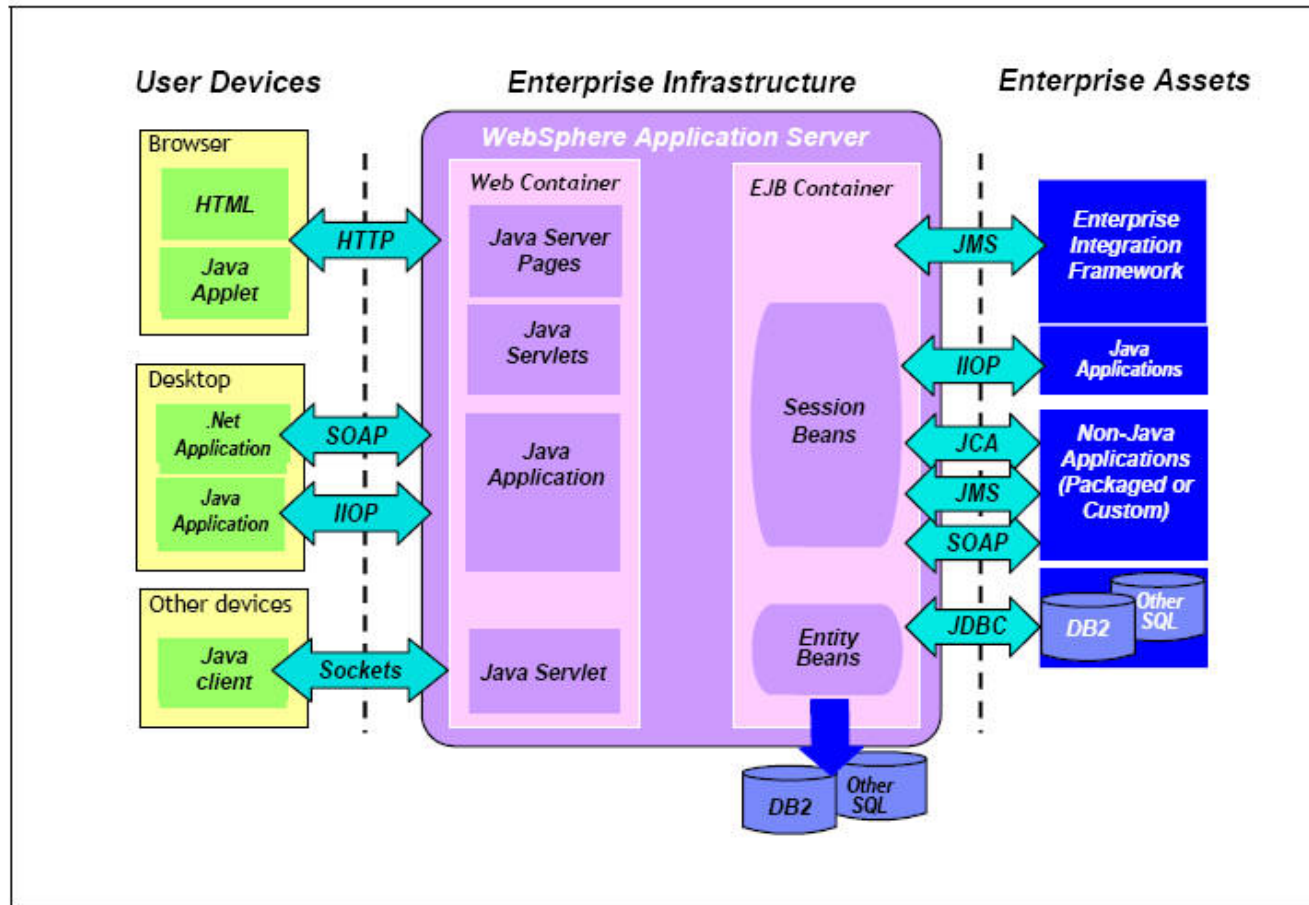
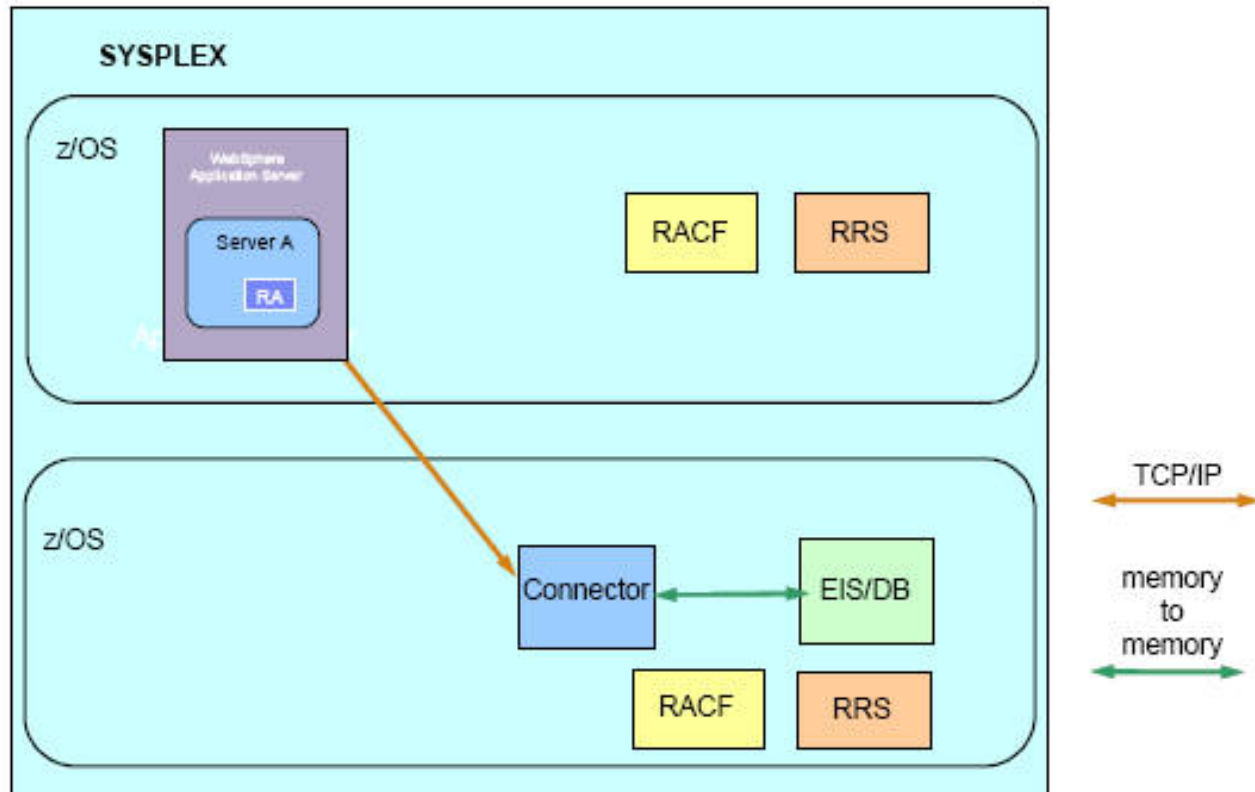


