Introduction to Java Enterprise Edition

"Enterprises today need to extend their reach, reduce their costs, and lower their response times"

IT must provide:

- High quality services
  - Highly Available
  - Secure
  - Scalable
- Powerful services
  - Useful
  - Flexible
  - Extensible
- Affordable

Difficult Tasks!

Complex software is expensive to:

- **produce**
- **purchase**
- **support**
- **maintain**
- **replace**

What is Java EE Enterprise Edition?

"The Java EE platform, Enterprise Edition reduces the cost and complexity of developing … multi-tier services, resulting in services that can be rapidly deployed and easily enhanced"

– *Simplified Guide to the Java 2 Platform, Enterprise Edition, Sun Microsystems, Inc.*

Java EE Provides:

- Enabling technology
- Standards based application model
- A common architecture that provides key common functionality:
  - Security
  - Session Management
  - Scalability

What does "multi-tier" mean?

- Applications architecture
- Partitioning of application logic
- Improves maintainability
- Simplifies software upgrades and management

What does "multi-tier" mean?

- Enables increased aggregate functionality
- Reduces component maintenance cost
- Separation of workload to improve scalability and reliability
- Strong separation prevents vendor lock-in

Two tier application ("thin client")

client *server*

- **Easy to upgrade client**
- **Expensive to scale and to add functionality**

**Business Logic*

Two tier application ("client/server")

client *server*

- **Strong scalability**
- **Poor upgradability and difficult management**

Three tier application

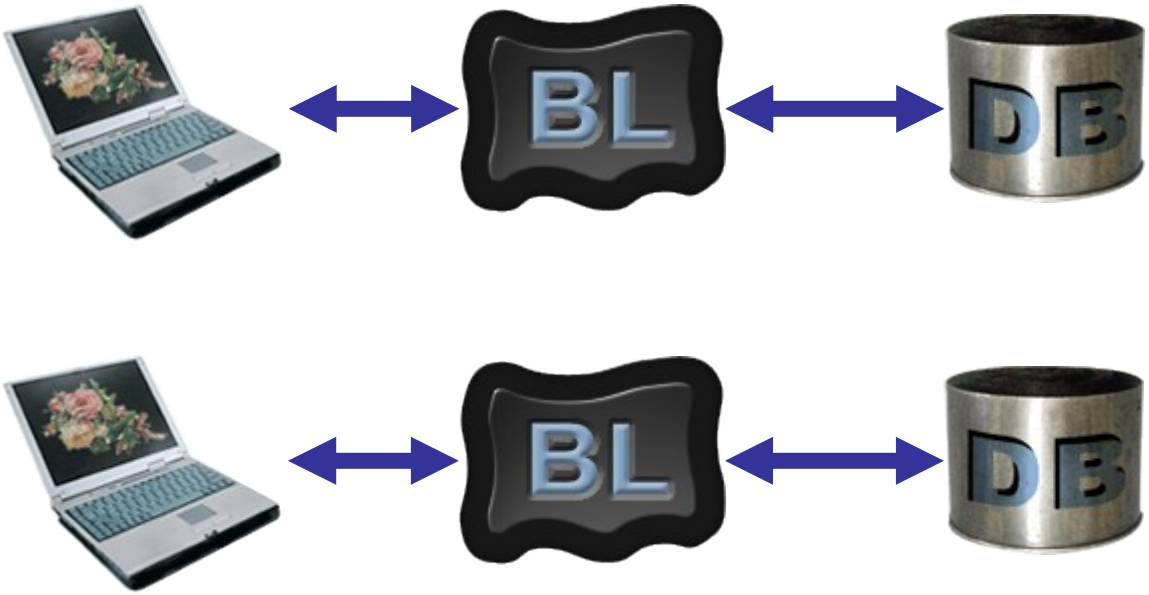
- Thin clients make easy upgrades
- Centralized application server management is easier to manage
- Model allows for horizontal scalability

Three tier application

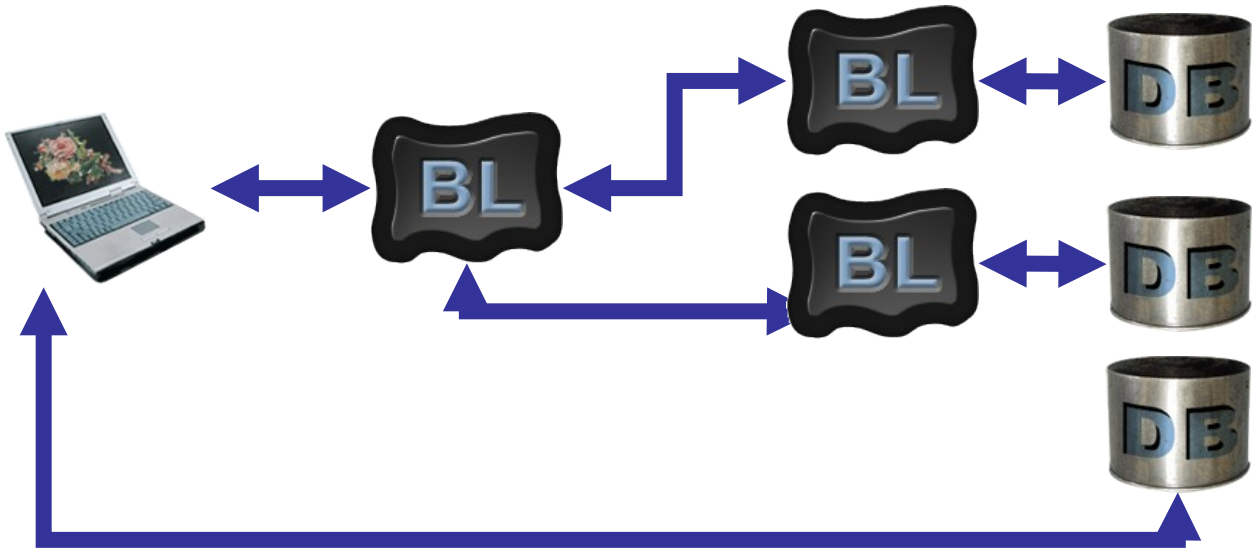
client *app server* *server*

- **Thin clients make easy upgrades**
- **Centralized application server management is easier to manage**
- **Model allows for horizontal scalability**

