



# Performance and scalability of EJB applications

---

*Emmanuel Cecchet*

Julie Marguerite

Willy Zwaenepoel

Rice University, Systems lab



# Goal

---

- EJB application scalability evaluation
- Several implementations of the same application to evaluate
  - design patterns,
  - container design,
  - communication layers.



# Outline

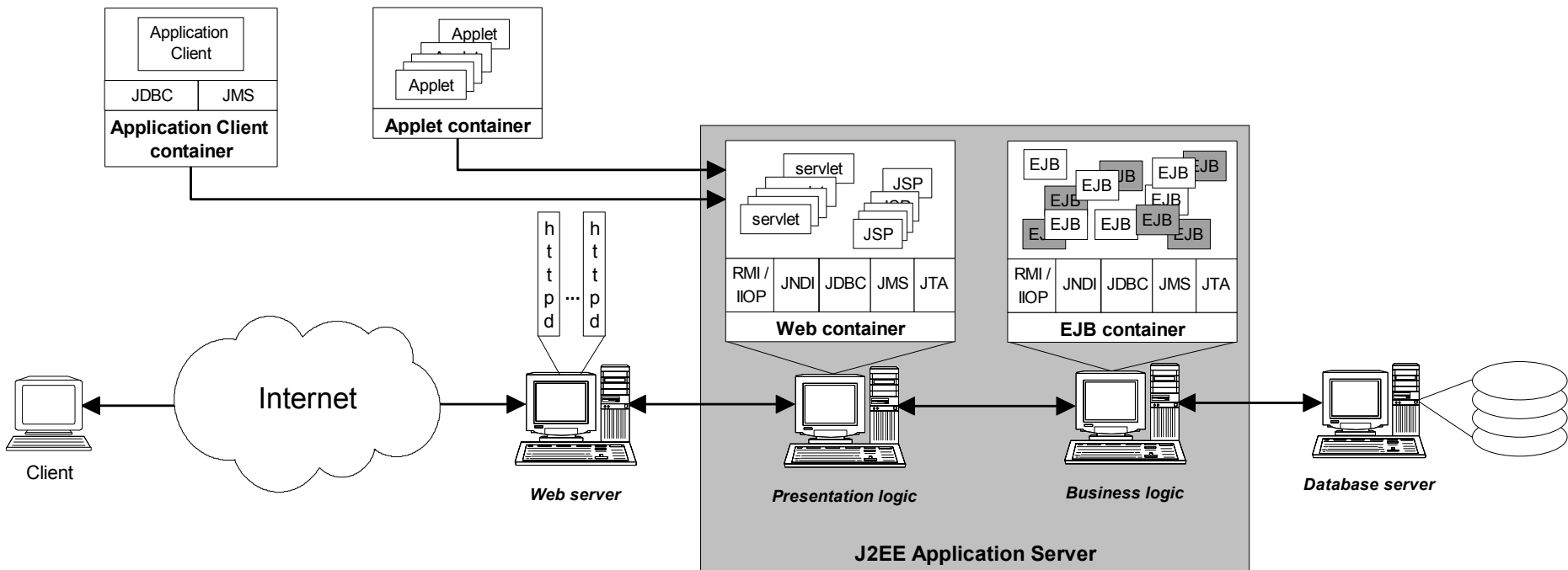
---

- Background
- RUBiS benchmark
- Performance and scalability evaluation
- Conclusion & future work

# Enterprise JavaBeans

## EJB

- large n-tiers projects
- presentation and business logic separation





# Enterprise JavaBeans

---

- Session beans
  - stateless: temporary operations
  - stateful: temporary objects
- Entity beans
  - map data stored in the database
  - Bean Managed Persistence (BMP)
  - Container Managed Persistence (CMP)



# Open Source EJB containers

---

## ■ JBoss

- [www.jboss.org](http://www.jboss.org)
- leader
- EJB 1.1 stable in v2.4, EJB 2.0/clustering in v3.0