Figure 2-1 Inbound and outbound access to/from WebSphere

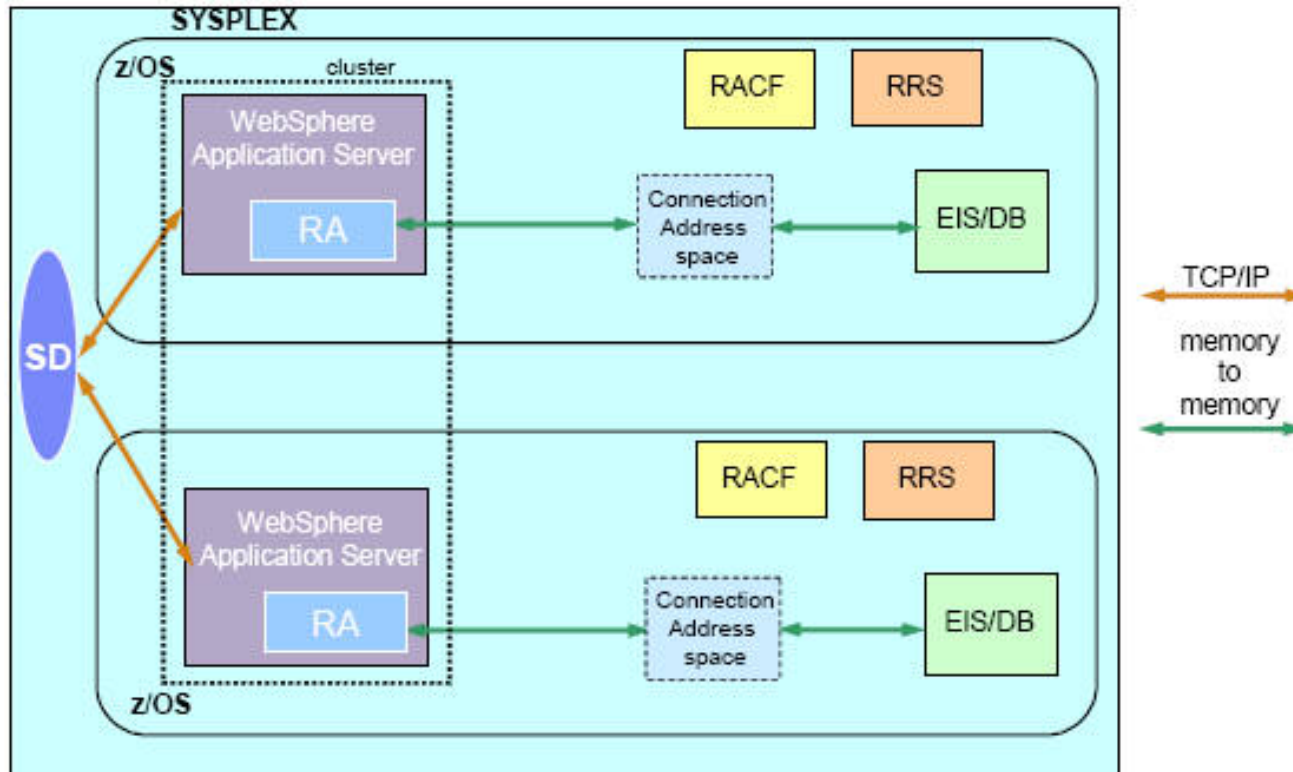
# WebSphere z/OS-Topologies

Connecting the Application Server remotely to some EIS/DB



# WebSphere z/OS-Topologies

Alternative method to connecting the Application Server to EIS/DB's





# What is WebSphere Extended Deployment?

What if you could deliver enterprise level function to your existing application infrastructure in a non-invasive fashion?

Provides a robust environment for business applications

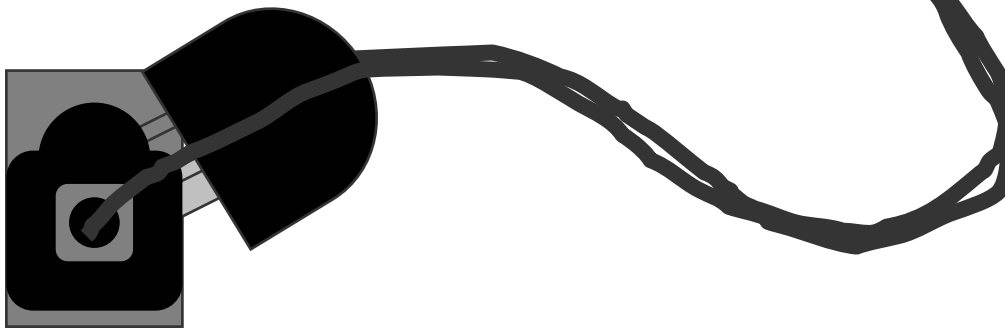
- ✓ J2EE 1.4
- ✓ Advanced Web Services

WebSphere Application Server for z/OS

Applications

WebSphere Extended Deployment for z/OS

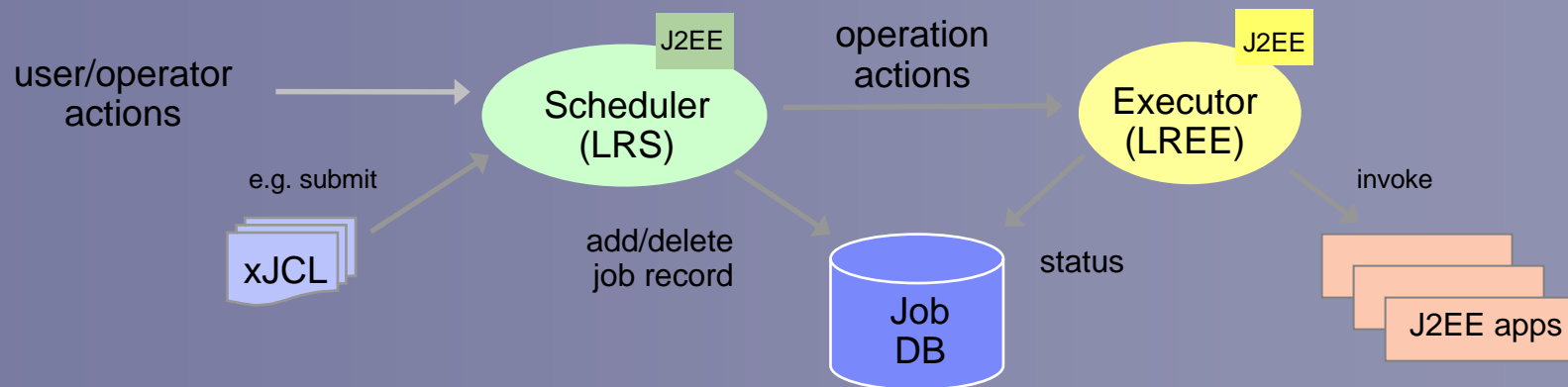
- ✓ Secure and reliable
- ✓ Common Web based administration
- ✓ Highest scalability & advanced differentiated WLM
- ✓ Integrated with z/OS and critical z/OS assets



# Flexibility

- WebSphere Batch environment:
  - ✓ Designed for structured J2EE batch workloads
  - ✓ Scheduling agent to ensure batch workloads are disseminated to garner unused WebSphere resources
  - ✓ Service policy support to differentiate workload importance
- WebSphere & non-WebSphere server support
  - ✓ Simplifies the management of diverse servers

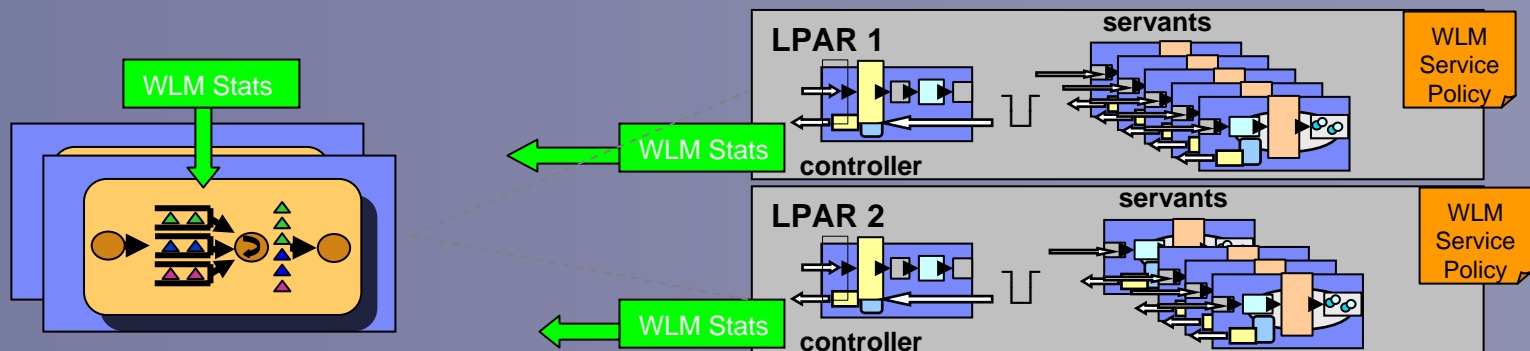
## *XD-Batch End-to-End flow*



# Intelligence

*WebSphere Extended Deployment for z/OS provides intelligence for application servers to share information and workloads*

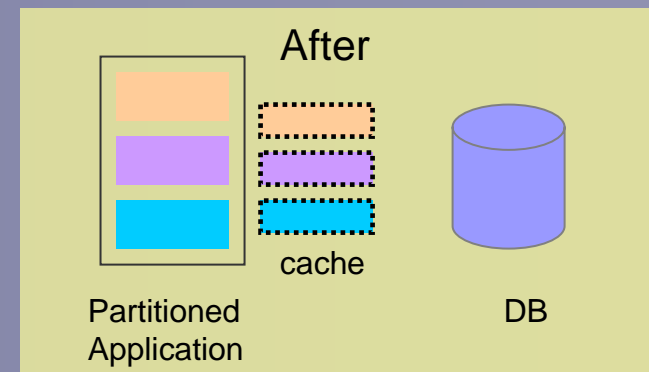
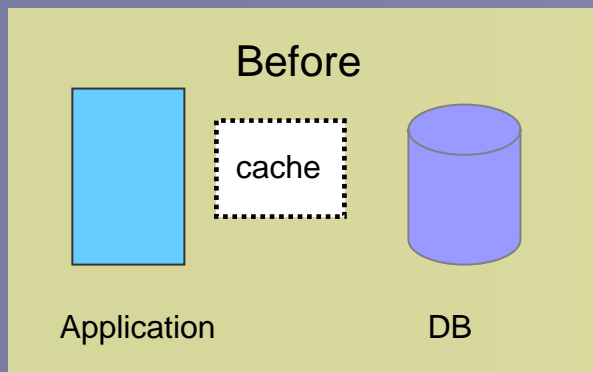
- On Demand Router (ODR)
  - ✓ Provides an intelligent proxy for workload routing within and across LPARs
- Health Policy
  - ✓ Proactively seeks software maladies, such as hung servers, excessive memory consumption, storm drain situations, and addresses them
- Service Policies
  - ✓ Enables granular classification of application workloads which is addressed within the ODR
  - ✓ Value is compounded by z/OS WLM which manages XD application workloads, as well as all zSeries workloads



