

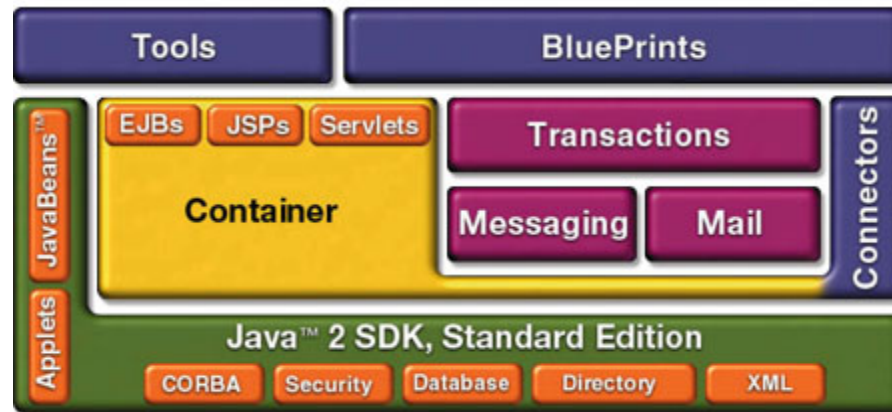
Components and Middleware

Doug Lea

State University of New York at Oswego

<http://gee.cs.oswego.edu>

Architectures and Frameworks



J2EE

Image from
java.sun.com/j2ee

- Component architectures will prevail for years
 - Massive middleware substrate
 - Simplified application programming
 - Frameworks, scripting, IDEs
 - Strong separation of development roles/tasks

Middleware Challenges

- Productivity and quality come at high cost
 - Ever slower and more bloated systems
 - Example: RPC → RMI → SOAP
 - Continuing need for better algorithms & designs
 - persistence, security, protocols, concurrency control, fault tolerance, VMs, reactive event handling, ...
- Expanding realm of supported applications
- New Compositional & Architectural issues
 - Moving from: **How to do it at all**
 - To: **How to make it fast/small/scale**

Quality of Service Challenges

- QoS becoming intrinsic to systems
 - Convergence of RT and enterprise systems?
 - Multimedia, telecom, time-dependent protocols
- Usually requires bounded latencies
 - Also memory, bandwidth, IO, ... guarantees
- Must reconcile with opportunistic designs
 - Systems optimize for high **average** throughput
 - Processors, caches, networking, concurrency, IO, GC, dynamic compilation and loading, ...
 - But have high **variances**
 - Three orders of magnitude not uncommon

Research Process

- Need open research platforms
 - Avoid ramp-up obstacles, facilitate collaboration and tech-transfer
- Need balance among
 - Formal analysis
 - New designs, algorithms, protocols
 - Empirical analysis
 - Proof of concept implementations
 - Usable systems