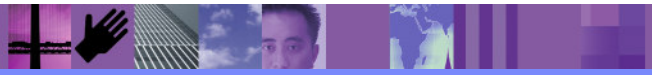
Virtualization Aspects of Web Application Servers

Exemplified by WebSphere Virtual Enterprise

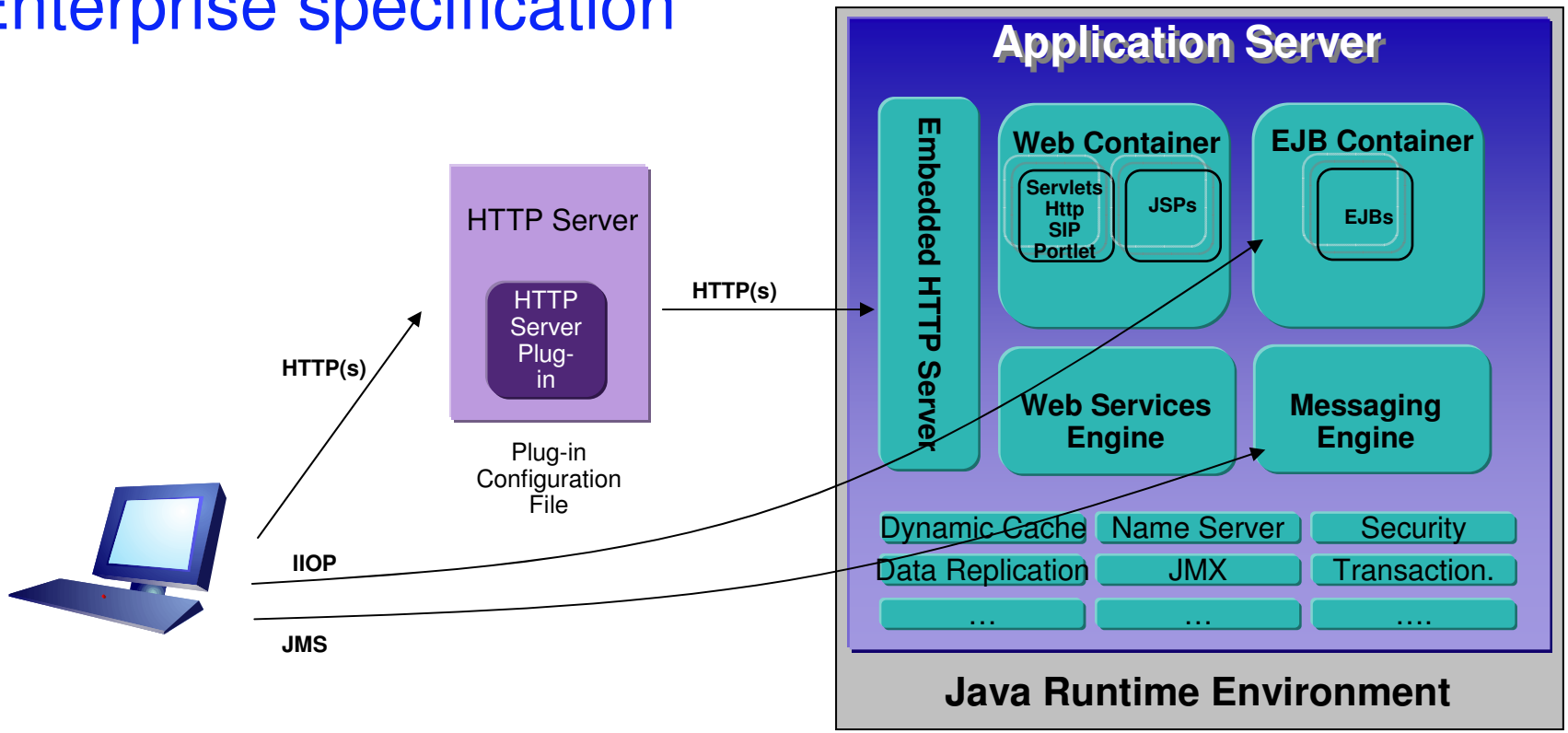
Reinhard Hohberger
IBM Software Group
WebSphere IT Specialist
reinhard.hohberger@de.ibm.com

Agenda

- **General Virtualization Aspects of Web Application Servers**
- **Virtualization Options with the IBM Product "WebSphere Virtual Enterprise"**
- **Limitations and Issues**