## ■ JOnAS

- ObjectWeb: INRIA, FT R&D, Bull
  - adaptable middleware platform ([www.objectweb.org](http://www.objectweb.org))
  - JOnAS: [www.objectweb.org/jonas](http://www.objectweb.org/jonas)
- EJB 1.1 stable in v2.4, almost EJB 2.0 in v2.6
- Jonathan: generic ORB
  - David: CORBA personality
  - Jeremie: RMI personality



# Container design

---

- Reflexive approach (JBoss 2.4)
  - dynamic proxy
  - home and component interfaces generated at run-time
  - proxy uses reflection for locating a bean or mapping method signatures
- Precompiled approach (JOnAS)
  - custom implementations of home and component interfaces
  - direct call of the bean instance
  - specific “compiler” generates
    - container classes for all beans
    - stubs and skeletons
    - use standard Java and RMI compilers



# Outline

---

- Background
- RUBiS benchmark
- Performance and scalability evaluation
- Conclusion & future work





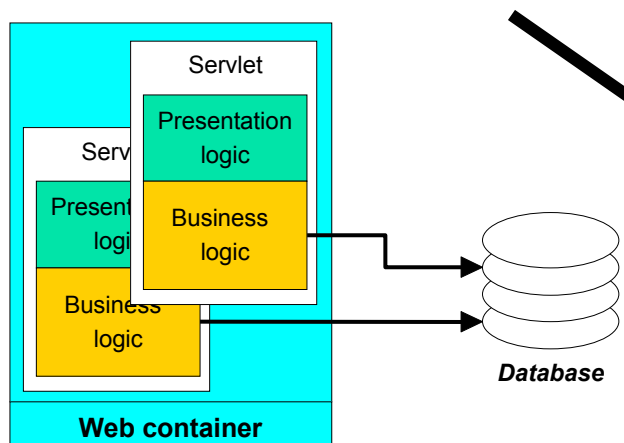
# RUBiS

---

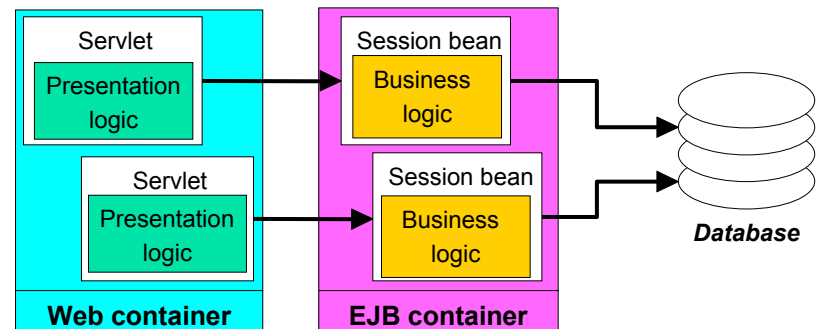
- auction site modeled after eBay
- 26 interactions
  - read-only: browsing items, myEbay, ...
  - write interactions: bidding, buying, selling, ...
- browsing mix: read-only mix
- bidding mix: 15% read-write interactions
- database: 1.4 GB
  - 1 million users, ~500000 comments
  - >500000 items, 330000 active bids