# Performance

*WebSphere Extended Deployment for z/OS helps eliminate performance hindrances with smart datasource access and caching*

- Partitioning Facility
  - ✓ Ability to break applications into partitions which enable smart, consistent caching at a partition level
  - ✓ Highly available application partitions
  - ✓ Provides reduced lock contention on shared DB2 resources
- ObjectGrid
  - ✓ Creates a caching fabric for shared object data which reduces unproductive backend datasource transactions

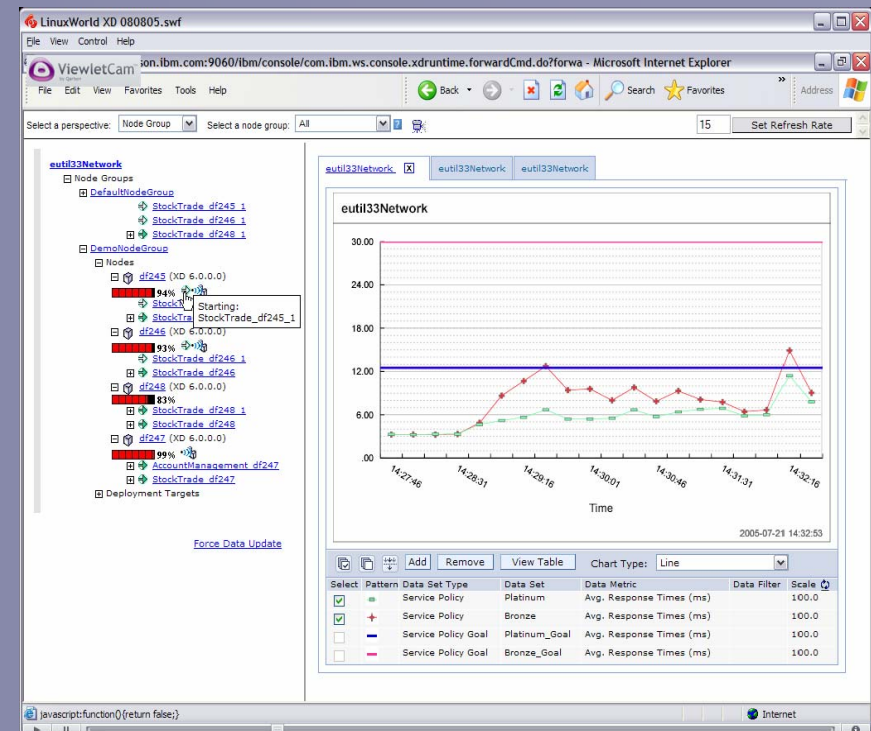


# Manageability

*WebSphere Extended Deployment for z/OS simplifies complex application environments*

XD for z/OS provides:

- Application Edition Manager
  - ✓ Provides the facility for multiple versions of production applications, including operational support for piloting, staging, and rollback
- Visualization
  - ✓ Customizable real-time charts display the success of your applications
  - ✓ Runtime maps enable at-a-glance assessments of the components of your application server cells
- Application lazy start
  - ✓ Deactivated applications are lazily started when requests come in



# Section

## *WAS XD-Dynamic Operations Overview*



# Dynamic Operations Overview

- Virtualized, policy-based, dynamic workload management
- Dynamic application placement
  - ▶ Enables starting and stopping server instances based on application load and user-defined goals
- On-Demand Router
  - ▶ Enhanced version of the Proxy Server
  - ▶ Controls request prioritization, flow, and routing in an Extended Deployment (XD) environment

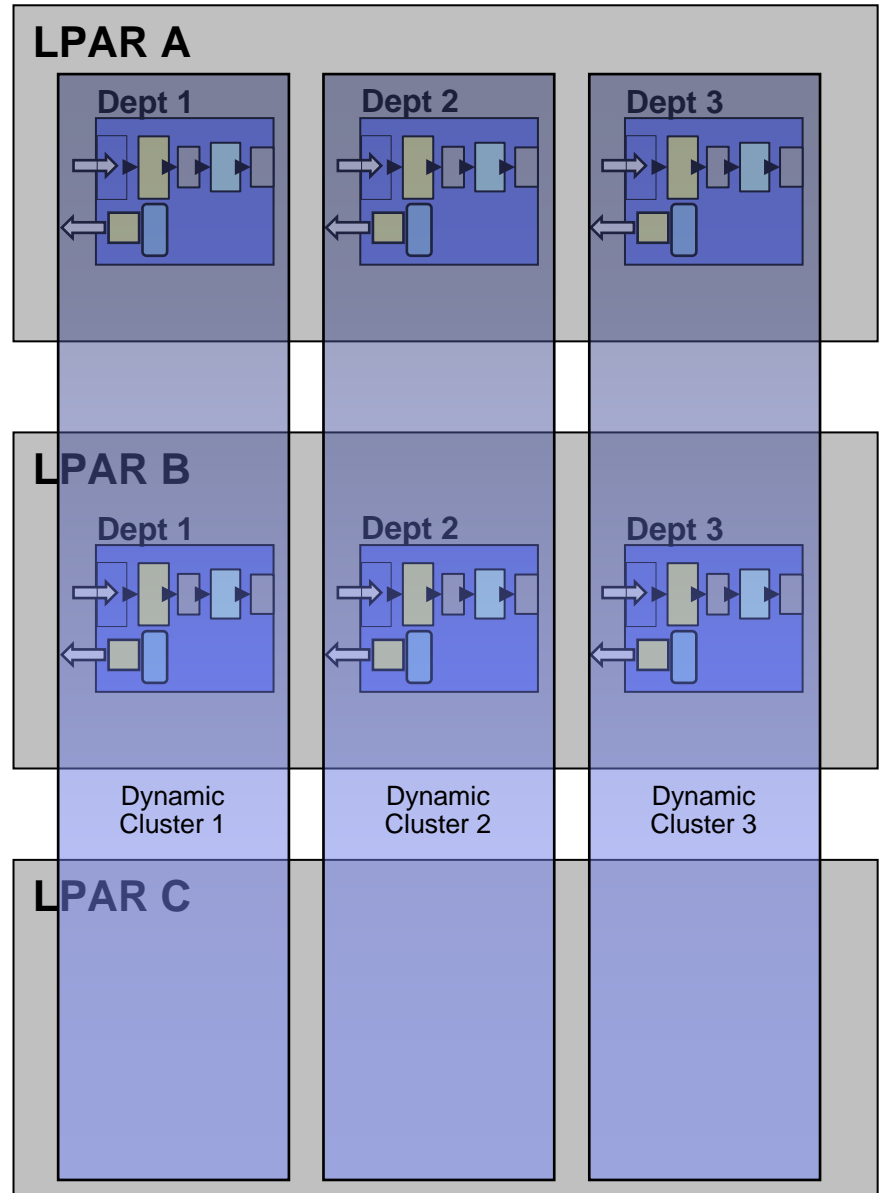
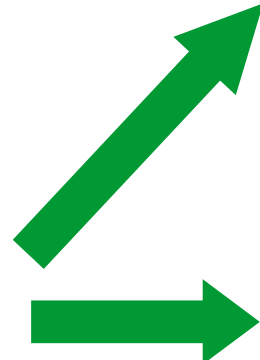
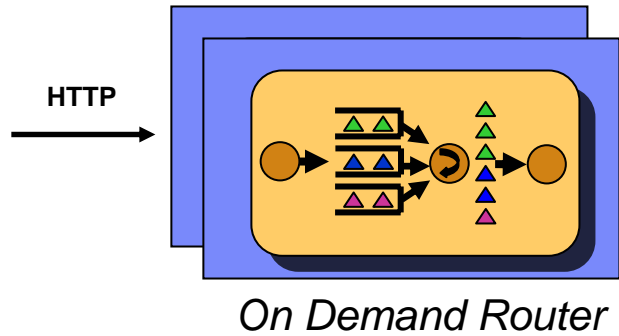
# Example: Overflow LPAR

- Three LPAR's
  - ▶ Two Primary LPAR's
    - Satisfy majority of requests
    - Provide redundancy
  - ▶ One Overflow LPAR
    - Limited memory available
    - Can only have a few address spaces running