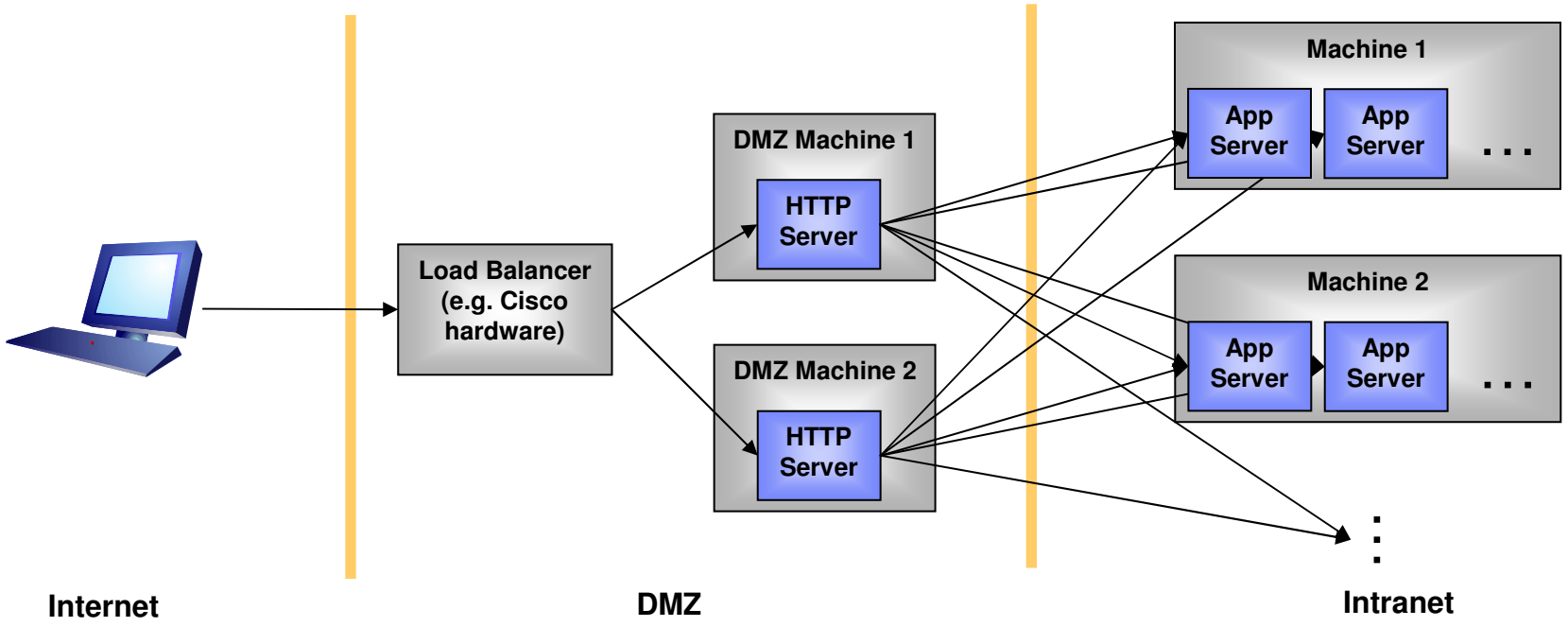


The standard architecture is derived from Java Enterprise specification



- This presentation concentrates on Java Enterprise architectures.
- Runtime containers, protocols and other architectural components are mostly standardized by the Java Enterprise specification from Sun Microsystems Inc.
- Protocols are important for virtualization options

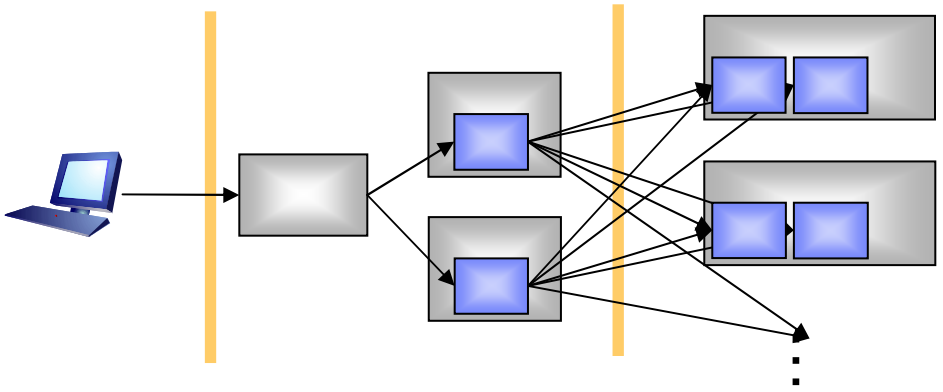
For a production environment one AppServer instance is not enough