# EJB implementations

## ■ From Servlets

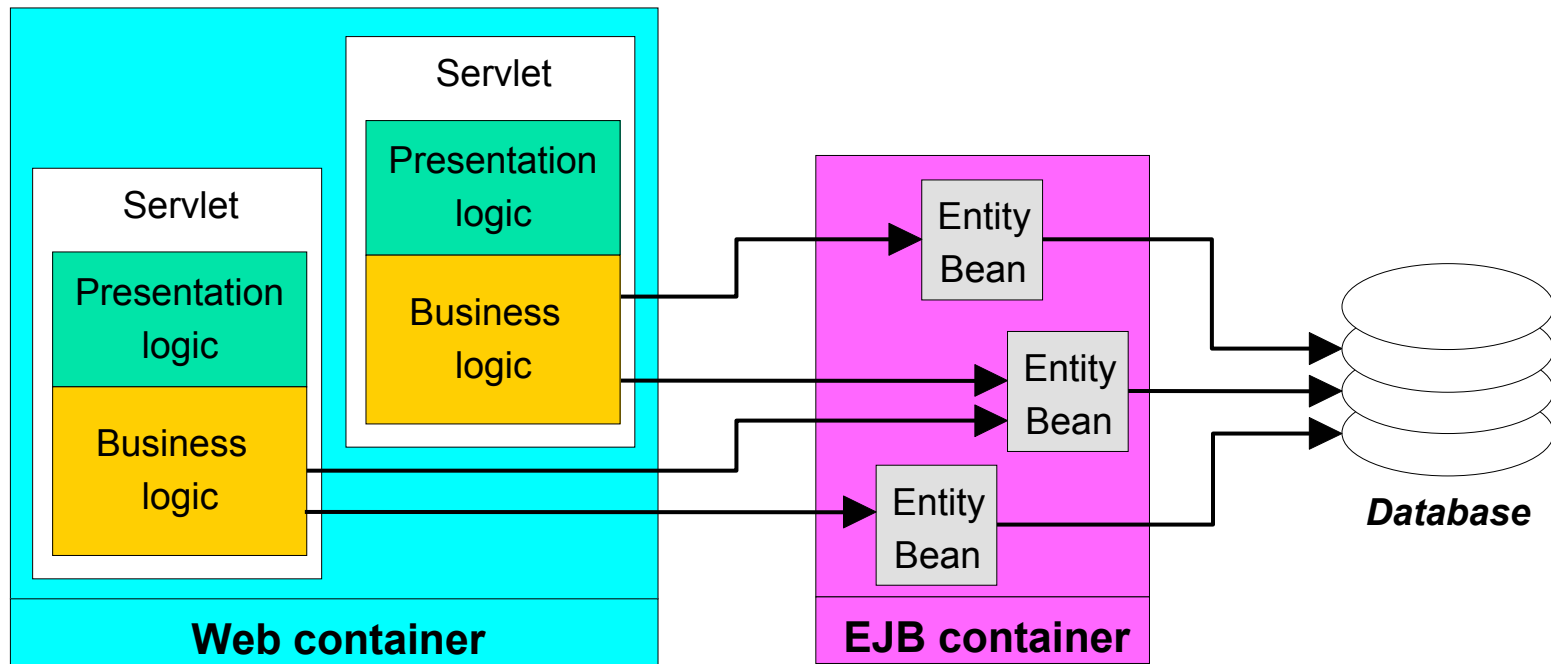


to Session Beans



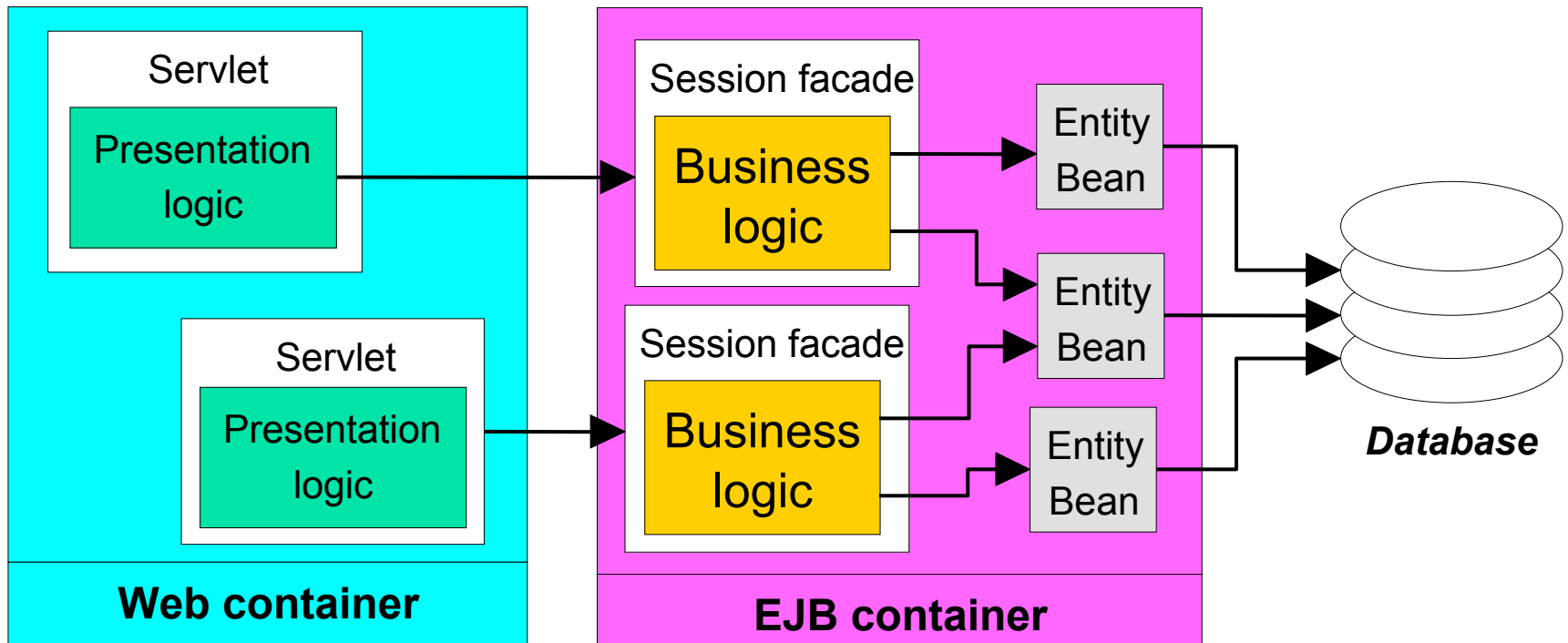
# EJB implementations

- Data Access Objects separation with Entity Beans



# EJB implementations

- Session façade





# Measurement methodology

---

- Client emulators
- 3 phases (up ramp, *runtime*, down ramp)
- 4 configurations
  - JBoss-RMI
  - JOnAS-RMI
  - JBoss optimized calls
  - JOnAS-Jeremie
- Throughput in interactions per minute
- Monitoring of cpu, disk, network and memory
- Profiling at the peak point using OptimizeIt



# Implementations summary

	<b>Presentation logic</b>		<b>Business logic</b>		<b>Total</b>	
	Classes	Lines of code	Classes	Lines of code	Classes	Lines of code
Servlets only	25	4590	-	-	25	4590
Session Beans	22	2730	51	5270	73	8000
EB CMP	23	3980	40	6780	63	10760
EB BMP	23	3980	40	9850	63	13830
Session façade	22	2660	85	10780	107	13440
EJB 2.0 local	22	2725	91	11070	113	13795