Today

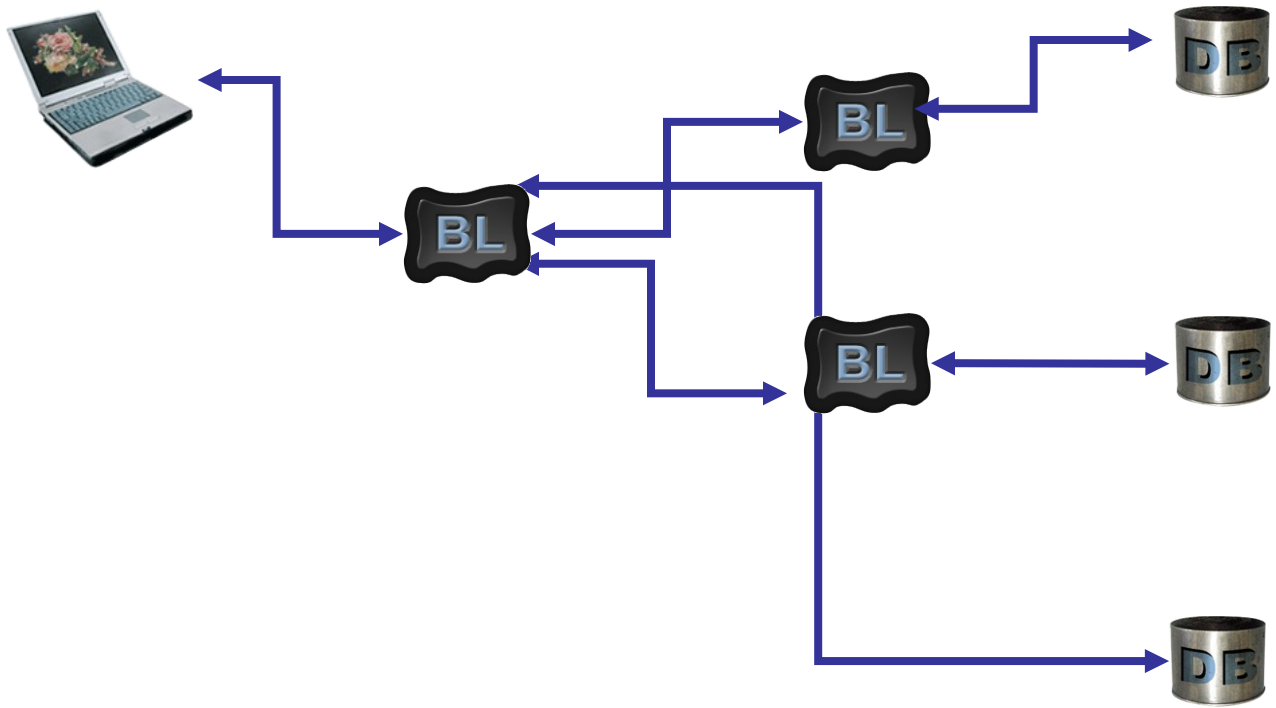


N-tier application

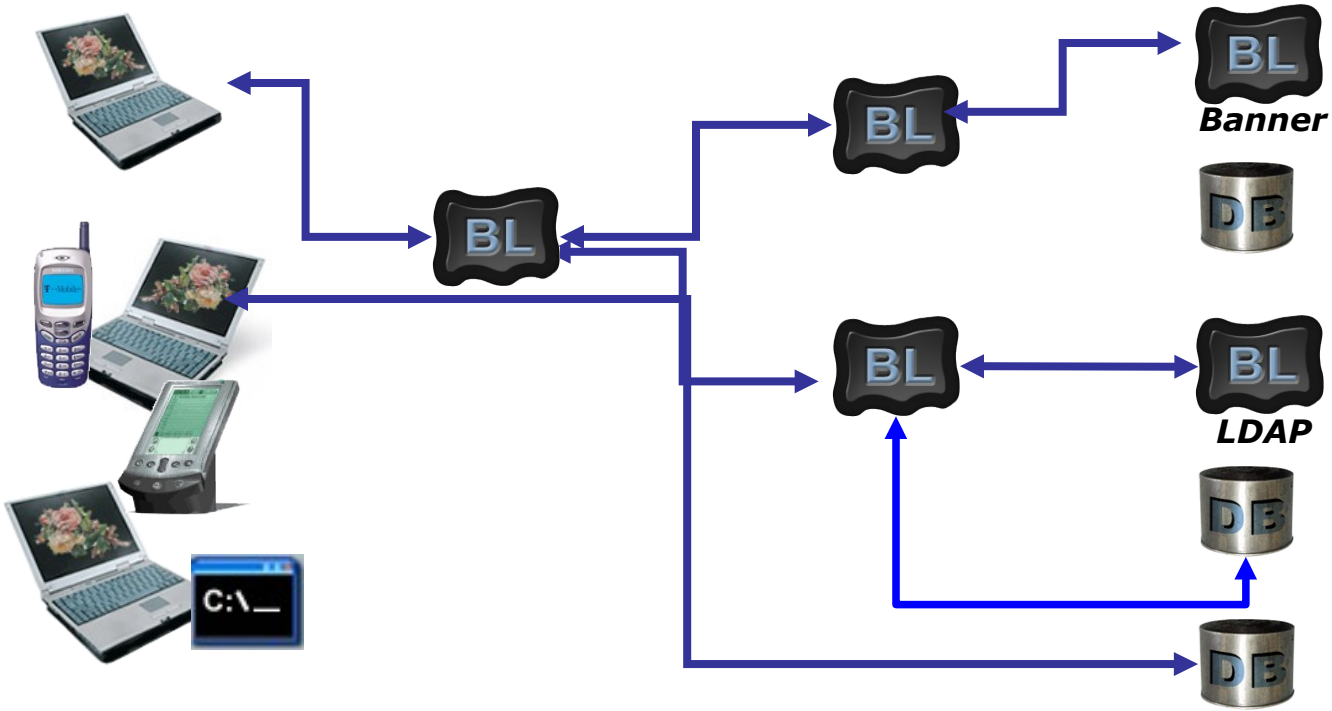


- View application server as a data store
- Leverage data abstraction

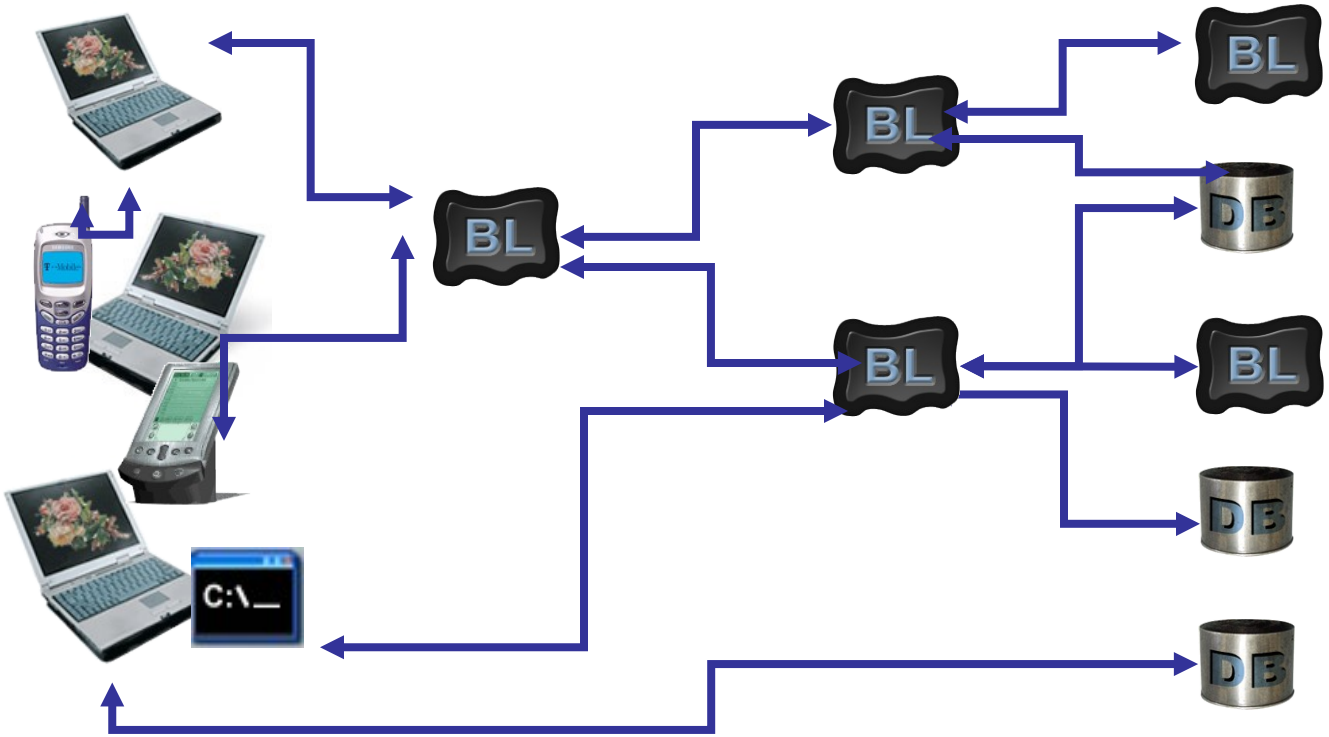
N-Tier Application



