

Java EE: Demo

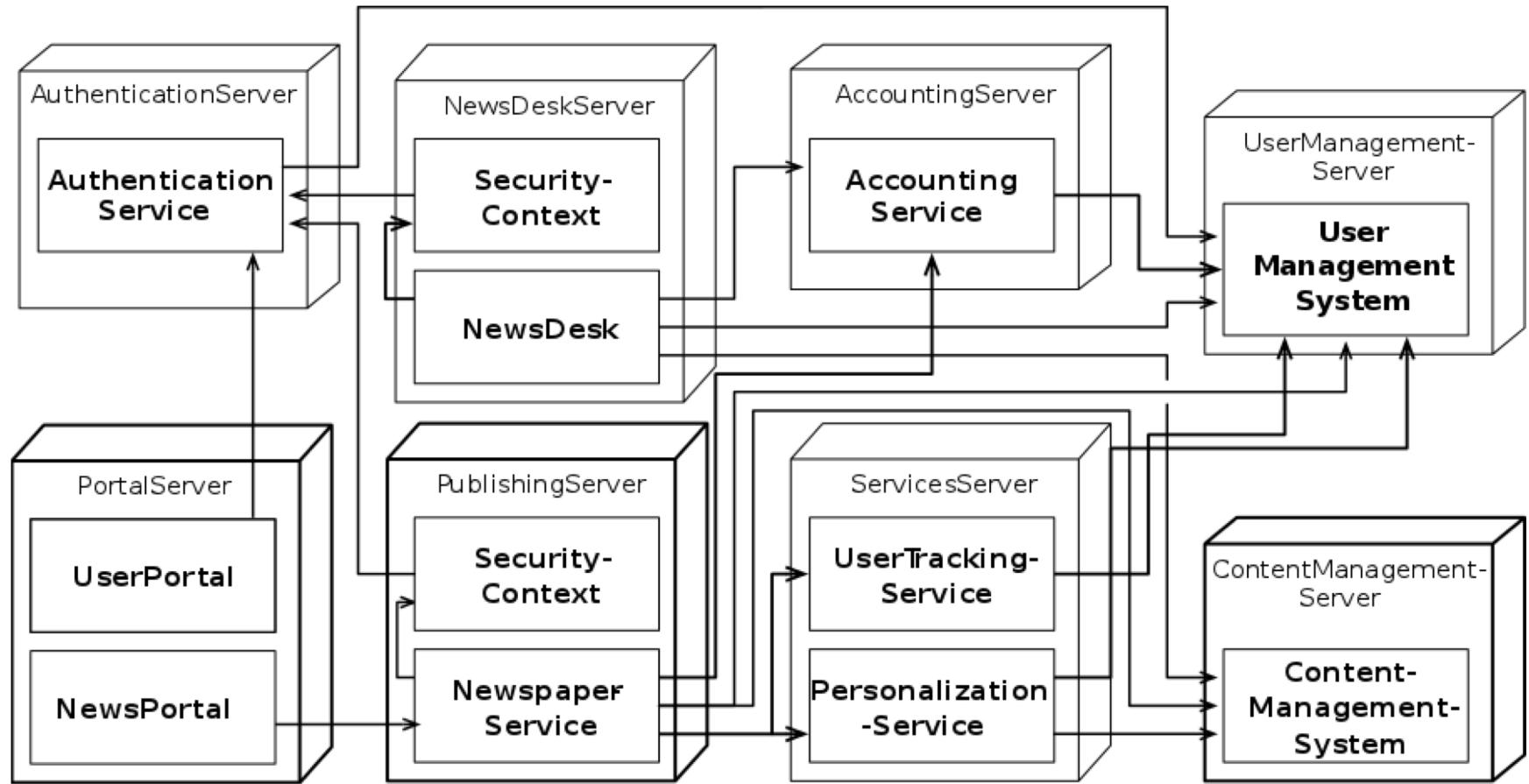
Emad Heydari Beni, Bert Lagaisse &

Wouter Joosen

October 2020



E-Publishing



Overview

- 
1. Why Java EE?
 2. Duke's Bank
 - Session beans
 - Persistence
 - Security
 3. Demo
 4. Overview of the lab sessions

Why Java EE and not RMI?

- Too many customers to serve from one machine
 - Multiple tiers
- Distribution is hard
 - Communication
 - Independent failure
 - Security
 - Consistency

Why Java EE and not RMI?

- Java EE
 - Offers component models and containers for different tiers
 - Provides common services to support distributed application

Why Java EE and not RMI?