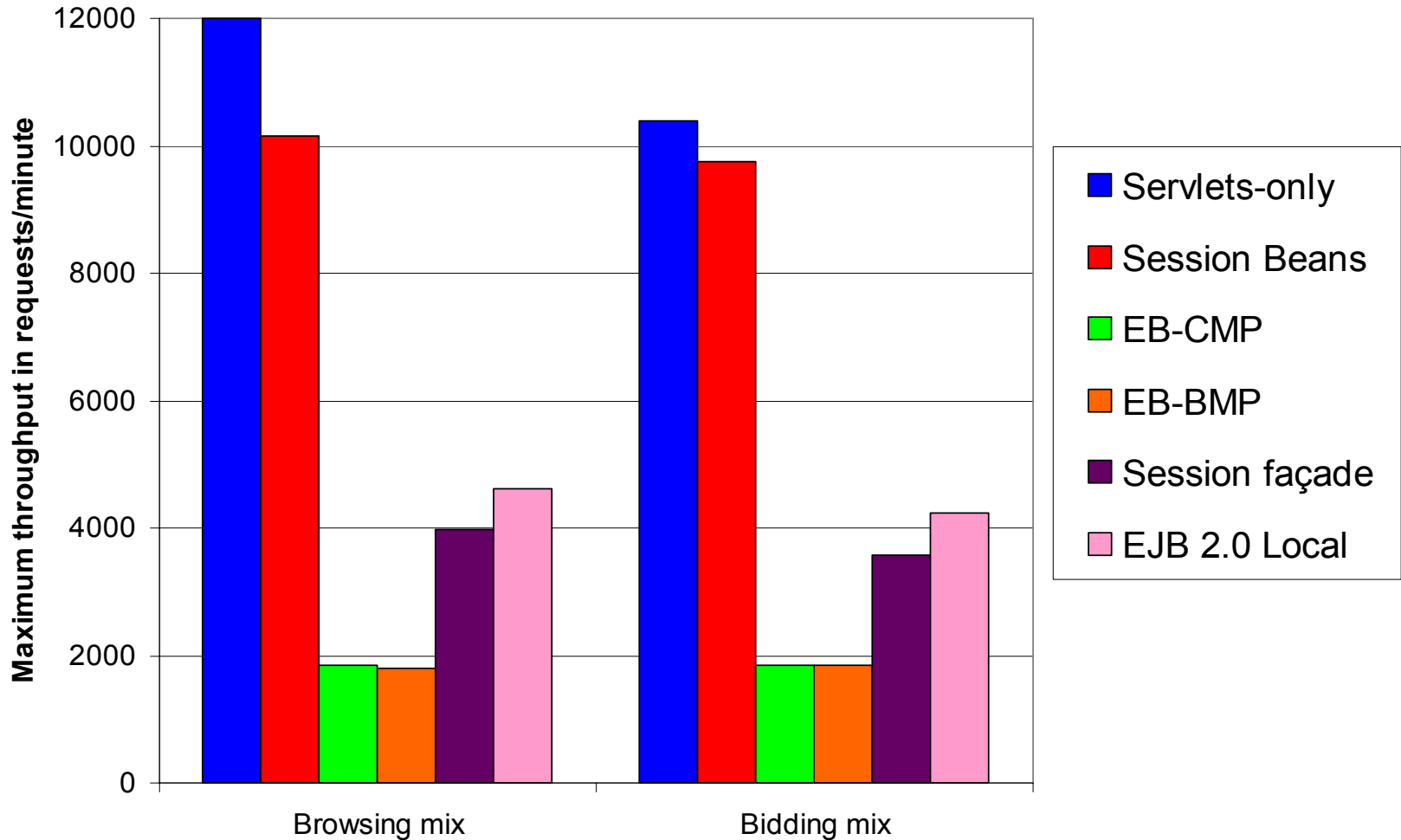


# Outline

---

- Background
- RUBiS benchmark
- Performance and scalability evaluation
- Conclusion & future work