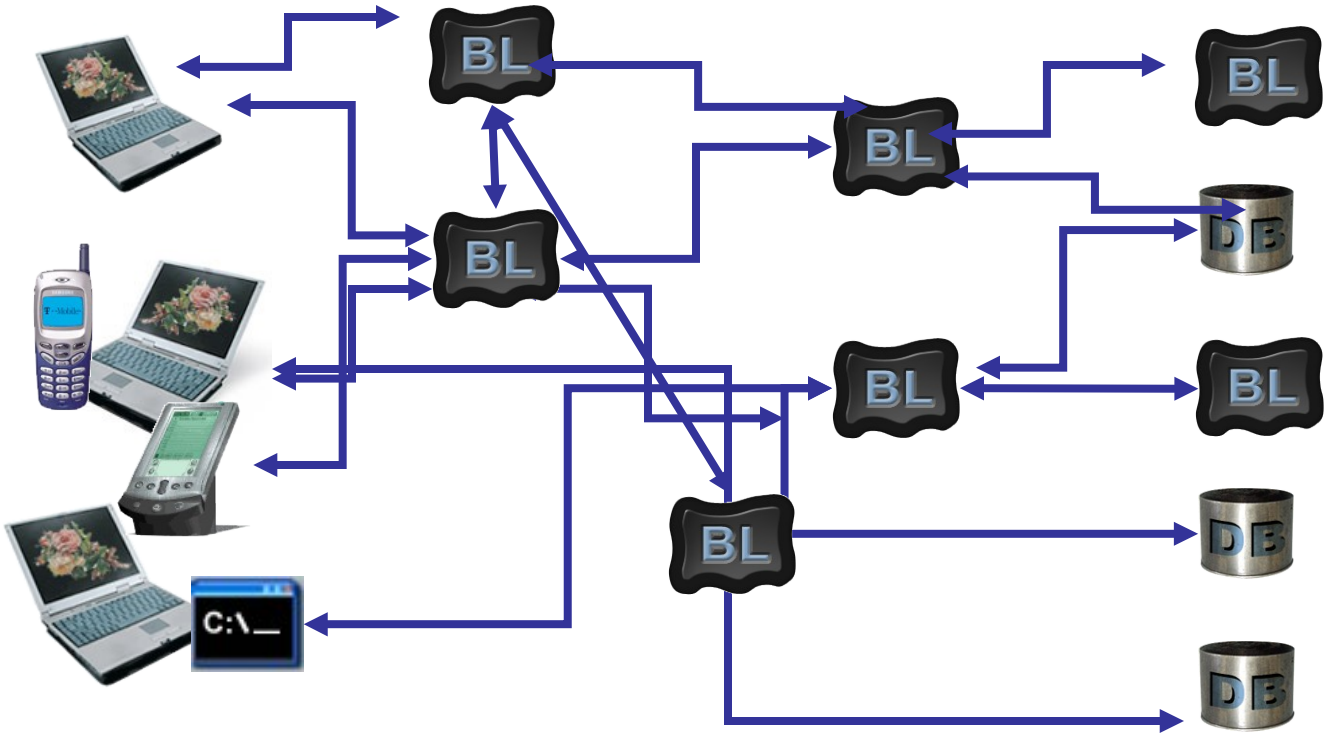
N-Tier Application



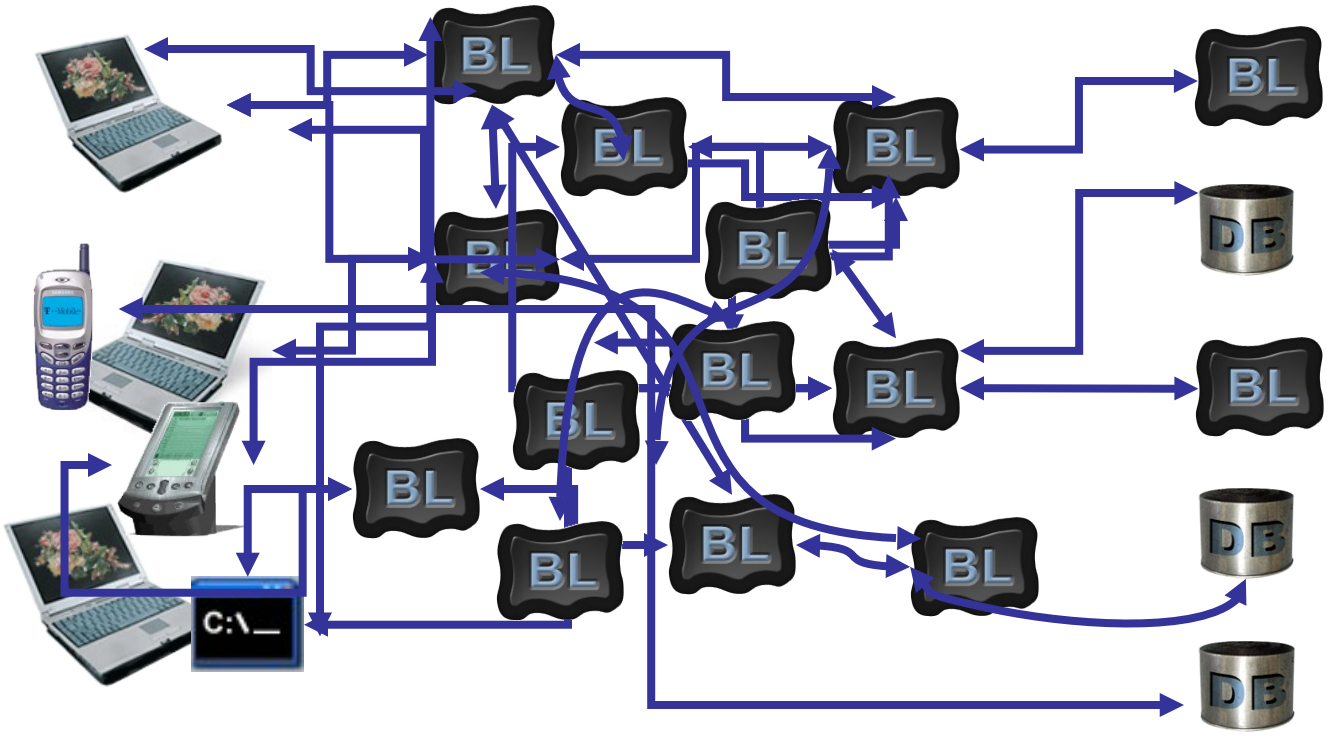
N-Tier Application



N-Tier Application

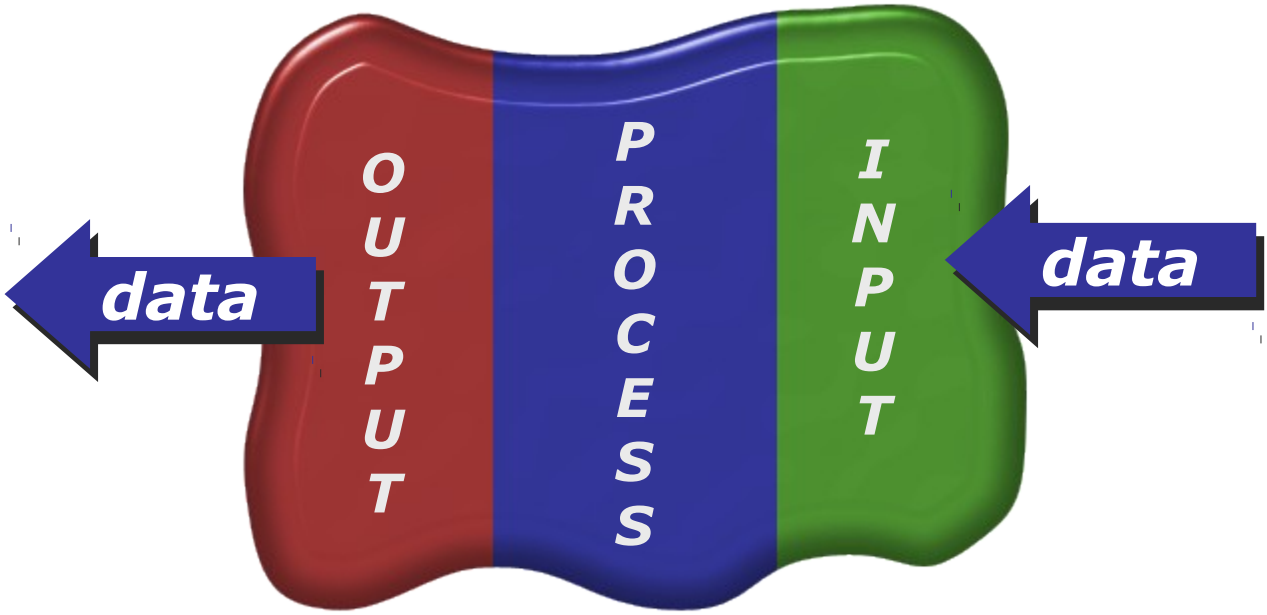


N-Tier Complexity

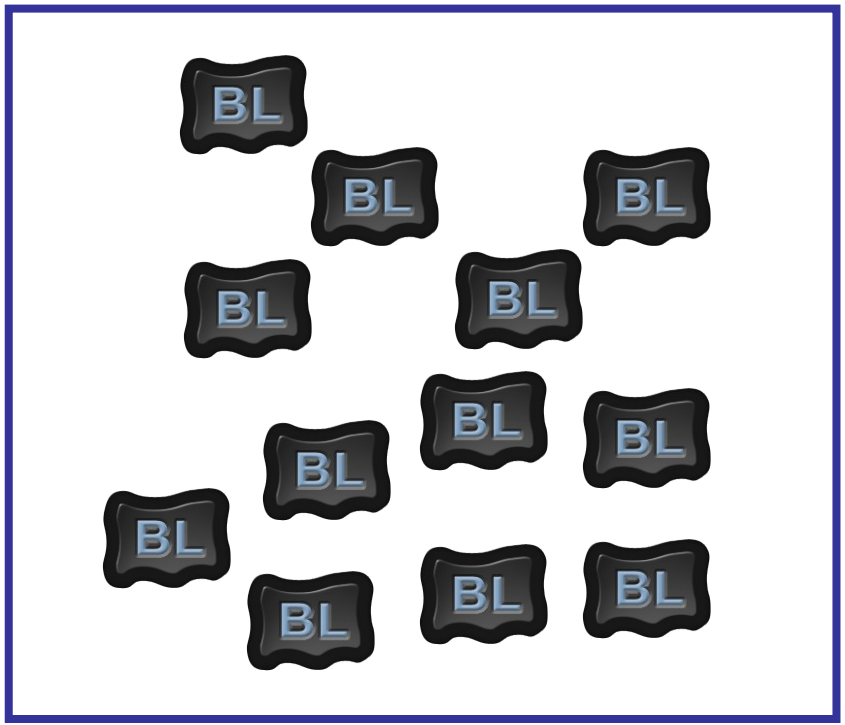



BL