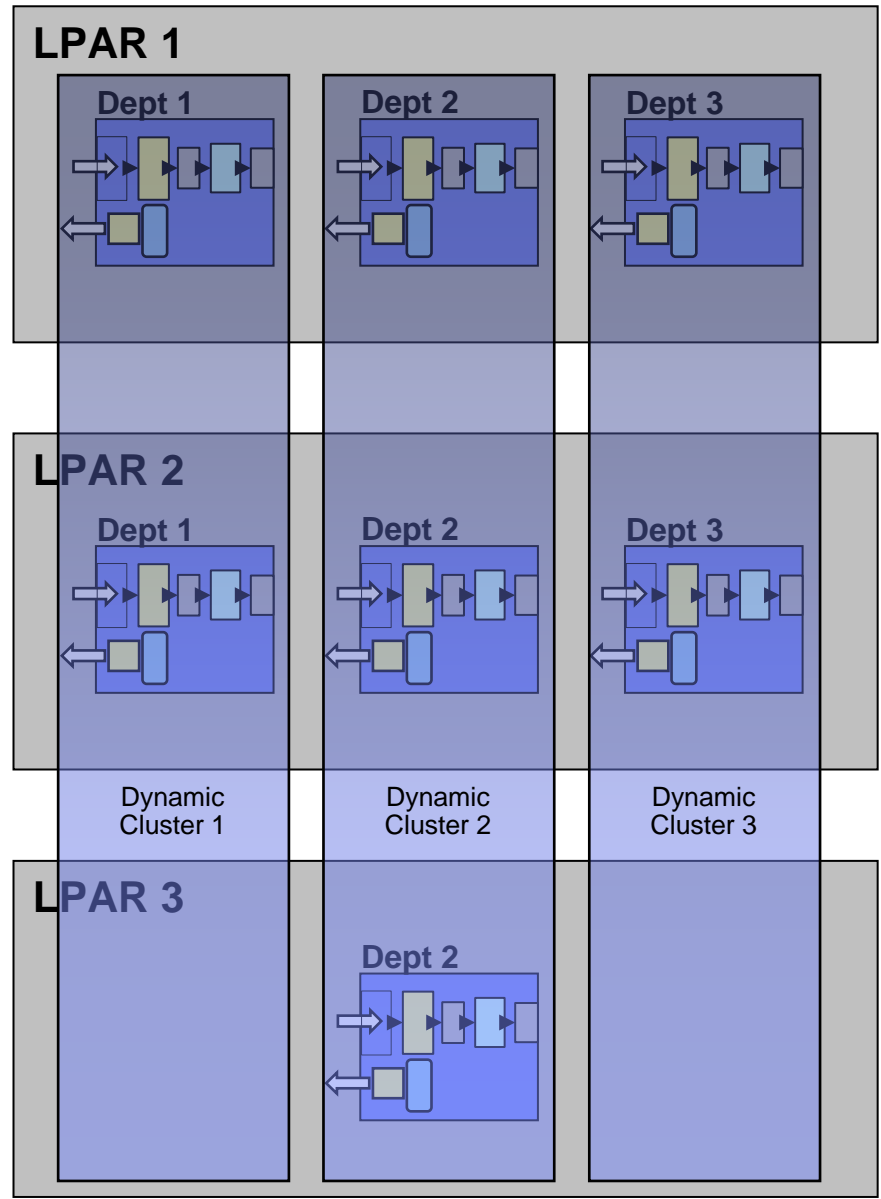
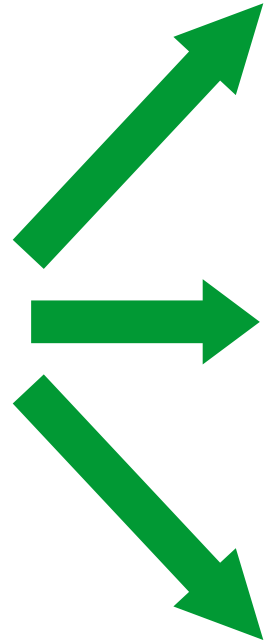
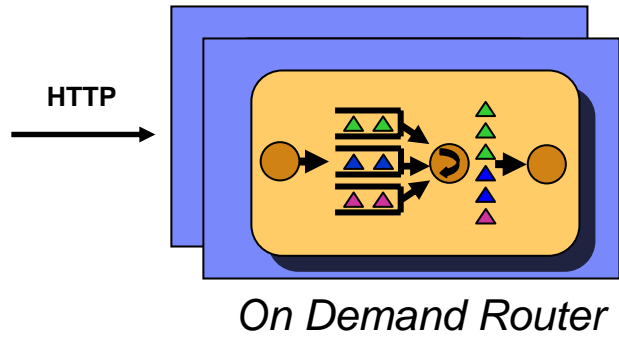




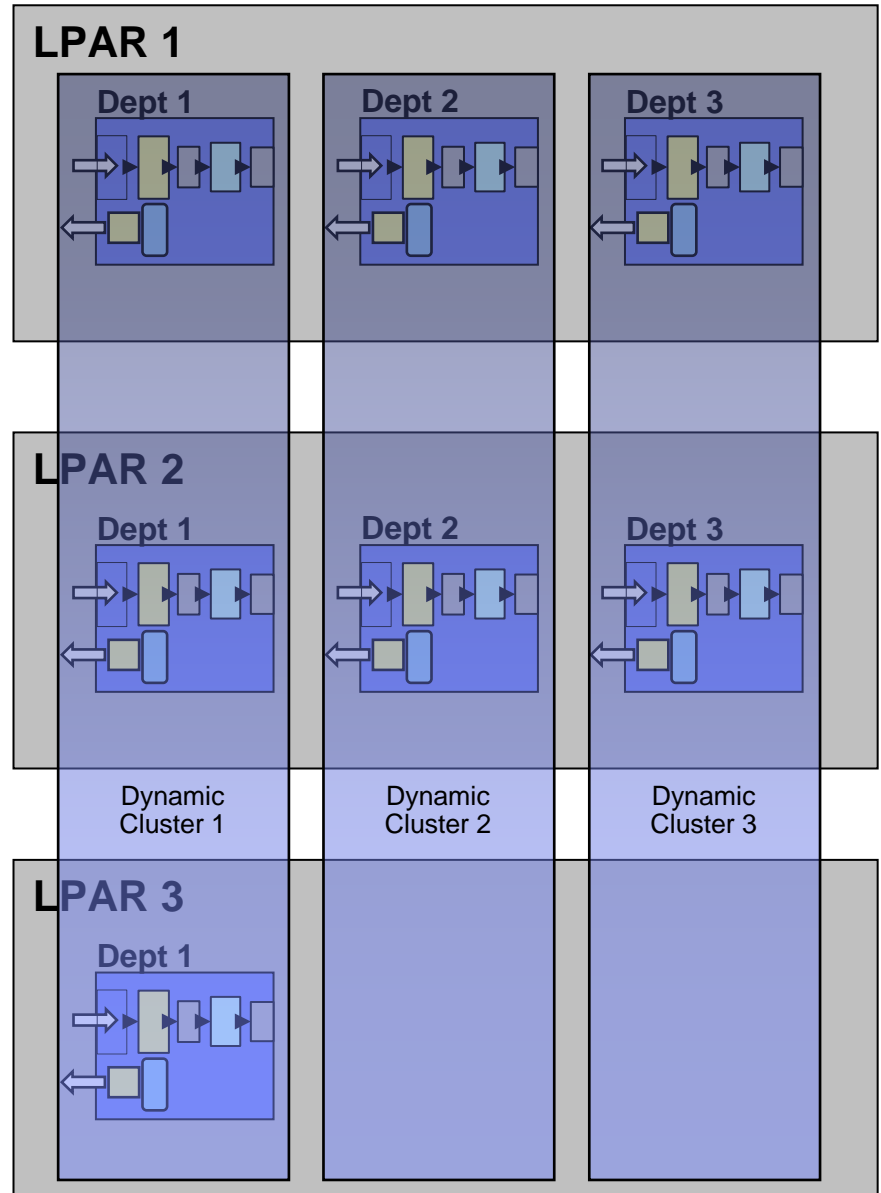
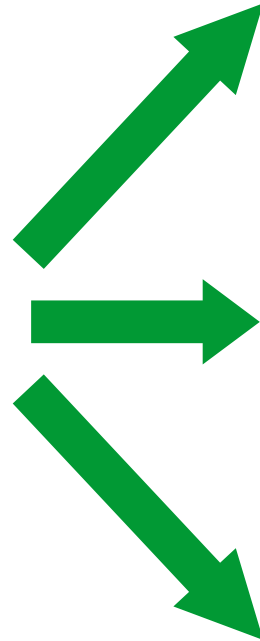
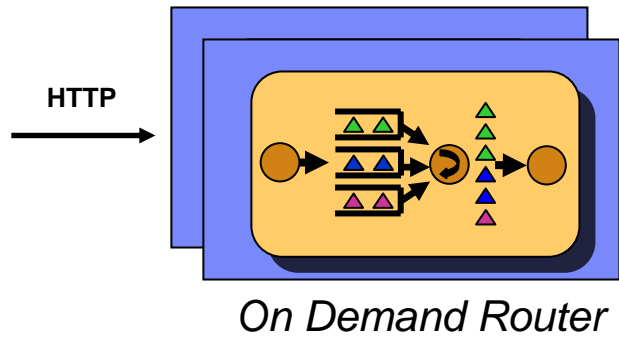
# Example (cont)