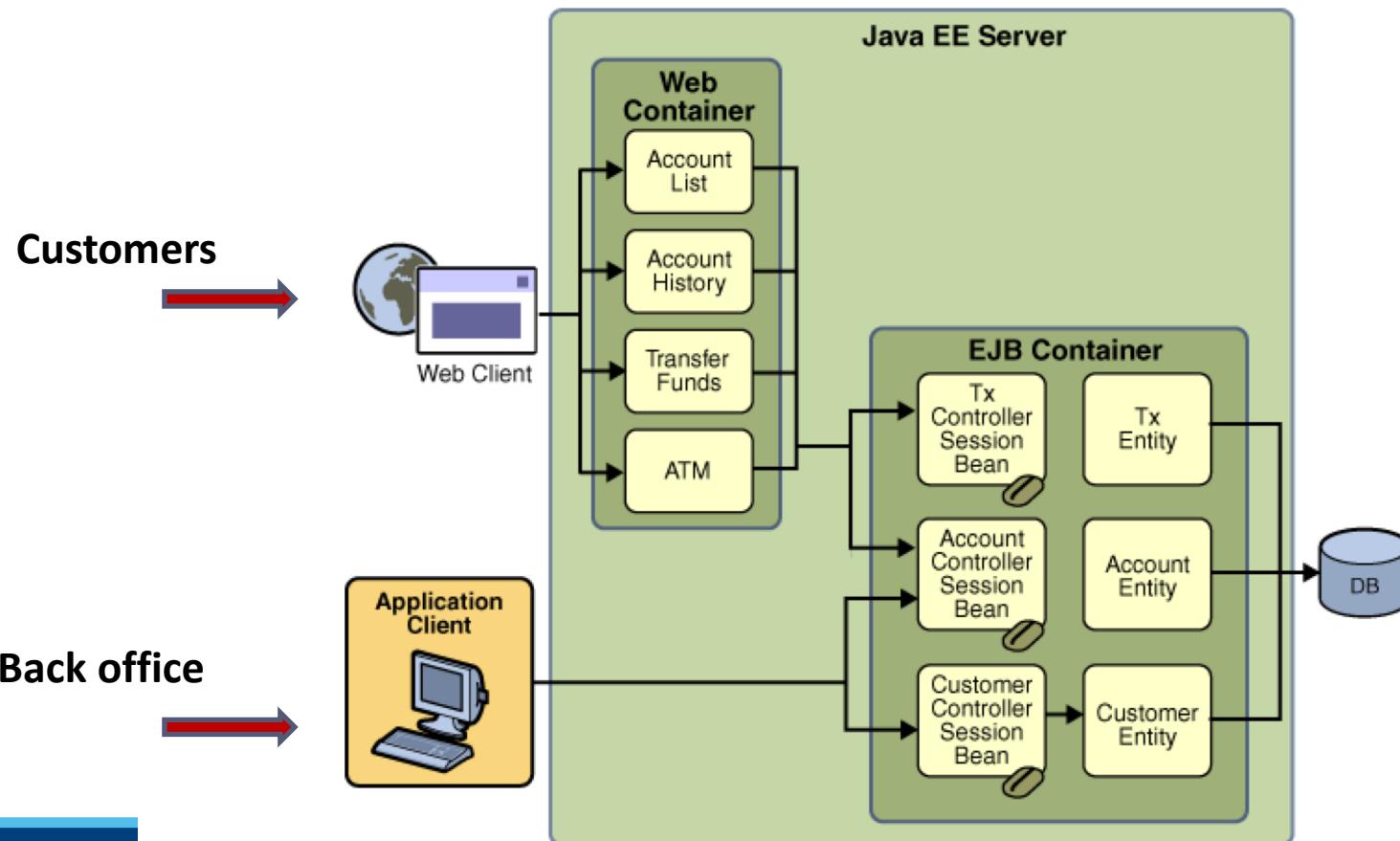
- Component models & containers
 - Web components (Servlet, JSP, JSF)
 - EJB components (EJB)
 - Persistence component (entities)
- Container services
 - ORB (RMI, RMI-IIOP)
 - Life-cycle management
 - Naming (JNDI)
 - Persistence (JPA)
 - Transactions (JTA)

Why Java EE and not RMI?

- Java EE
 - Separation of concerns
 - Developer
 - Deployer / system administrator
 - ↔ Java RMI
 - No standardized configuration
 - Deployment information in code

Duke's Bank

Client Tier Web Tier Business tier EIS Tier



Session Beans

- A client's view of the application
- Hiding server routines
 - Business logic
 - Database access
 - Entity relation management
 - Error checking and handling

AccountControllerBean
CustomerControllerBean
TxControllerBean



Persistence

- Business entities
- Object view of these database tables:
 - *bank_account*
 - *bank_customer*
 - *Bank_tx*
- The enterprise bean container manages all data in the underlying data source, including adding, updating, and deleting data from the database tables.
- **Transactions**
 - Container-managed transaction
 - e.g., TransferFund
- *More on this next week(s)*

Account
Customer
Tx



Security

- Protecting enterprise beans by specifying **security roles**
 - *bankAdmin*
 - *bankCustomer*
- The system restricts access to these functions to the appropriate role by setting method permissions on selected methods of the beans
 - The *bankAdmin* role → can invoke *createAccount()*
 - The *bankCustomer* role → cannot invoke *createAccount()*

Demo: Duke's bank



Overview of the lab sessions

- No web tier
 - Client = client-side component
 - Focus on back-end technologies
- Tools
 - **Lab: NetBeans + Glassfish server**
 - Supports client-side component application
 - Easy-to-use and well integrated

Overview of the lab sessions

- Session 1:
 - Introduction
 - Java EE & Session beans (EJB)
 - Debugging in the context of Java EE
- Assignment
 - Car rental agency (portal) with multiple users
- **Deadline:**

Submit your code on Toledo before Friday 6 November 2020, 19:00.

Overview of the lab sessions

- Session 2 + 3:
 - Use of additional container services
 - Persistence
 - Transactions
- Assignment
 - Making car reservations persistent
 - Making reservation = atomic operation
- **Deadline:**
Submit the final version of (1) the report and (2) the code on Toledo before Friday 20 November 2020, 19:00.

Java EE: Demo Questions?