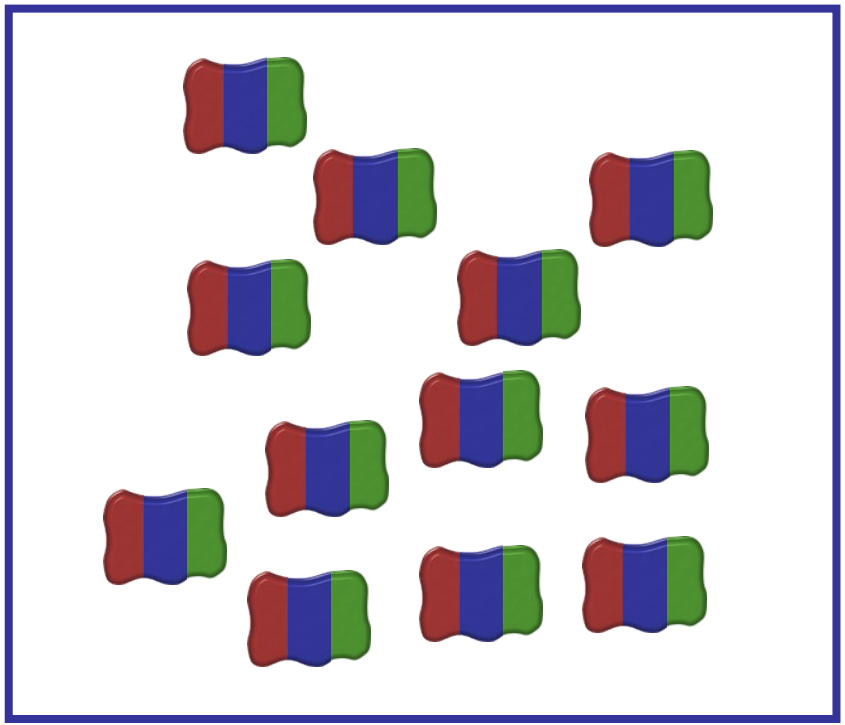
Business Logic



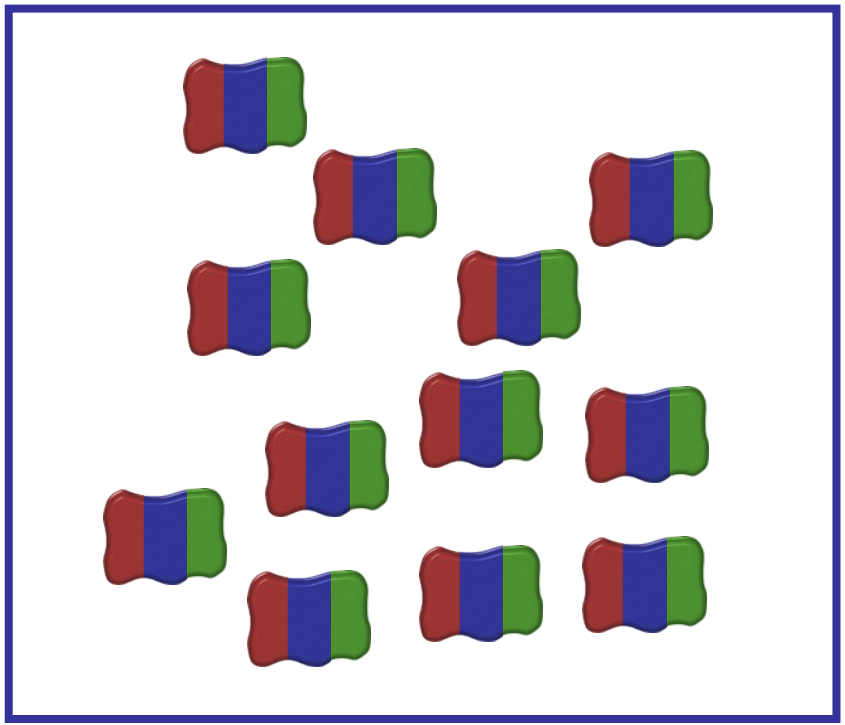
N-Tier Complexity



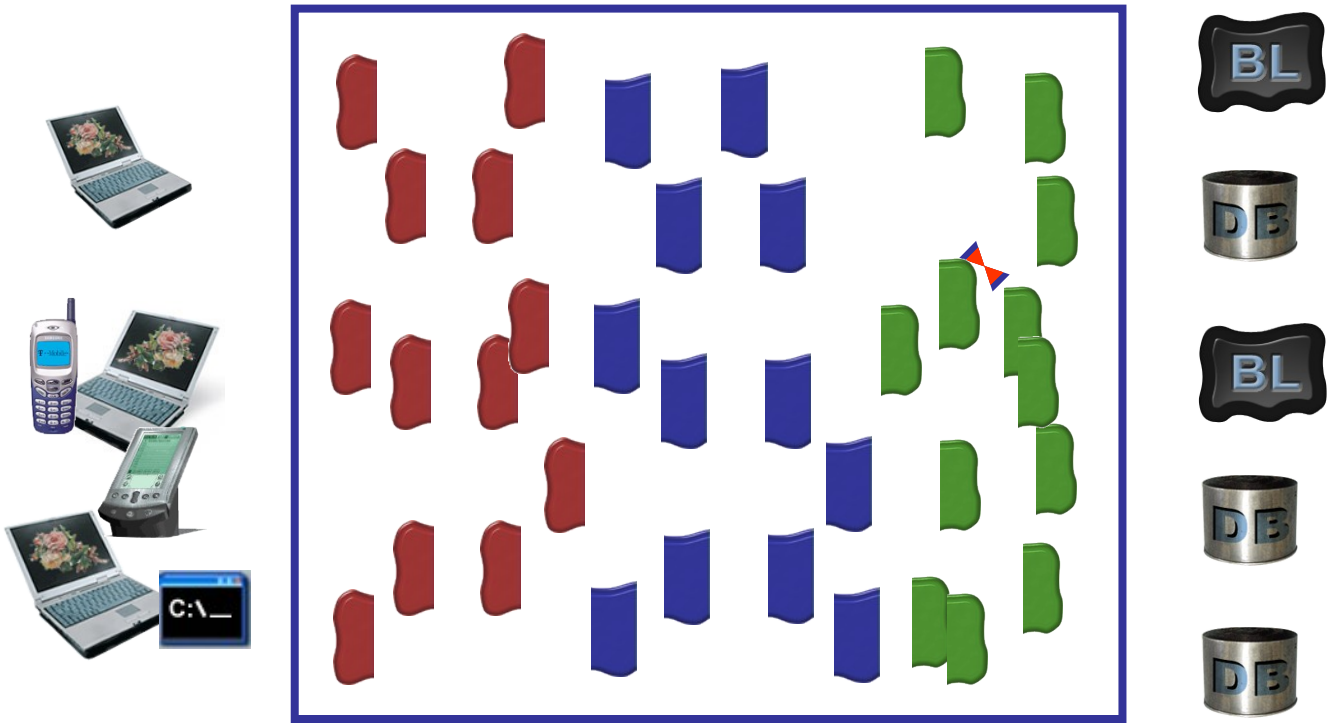
N-Tier Complexity



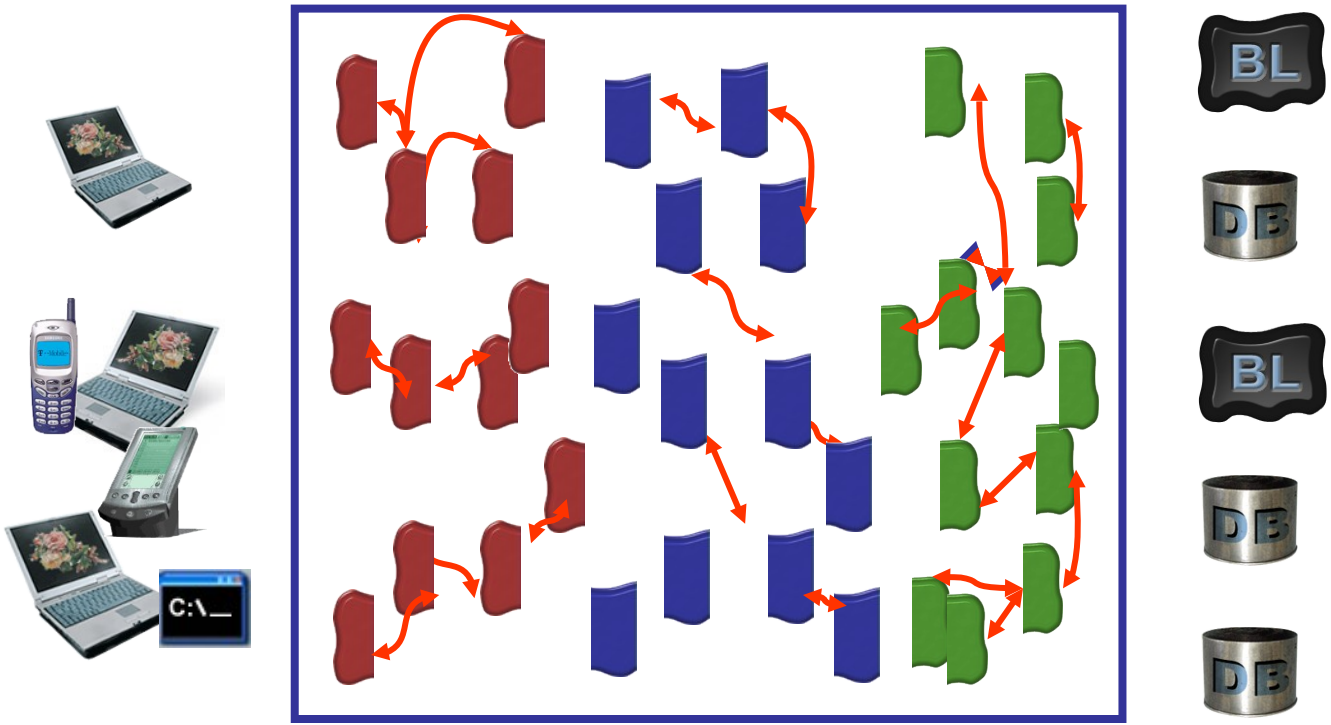
N-Tier Complexity



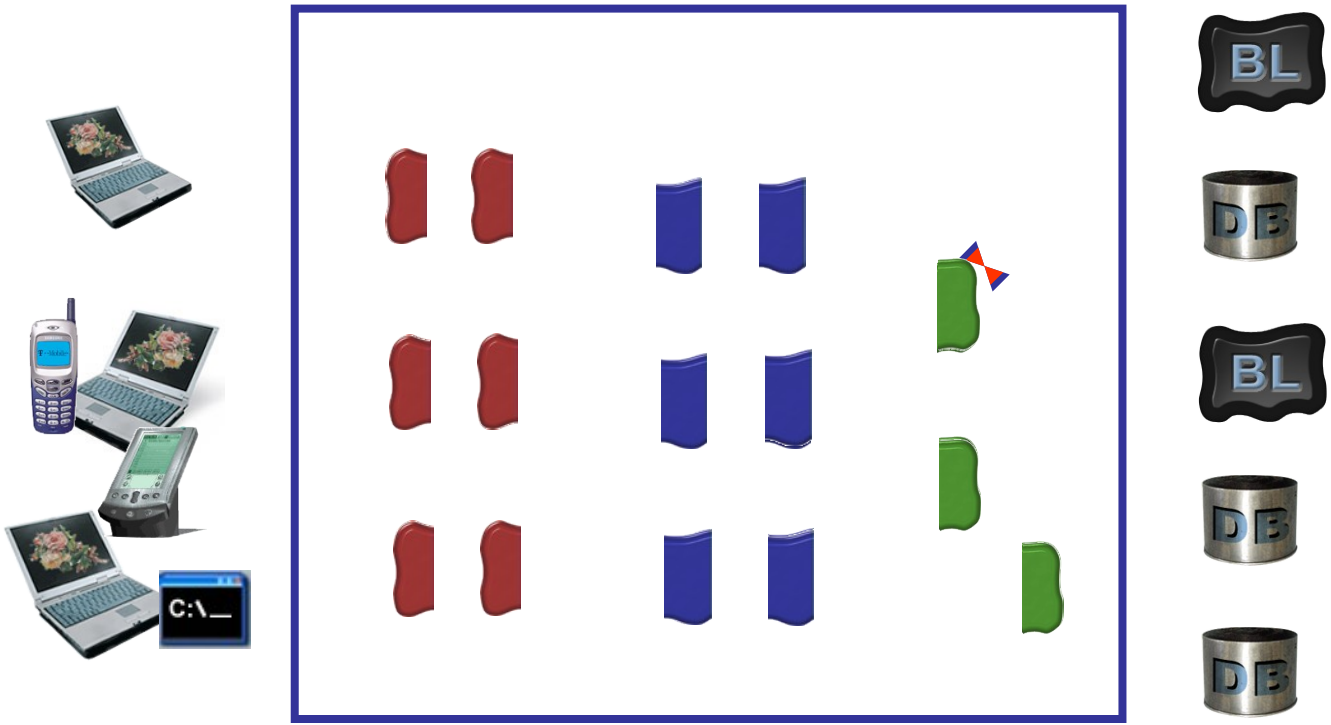