# Example (cont)



# Example (cont)



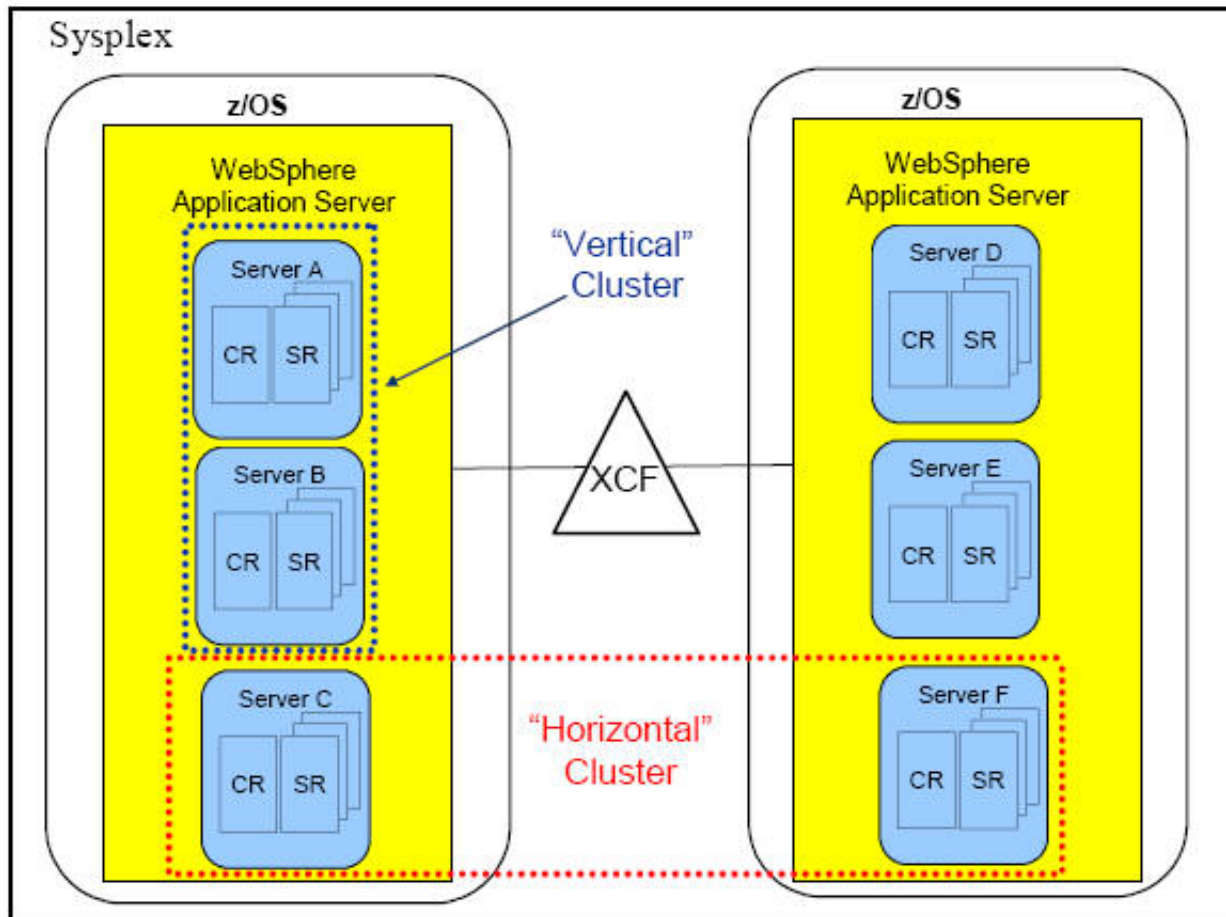
# Benefits

- Enables more efficient hardware utilization
  - ▶ Dynamic allocation of resources to handle variations in traffic
    - Takes advantage of differing peak times
  - ▶ Server consolidation reduces total cost of ownership
- Helps ensure a consistent level of service for critical applications
  - ▶ Decisions are based on user-defined policies
  - ▶ In times of contention, more important requests will perform better than less important requests