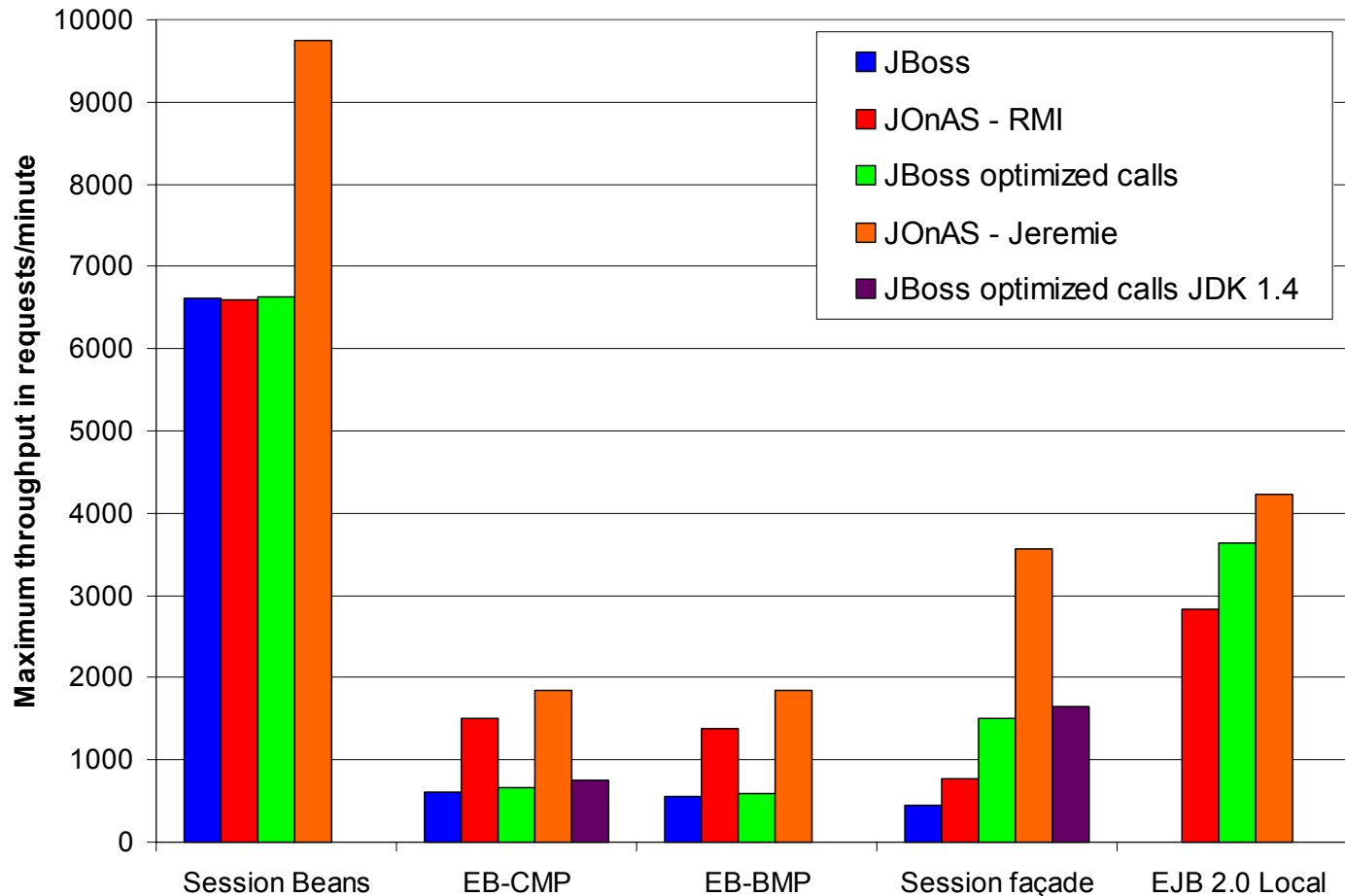
# RUBiS – Results overview





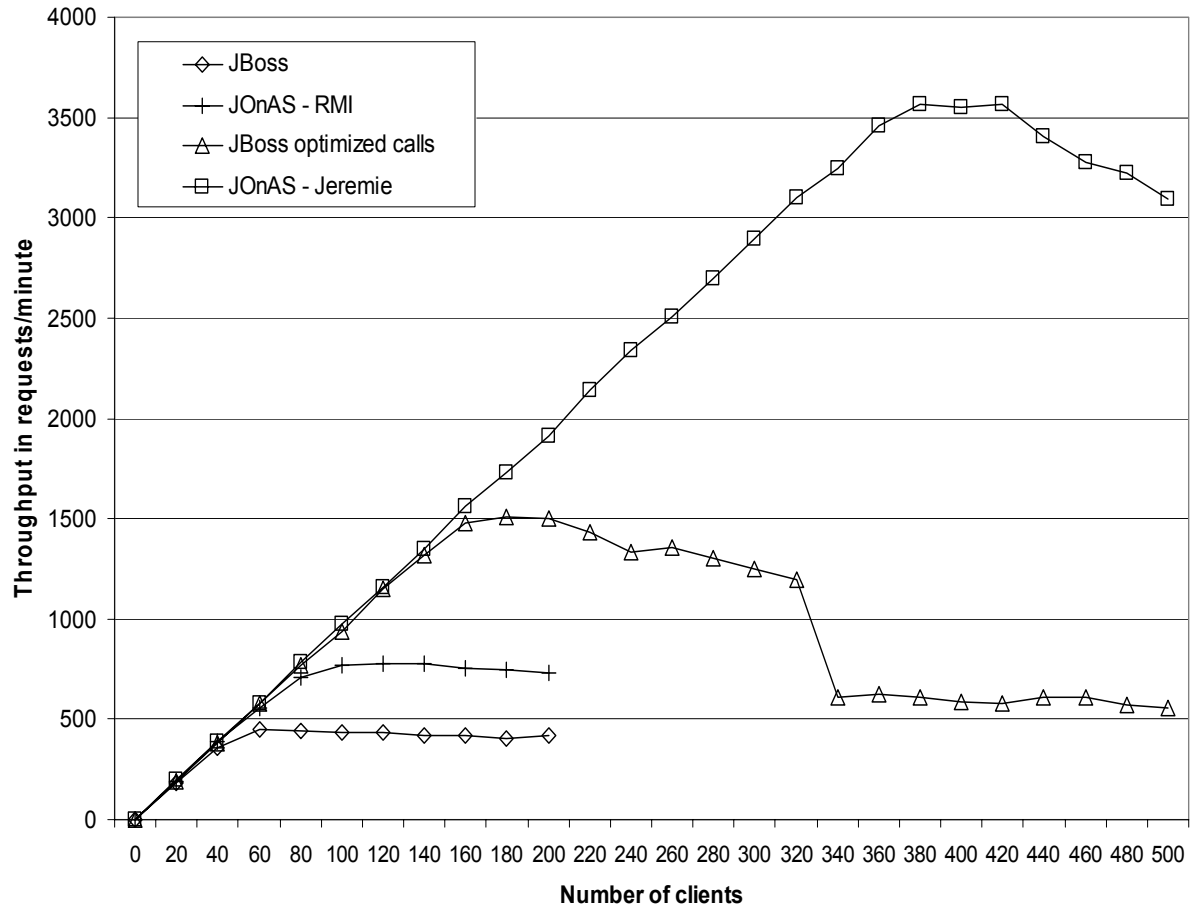
# RUBiS – Throughput

## ■ Bidding mix

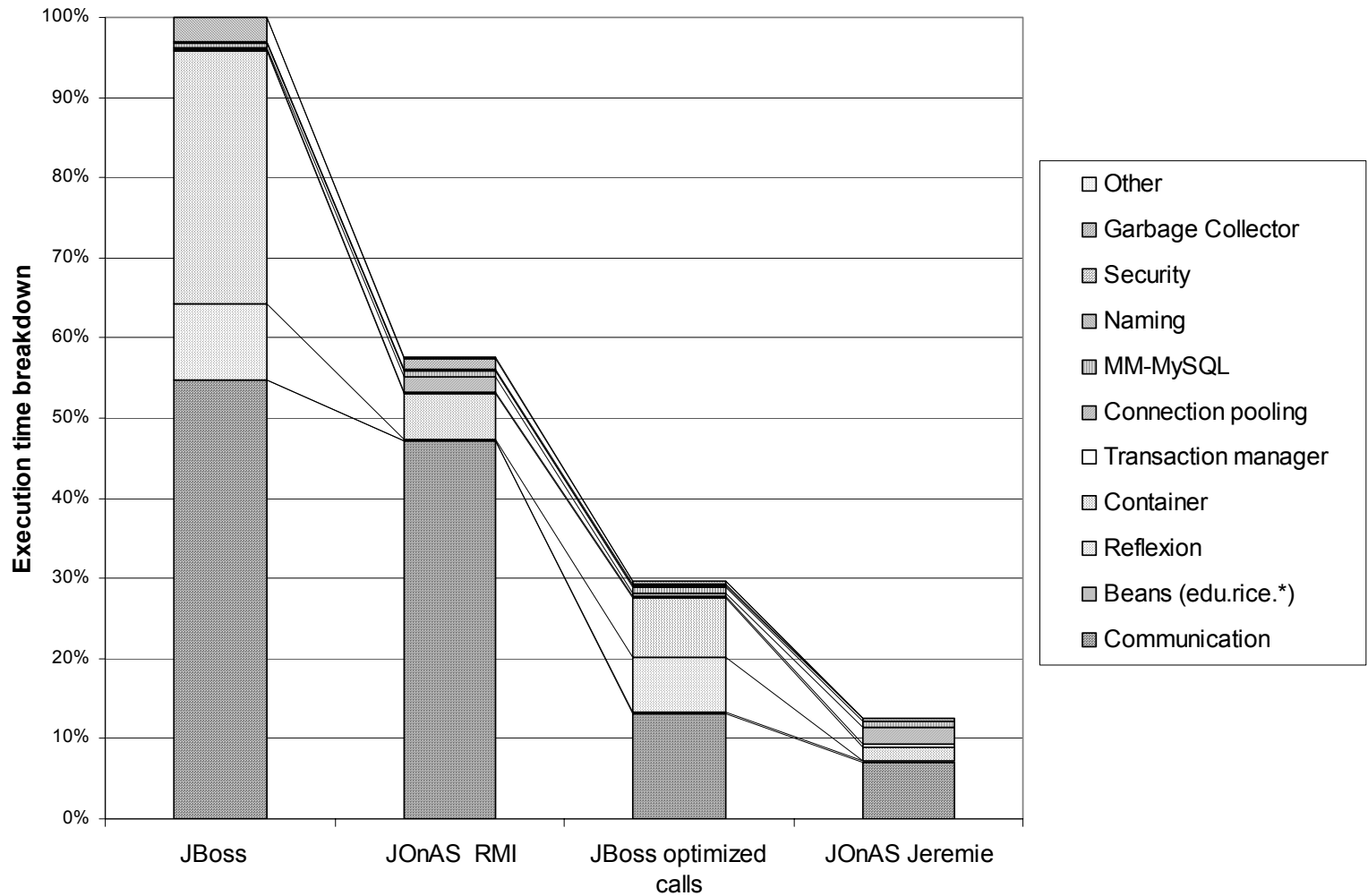


# RUBiS – Session façade

## ■ Bidding mix



# RUBiS – Session façade





# Main results (1/2)

---

- Session beans
  - performance similar to servlets-only
  - bottleneck: communication layers
  - container design has low impact on performance
- Entity beans 1.1
  - BMP or CMP have similar performance
  - too fine grain data access
  - container design important (reflection is a bottleneck even with JDK 1.4)
  - performance do not scale



# Main results (2/2)

---