N-Tier Complexity

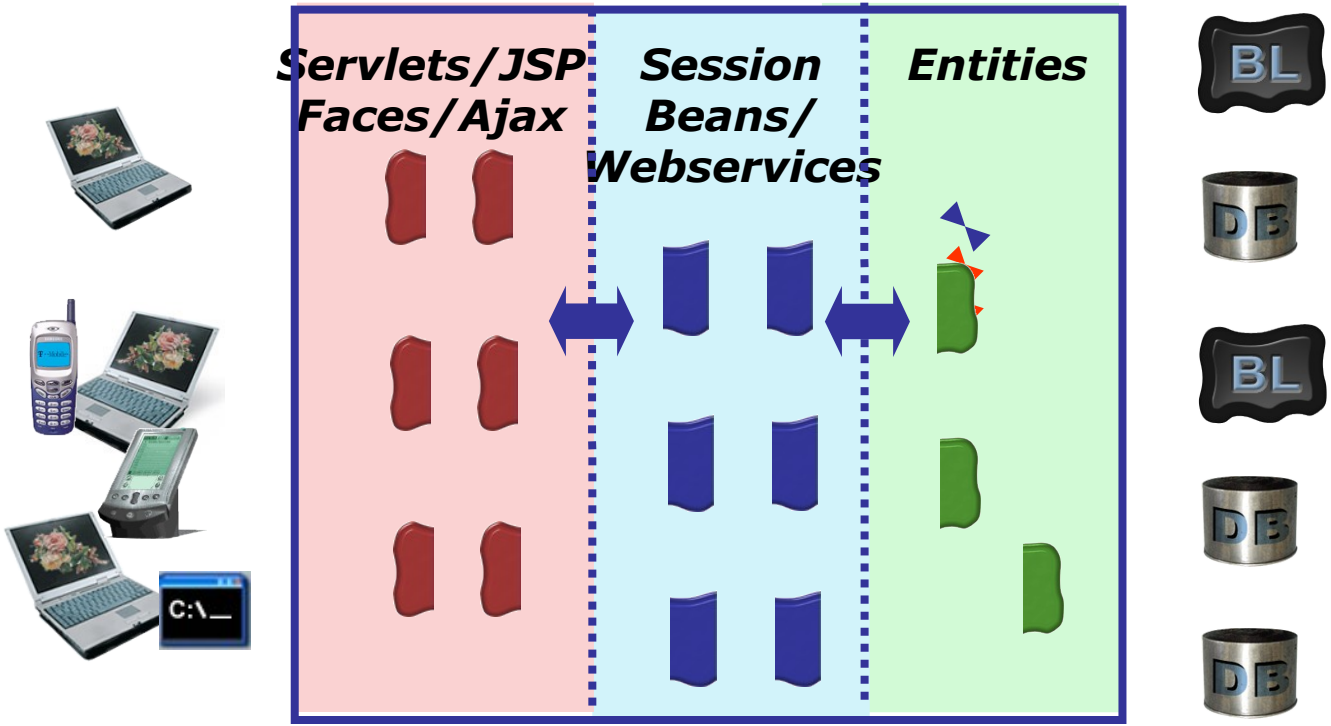


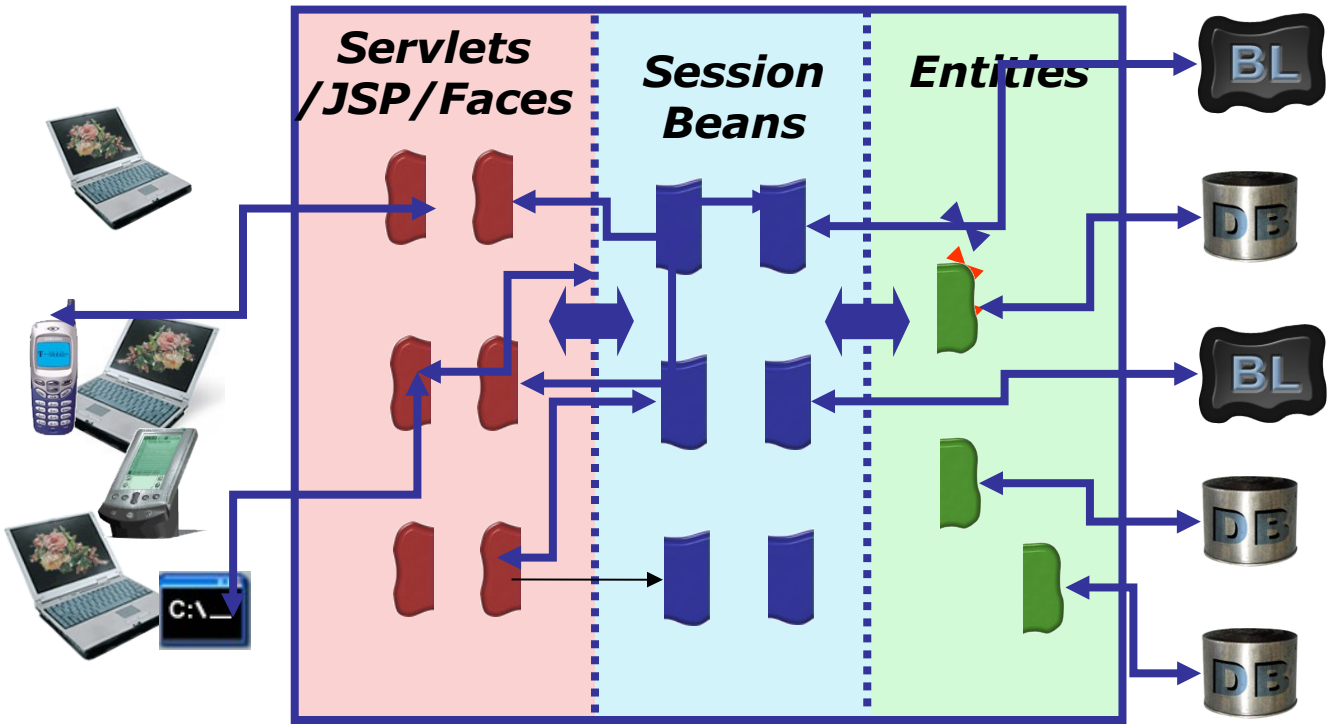
N-Tier Complexity



N-Tier Complexity

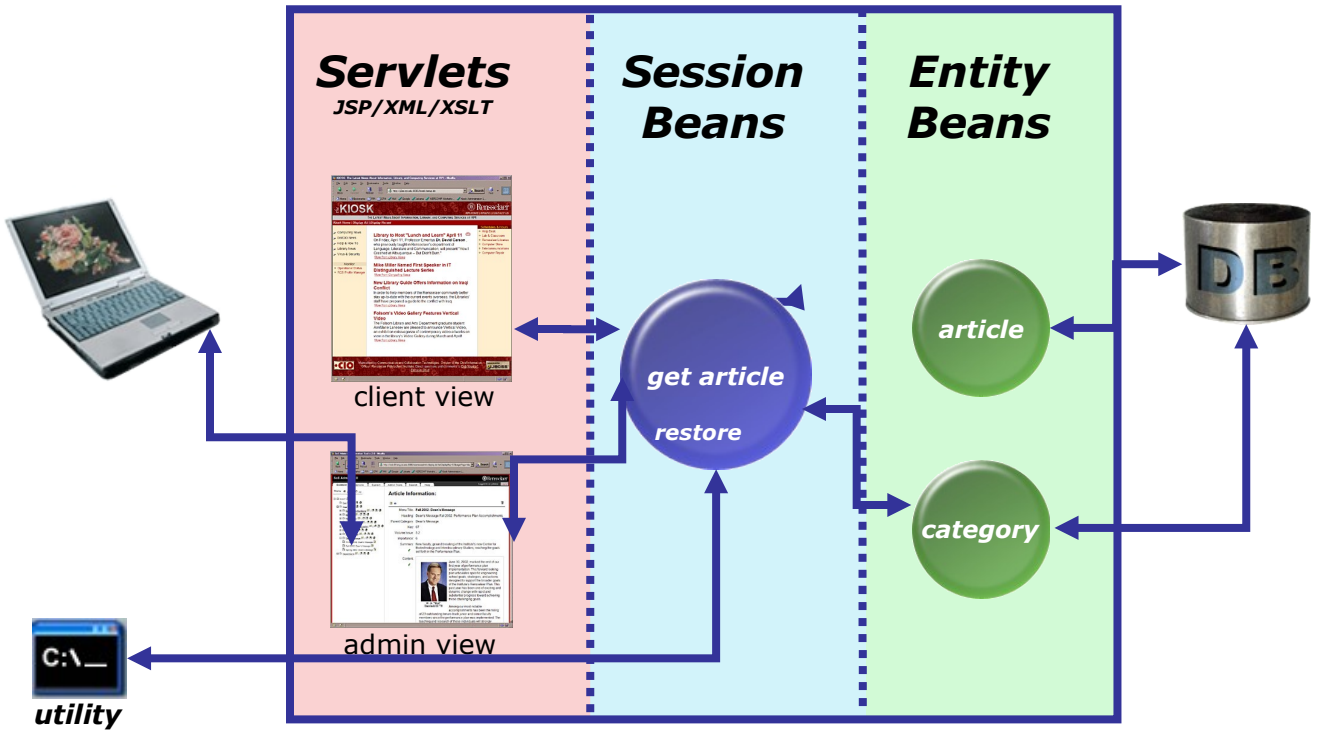



Servlets/JSP
Faces/Ajax
Session
Beans/
Webservices
Entities
BL
DB
BL