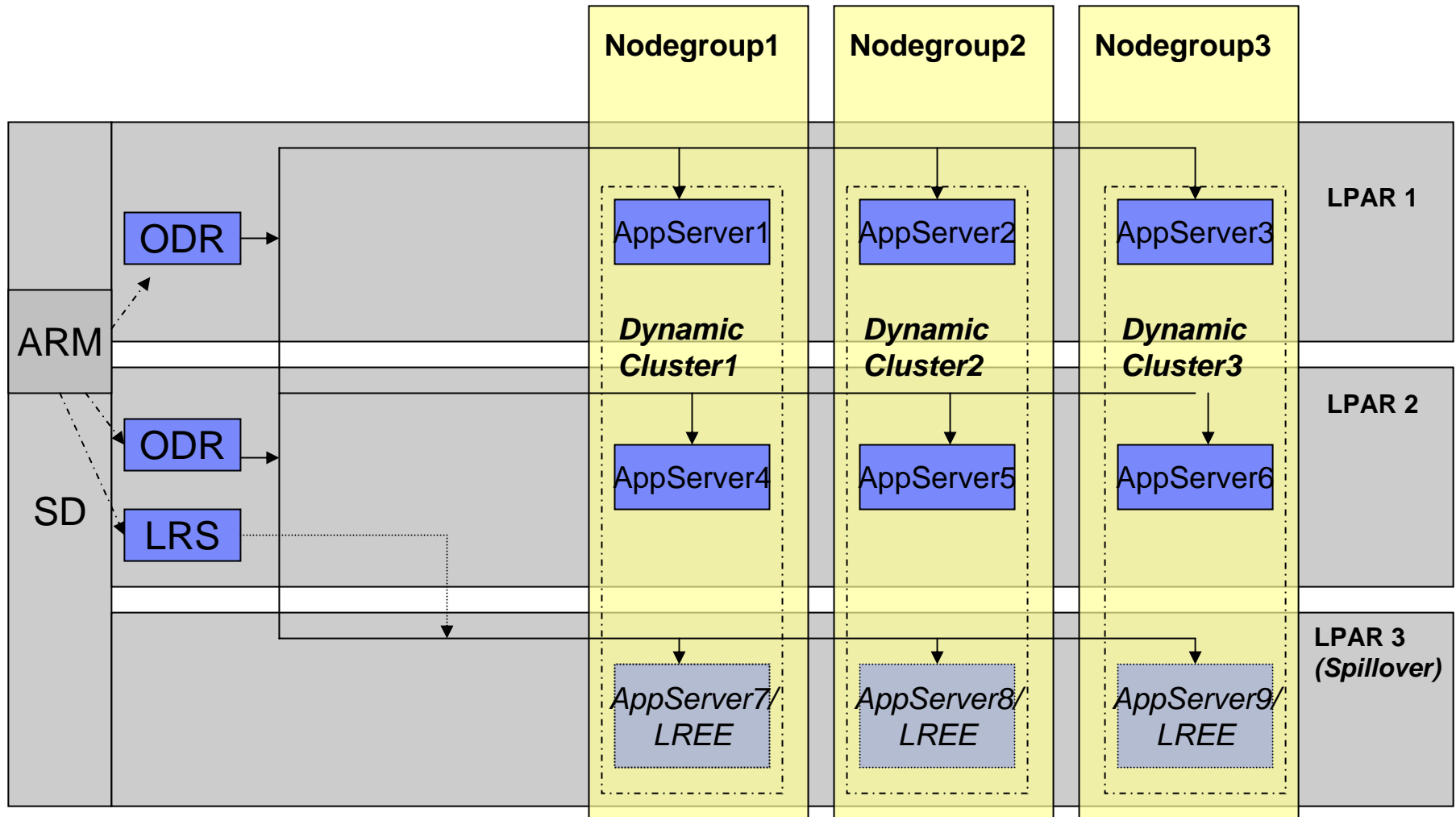
# WebSphere XD

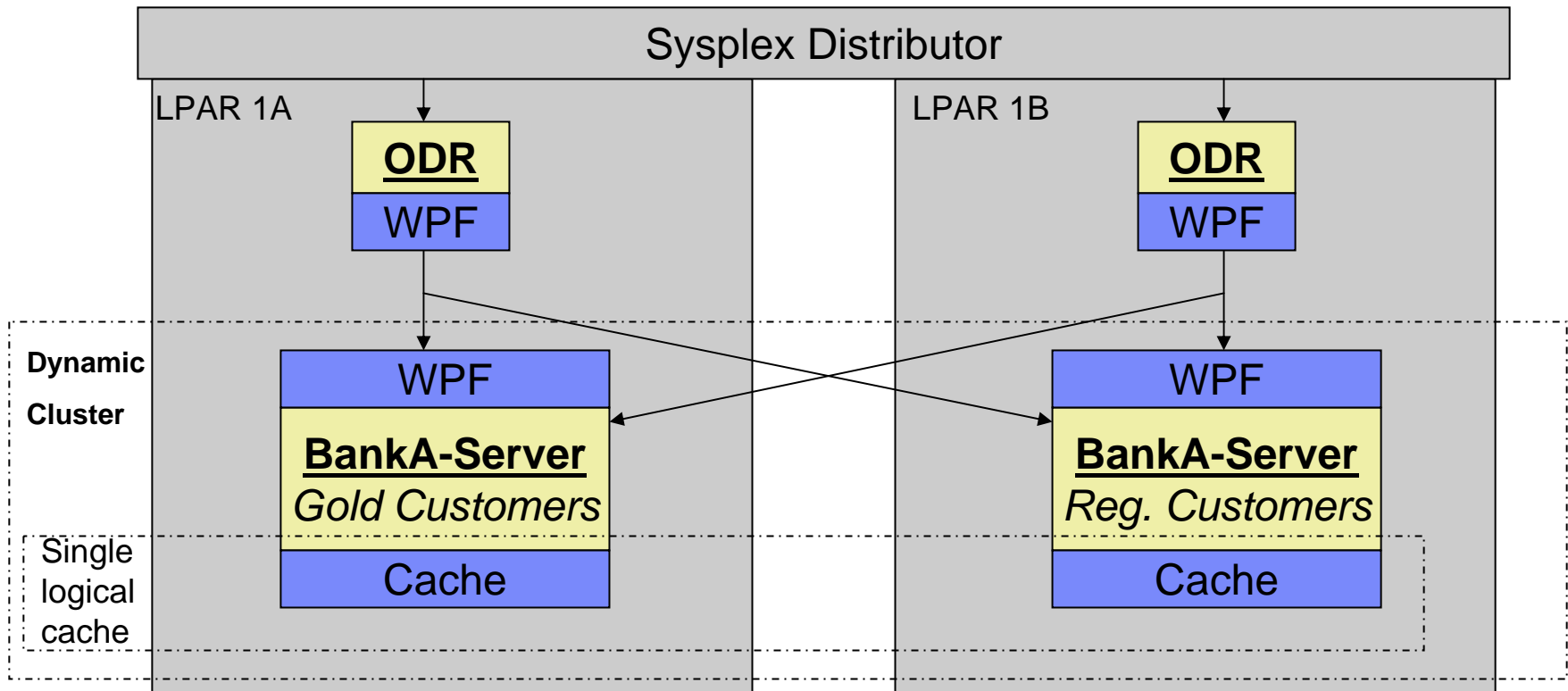
What is horizontal and vertical scaling?



# WAS XD z/OS- Example Topology

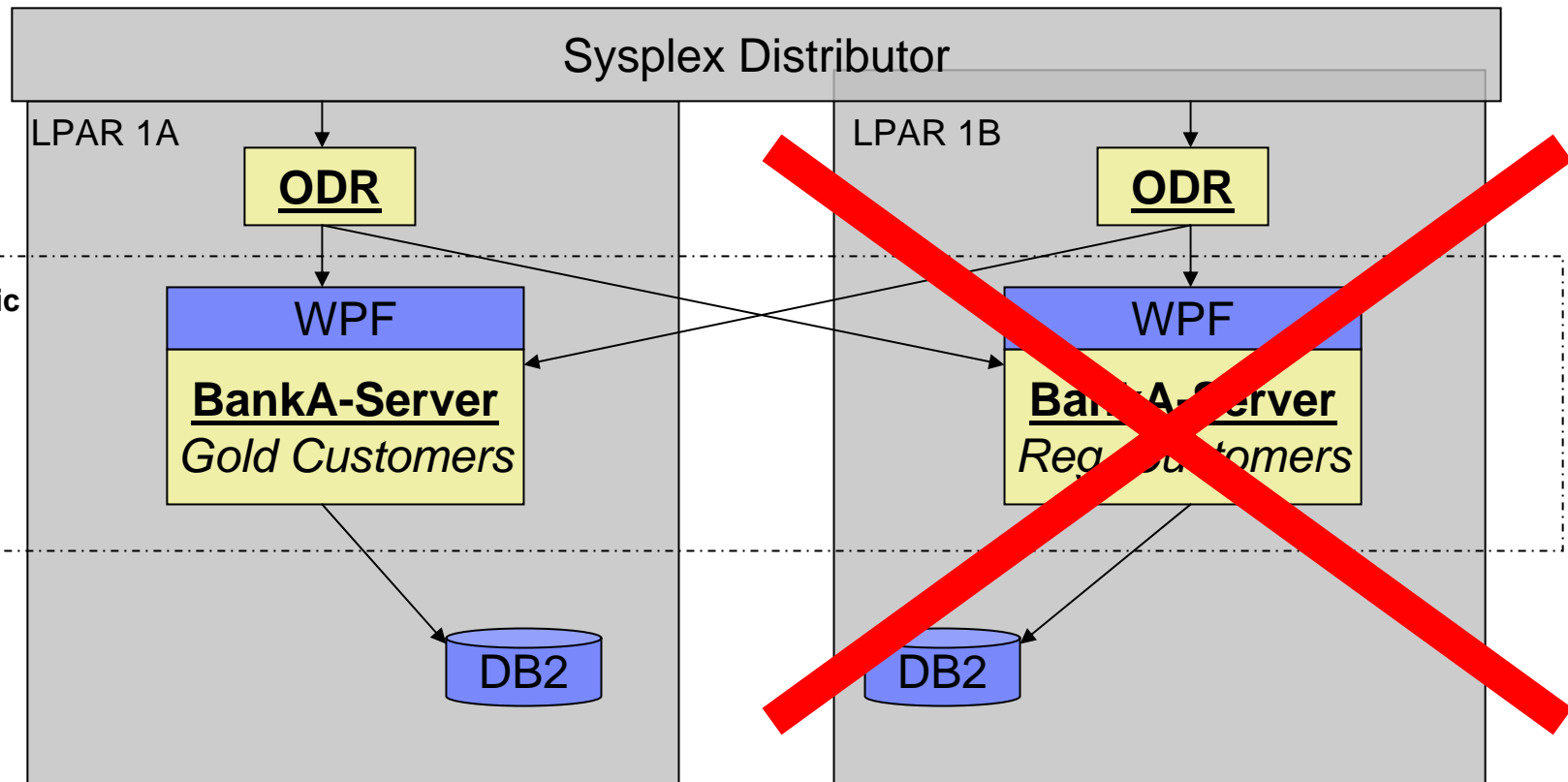
## Dynamic Clusters with 24x7 Batch





- With WPF, we establish affinity between the workload and the servers to which they are executed on. This affinity creates a single logical cache that spans both servers (*Cache entries will not be duplicated across the two servers*). The ODR in conjunction with WPF and the Sysplex Distributor handle failover just as failover is handled in today's clustered environment. We save in the following areas:

- *Cache-Misses are reduced since we have effectively doubled our cache size*
- *Data is not duplicated in memory among multiple servers*
- *near-linear scalability is possible as customer accounts grow and more servers are required.*



In case of failure, two mechanisms within this environment could detect and take action.

- Sysplex Distributor will not route work to a failed LPAR (LPAR1B), instead work is routed to LPAR 1A.
- ODR for LPAR1A will detect that the server on LPAR 1B has failed. WPF routing will temporarily be suspended and all work will be routed to the server on LPAR 1A.