- For production systems (with scalability, failover, etc. requirements) you need more than one AppServer instance. Often you have multiple instances running on the same machine.
- Similar configured AppServer instances running the same application(s) may be called “server groups” or “clusters” (depending on AppServer provider)

AppServer clusters provide several virtualization functions

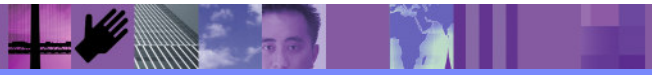
- AppServers can be installed on top of other virtualization technologies: VMWare, Xen, LPARs (AIX), Zones (Solaris), …
- Administration is done on cluster level not on single server:
  - start/stop cluster
  - add/remove a server / machine
  - deploy/undeploy application on cluster



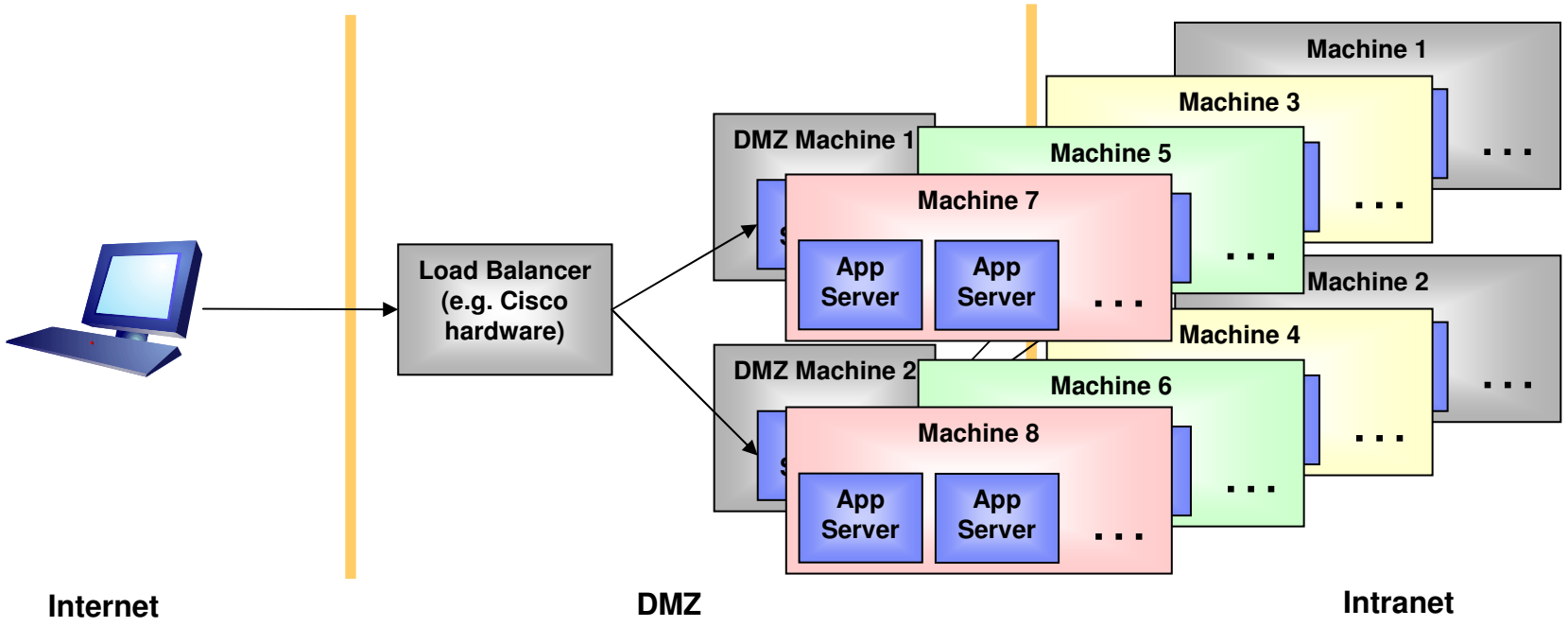
- AppServer cluster provides load balancing, failover, etc.
- AppServer cluster could span multiple machines, different operating systems, sometimes different minor versions (different patch levels)
- An HTTP client does not see the complexity of the AppServer environment, because the HTTP server forwards requests to AppServers. EJB and JMS clients need to contact the AppServer directly.