- Session façade
  - reflection cost even more important
  - optimized local communications necessary
  - only JOnAS/Jérémie scales
- EJB 2.0 local interfaces
  - new containers (JBoss 3.0/BCEL, JOnAS 2.5)
  - better perf but still below half of session beans
  - 16% improvement with Jeremie and local comms
  - JBoss/JOnAS same perfs
- less than 2% of execution time in bean's user code



# Outline

---

- Background
- RUBiS benchmark
- Performance and scalability evaluation
- **Conclusion & future work**



# Conclusion

---

- EJB performance can be close to Servlets/PHP
- EJB 1.1 persistence gives poor performance due to fine-grain access exposed by EBs
- Container design has a significant impact on performance
  - communication is the main bottleneck
  - reflection adds significant overhead
  - user bean code represents less than 2% of exec time
- RUBiS
  - source code and experimental results available at <http://www.cs.rice.edu/CS/Systems/DynaServer/>
  - 400 downloads of RUBiS 1.1



# Ongoing work

---

- RUBiS
  - JMS
  - EJB 2.0 CMP
- J2EE clustering
  - workload generation
  - database clustering
- RUBBoS
  - benchmark modeled after slashdot.org
  - more coarse grain data





Questions ?

---

Questions ?

<http://www.cs.rice.edu/CS/Systems/DynaServer/>

email: [cecchet@rice.edu](mailto:cecchet@rice.edu)