*Note: Since we qualify requests with a TCLASS, WLM will attempt to maintain the stated QoS for gold customers and regular customers.*

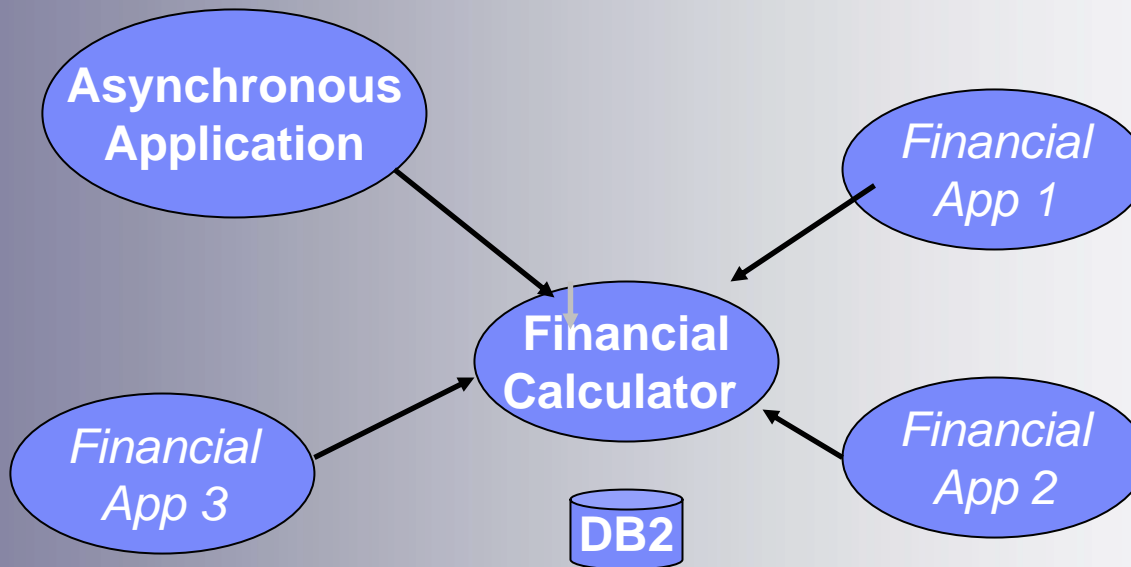


# WebSphere XD z/OS- Business Grid summary

- The XD Batch Container supports J2EE-based Batch Applications.
- These applications allow for asynchronous access to enterprise applications hosted in WebSphere. They have available to them WebSphere resources:
  - Transactions
  - Security
  - high availability
  - and so on
- The XD Batch Container provides services such as
  - Check Pointing- the ability to resume batch work at some selected interval
  - Result Processing- the ability to intercept step and job return codes and subsequently process them using any J2EE facility (Web Service, JMS message, and so on)
  - Batch Data Stream Management- the ability to handle reading, positioning, and repositioning data streams to files, relational databases, native z/OS datasets, and many other input and output sources.



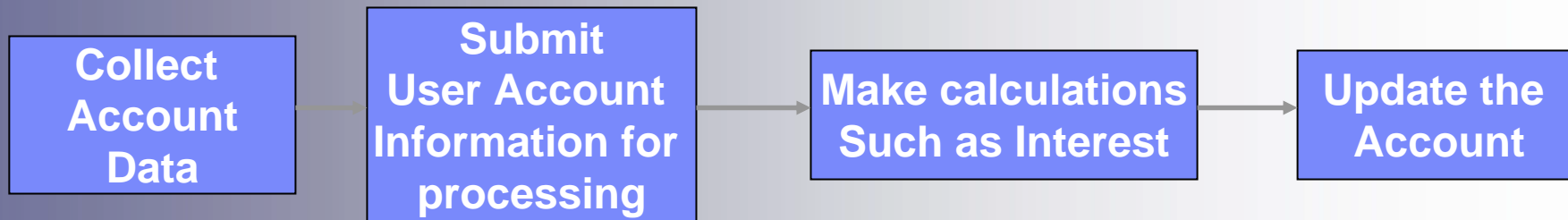
# Financial Business Application



The Financial Calculator application is a 'Kernel' application that must be available to numerous other banking applications, including asynchronous batch-type applications that execute tasks such as calculating interest and credit scores

# The Business Problem

An example batch task is to calculate interest for each account, the flow is as follows:

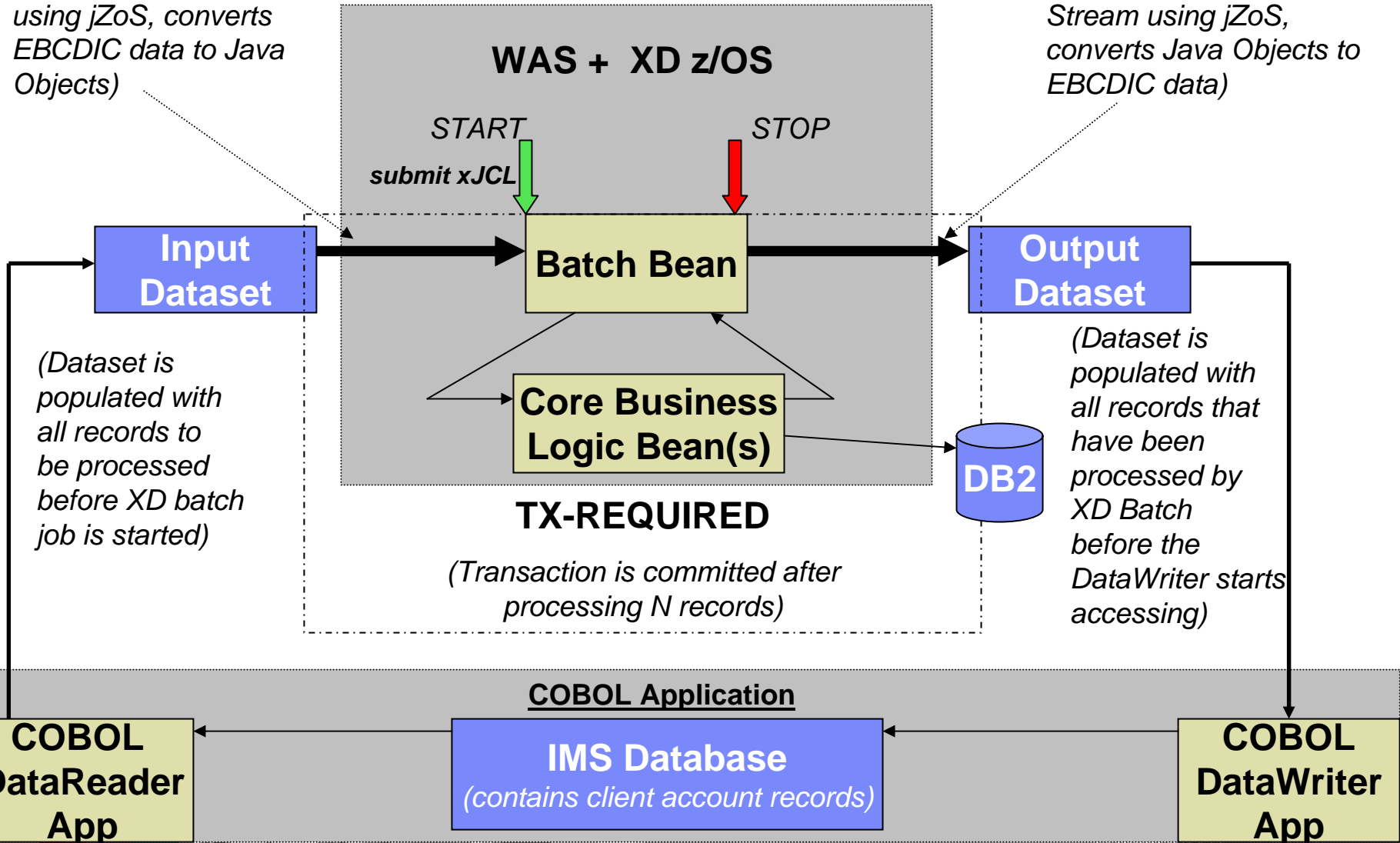


- How do we achieve this while ensuring data integrity for each account?
- How can we build an architecture that allows us to only have a single set of calculation applications which can be shared by any other business applications in the system?