Agenda

- **General Virtualization Aspects of Web Application Servers**
- **Virtualization Options with the IBM Product "WebSphere Virtual Enterprise"**
- **Limitations and Issues**



Applications often do not share AppServers

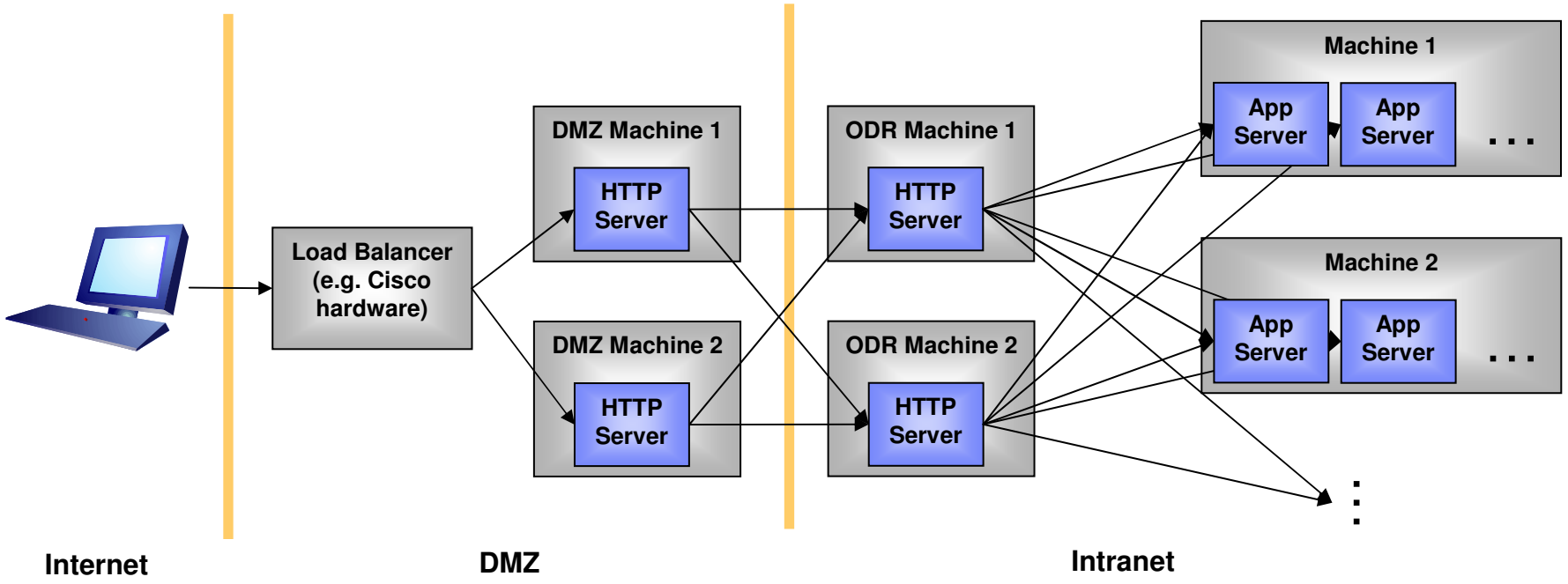


- One Application Server could host multiple applications, but often this is not the case. Many times different applications do not even share machines.
- Result:
  - a huge number of machines
  - most of them running with very low utilization most of the time

The idea is to put all machines into one big resource pool that can be used by all applications