DB
DB
C:_

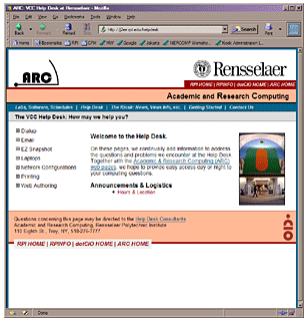


Example: The Kiosk

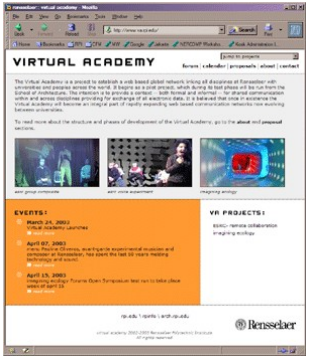
Example: The Kiosk



Clones

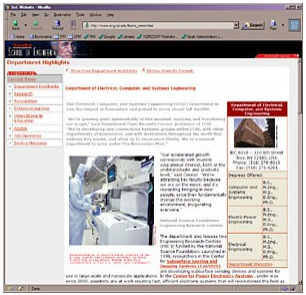


http://j2ee.rpi.edu/helpdesk



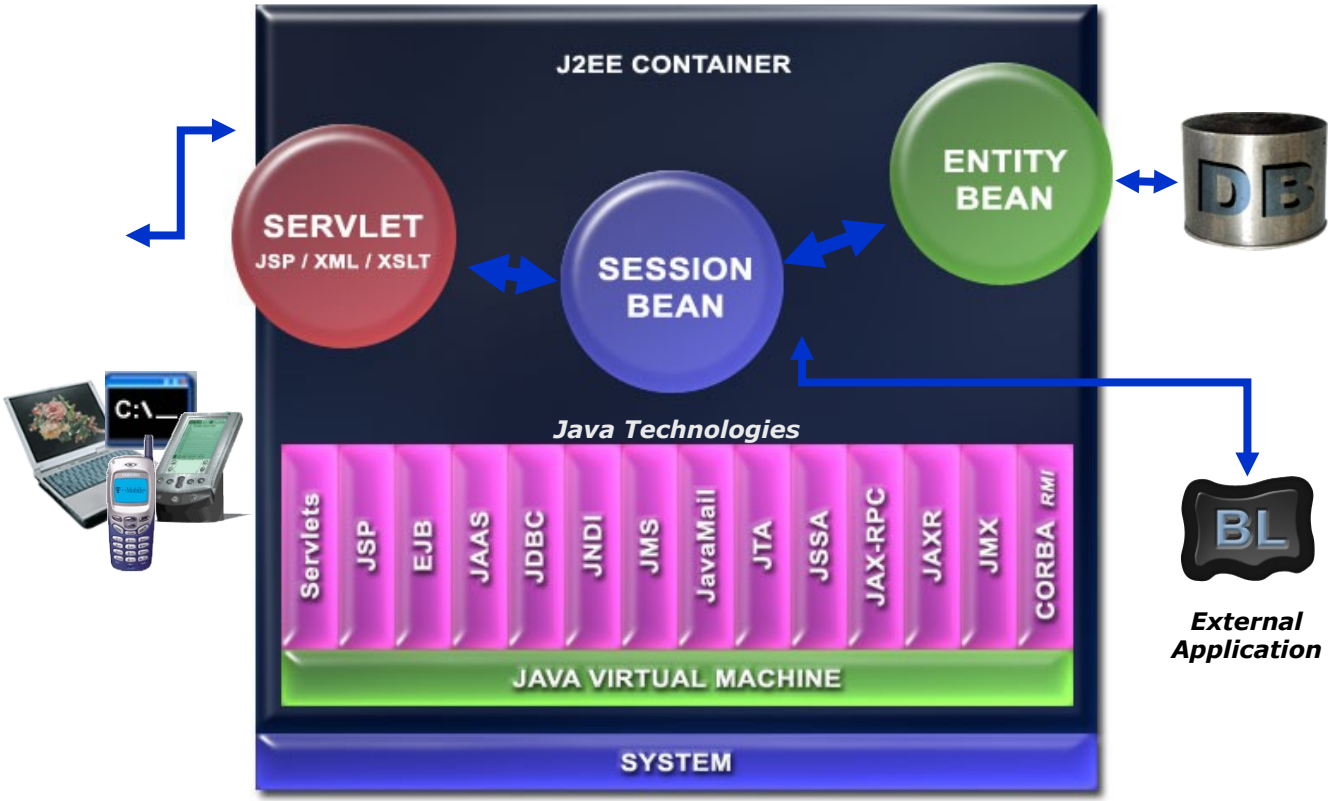
http://www.va.rpi.edu

http://j2ee.rpi.edu/kiosk



http://www.eng.rpi.edu/frame_news.html

The JavaEE Framework



WEB ENABLING TECHNOLOGIES

- Java Database Connectivity (JDBC)
- Servlets
- JSP
- JSTL/EL
- AJAX
- International and Localisation
- Web Design Patterns and Frameworks
- Struts
- Java Server Faces(JSF)
- Spring and Hibernate

BUSSINESS LOGIC AND EIS

- Enterprise Java Beans (EJB)
- Java Messaging Service (JMS)
- Java Persistence API
- JNDI
- Java API for XML
- Web Services
- REST Services
- Transactions