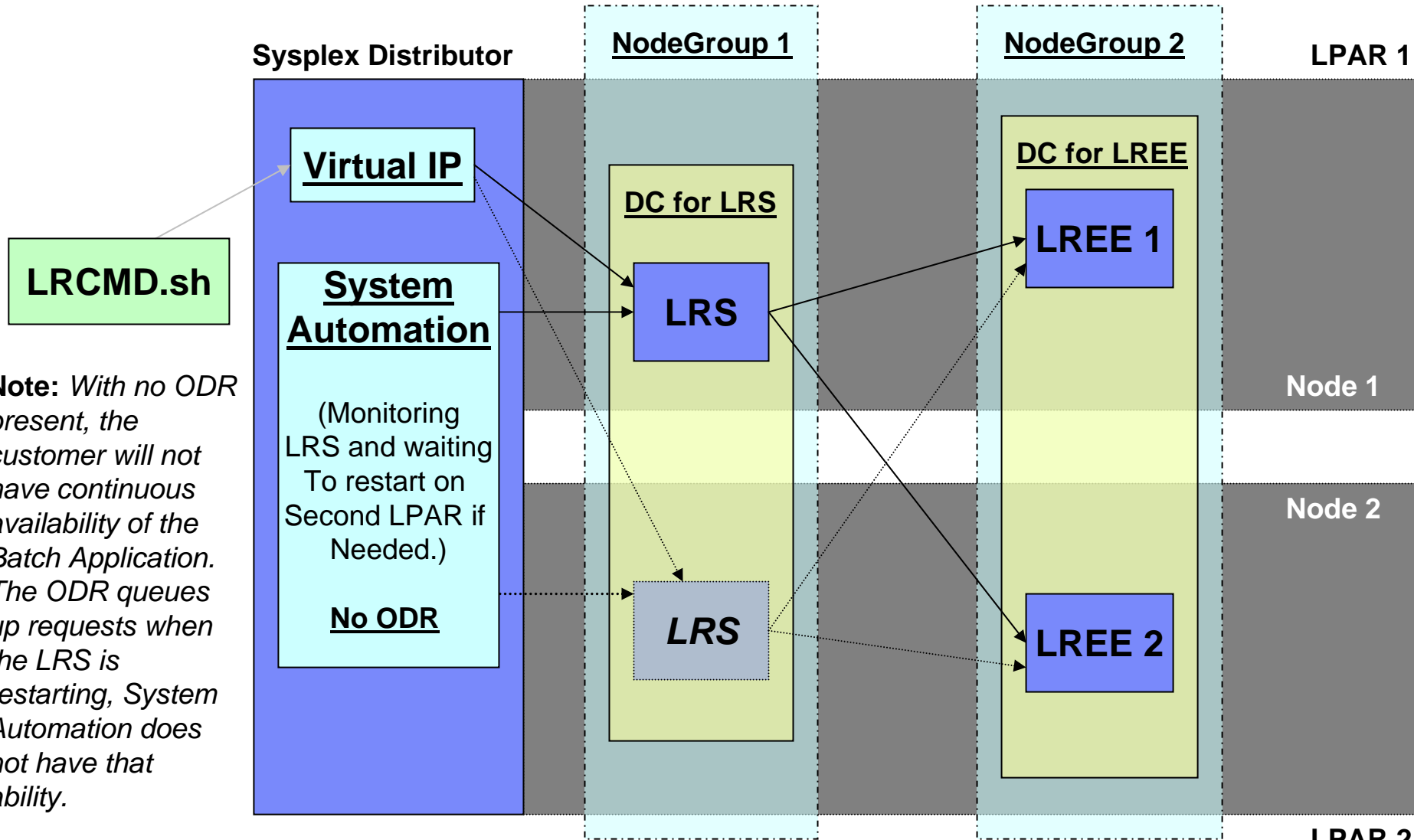
# Business Problem solved with XD

*(Input Batch Data Stream using jZoS, converts EBCDIC data to Java Objects)*

*(Output Batch Data Stream using jZoS, converts Java Objects to EBCDIC data)*



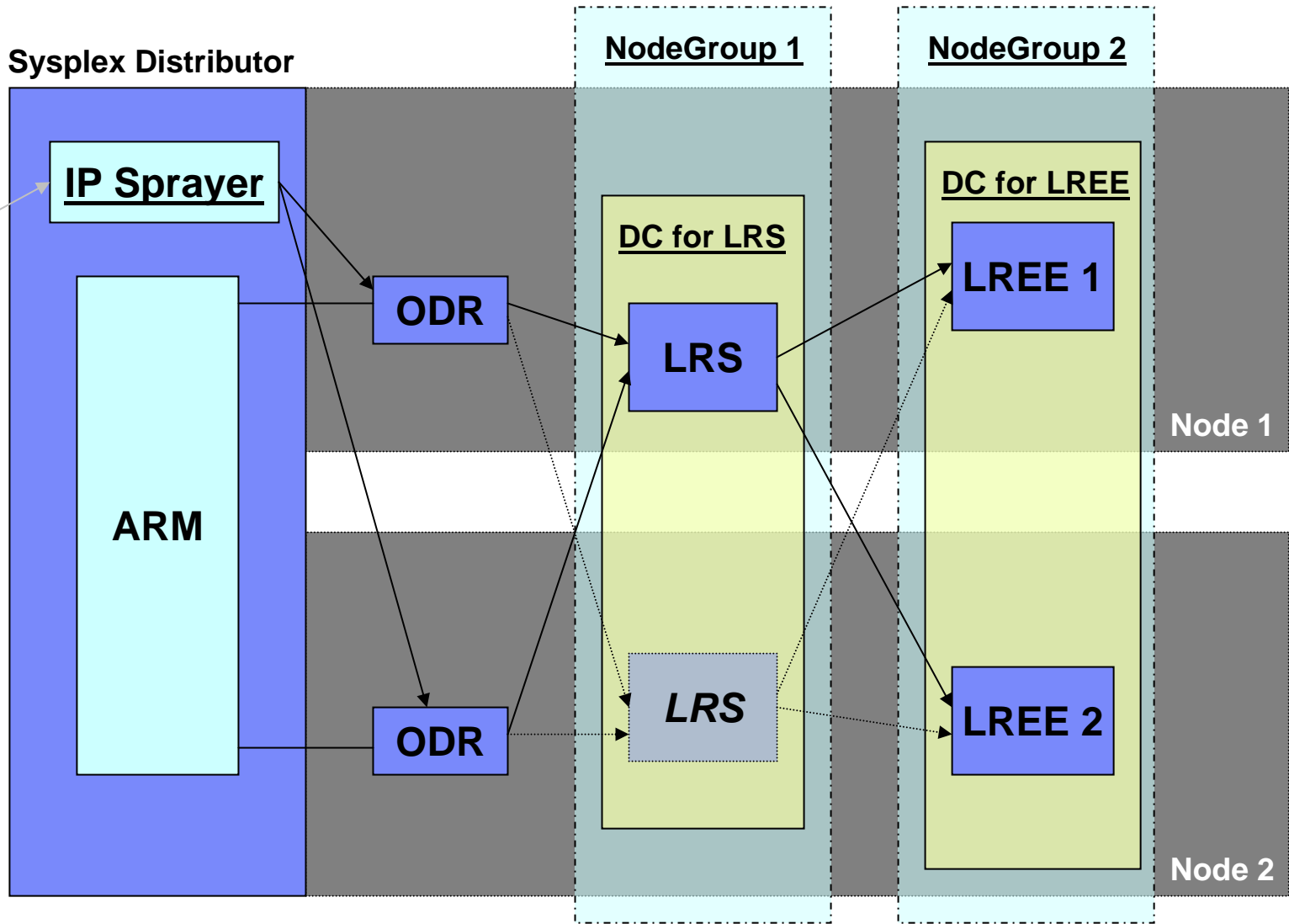
# Customer WebSphere XD for z/OS Topology



**Note:** With no ODR present, the customer will not have continuous availability of the Batch Application. The ODR queues up requests when the LRS is restarting, System Automation does not have that ability.



# Enabling continuous availability of the LREE



**LRCMD.sh**

**Note:** With no ODR present, the customer will not have continuous availability of the Batch Application. The ODR queues up requests when the LRS is restarting, System Automation does not have that ability.

# A bright and rising future for CICS Transaction Server

*Rock-solid deployment platform, 100% aligned with SOA technologies*

## **CICS Transaction Server V3.1**

- Build Web services with no change to existing applications
- Highest levels of data integrity and security
- Optimized throughput and performance
- Simple and intuitive management
- New SOA Statement of Direction
  - ▶ Integration of the CICS Service Flow Runtime with WebSphere Process Server
  - ▶ CICS integration with WebSphere Service Registry/Repository
  - ▶ Support for Event Driven Applications



# CICS Web Services

*“By some estimates, the total value of the applications residing on mainframes today exceeds \$1 trillion.” - ComputerWorld April 24, 2006*

- Web services capabilities enable CICS assets to directly participate in an SOA
  - ✓ Both as a Web service provider and requester
- Rich set of Web services standards supported
  - ✓ SOAP 1.1 and 1.2 to send and receive Web services messages
  - ✓ WS-I Basic Profile 1.1 for SOAP interoperability
  - ✓ WS-Coordination extensible coordination framework
  - ✓ WS-AtomicTransaction for transaction coordination
  - ✓ WS-Security for authentication and encryption of all or part of a message. SOAP Message Security, Username Token Profile 1.0, X.509 Certificate Token
- Both HTTP and WebSphere MQ network layers supported
  - ✓ For flexible deployment options dependant on application and IT requirements
  - ✓ CICS assets are agnostic to the transport mechanism used



## Additional information

- IBM Red Books:
  - ▶ “WebSphere for z/OS Connectivity Architectural Choices”
  - ▶ “Revealed! Architecting e-business Access to CICS”
  - ▶ “Architecting High Availability Using WebSphere V6 on z/OS”
  - ▶ “WebSphere for z/OS Extended Deployment XD - Building Basic Infrastructure”
  
- Developer Works Articles
  - ▶ “Solving Business Problems with WebSphere XD- Using XD's Long Running Execution Model to asynchronously access enterprise applications deployed in WebSphere” *Coming Soon*

Thank  
YOU