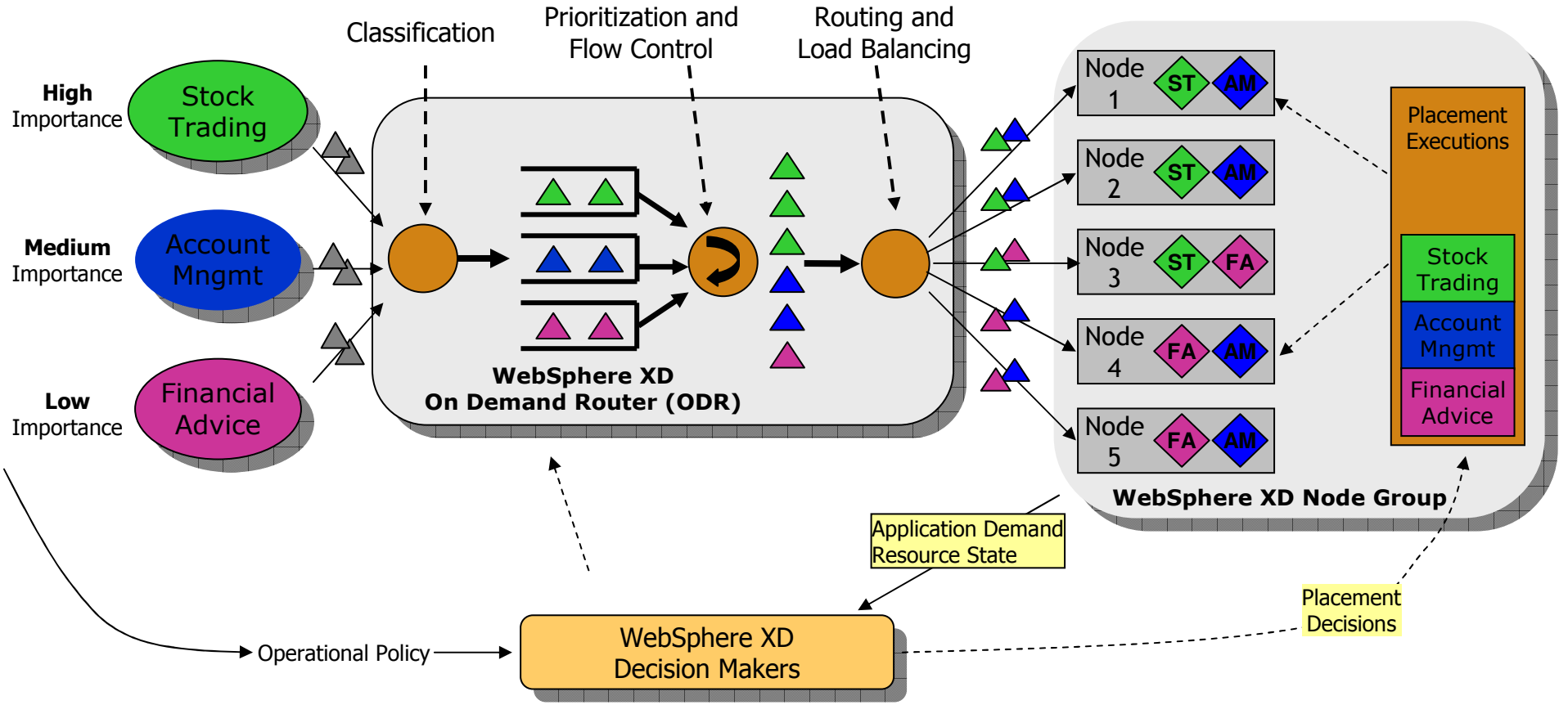
- Allow applications to share the same resources: cpu and memory of all available machines. Need to specifiy on which machines an application might be deployed. This leads to
  - Dynamic Clusters – possible deployment targets (AppServers) for an application
- Define rules that specify how many resources an application actually needs and establish mechanisms to provide these resources dynamically. If the resources are no longer needed, other applications can use them. This leads to
  - Classification of requests and Service Policies – how many resources are needed
  - Automatic Placement of applications – how many resources are provides
- Define rules for the situation that there are not enough resources for all applications. This leads to priorization of requests.
- For theses tasks some new components have to be introduced into the overall architecture

An On Demand Router (ODR) tier is introduced into the architecture



- The On Demand Router (ODR) is a Java-based HTTP Proxy Server.
- The ODR sits between the HTTP Server and the AppServer. It is not in the DMZ because it is a Java based server (-> security reasons).
- The ODR controls the HTTP(s) traffic to the AppServers
- For IIOP traffic the ORB in the JRE is doing this work

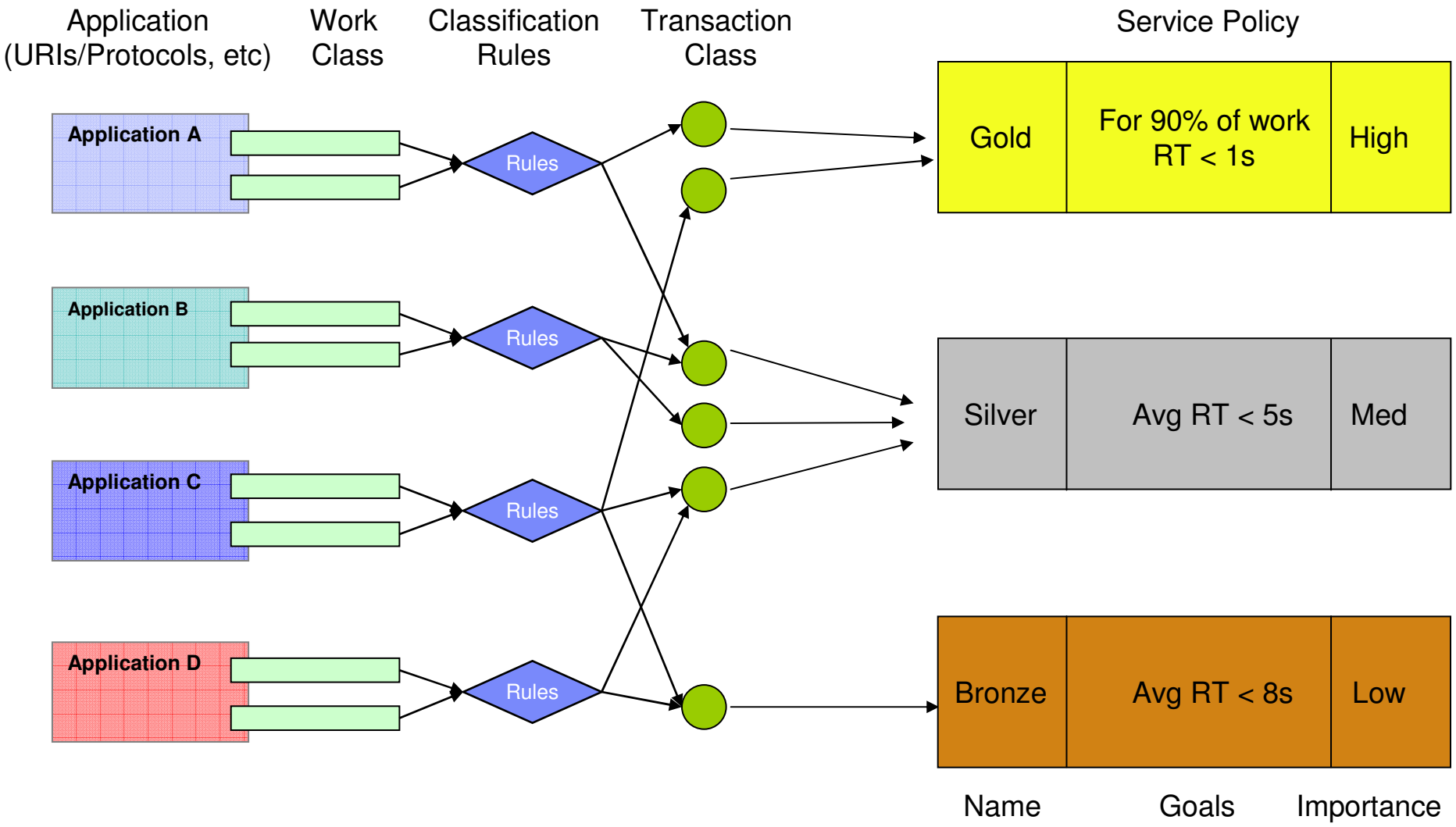
The IBM product "WebSphere Virtual Enterprise" provides advanced routing via an ODR



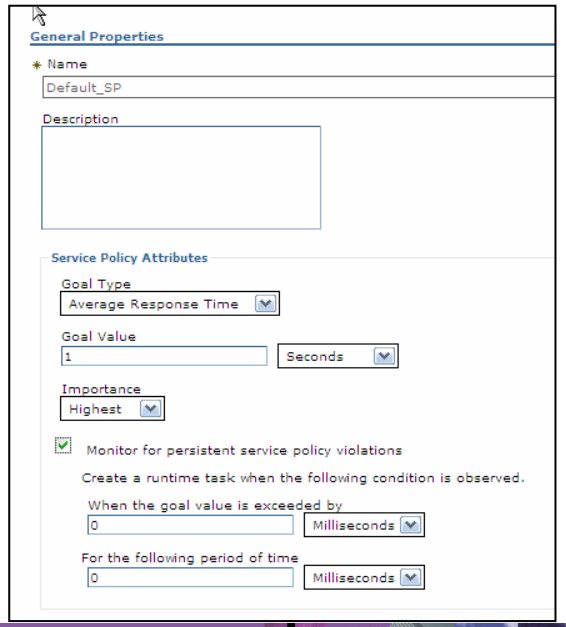
A Dynamic Cluster (DC) is the deployment boundary for an application

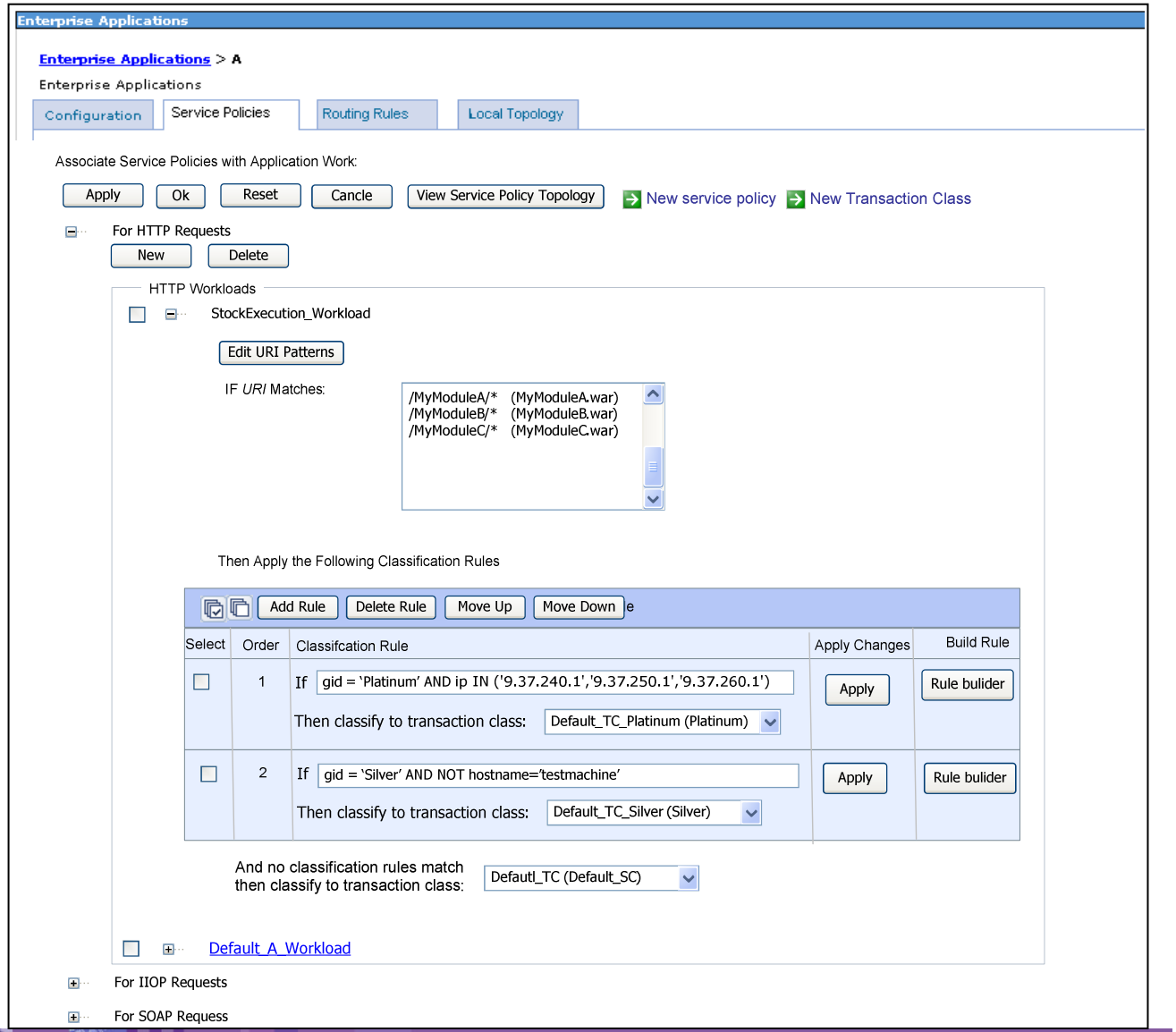
- A dynamic group of AppServers to which applications can be deployed.
- Similar to a static cluster, but number of active AppServers can be resized dynamically at run-time
- Dynamic cluster server types
  - WebSphere Application Server
  - PHP
  - Apache Tomcat
  - Apache HTTP Server
  - JBoss Application Server
  - WebSphere Application Server Community Edition
  - BEA WebLogic Server
  - Custom HTTP servers

Classification defines the mapping of requests to Service Policies



Definition of Service Policies



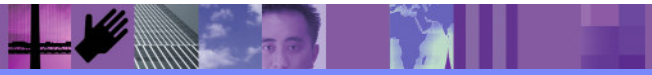


This architecture allows more sophisticated management functions

- Isolation policies of applications
- Minimum/Maximum number of servers for an application
- Health Policies: Proactively deal with application and application infrastructure issues before they become acute problems … automatically:
  - Health policies can be defined for common server health conditions
  - Health conditions are monitored and corrective actions taken automatically
    - Notify administrator
    - Capture diagnostics
    - Restart server
  - Health Conditions
    - Age-based: amount of time server has been running
    - Excessive requests: % of timed out requests
    - Excessive response time: average response time
    - Excessive memory: % of maximum JVM heap size
    - Memory leak: JVM heap size after garbage collection
    - Storm drain: significant drop in response time
    - Workload: total number of requests
- Custom health conditions and actions can be defined

Agenda

- **General Virtualization Aspects of Web Application Servers**
- **Virtualization Options with the IBM Product "WebSphere Virtual Enterprise"**
- **Limitations and Issues**

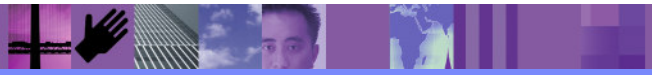


Technical issues

- Works best, if applications have the different resource utilization patterns: peak loads at different times of the day.
- Helps, if applications are compute or memory bound. If I/O or other JEE resources (e.g. database) are the bottleneck, then dynamic placement will not help.
- To be able to define Service Level Agreements (SLAs) a good understanding of the application performance is necessary. Performance tests and optimization are mandatory.
- Management and monitoring of the complete environment is still complex and not integrated into standard tooling.

Organizational issues

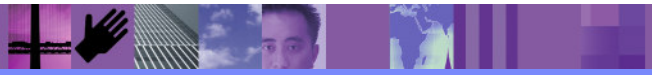
- Is the application owner willing to share the application server environment with others?
- Is it possible to define important and less important applications? A payment system might help to think about importance of applications.
- Is there enough trust into the automatic management functions of the system?



General Remark

There is no silver bullet. A bad application will not become better, if it is running in a virtualized environment.

It is much more likely that there will be a negative impact on all the other applications that share the same resources.



धन्यवाद
Hindi

多謝
Traditional Chinese

ขอบคุณ
Thai

Спасибо
Russian

Gracias
Spanish

Thank You
English

شكرا
Arabic

Merci
French

Obrigado
Brazilian Portuguese

Grazie
Italian

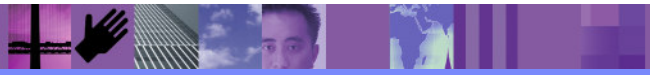
多谢
Simplified Chinese

Danke
German

நன்றி
Tamil

ありがとうございました
Japanese

감사합니다



References and Further Reading

- IBM White Paper: “WebSphere Virtual Enterprise and Virtualization”:
  http://www.ibm.com/developerworks/wikis/download/attachments/73040328/virt_book.pdf?version=1
- IBM InfoCenter for “WebSphere Virtual Enterprise”:
  http://publib.boulder.ibm.com/infocenter/wxdinfo/v6r1/index.jsp
- IBM White Paper:”Extending virtualization in the data center with application infrastructure virtualization”
  ftp://ftp.software.ibm.com/software/webservers/appserv/WebSphere_Virtual_Enterprise_wp.pdf
-
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both

Virtualization Aspects of Web Application Servers

Exemplified by WebSphere Virtual Enterprise

Reinhard Hohberger
IBM Software Group
WebSphere IT Specialist
reinhard.hohberger@de.ibm.com