
State, Activity, Component, Deployment Diagram

SI-215

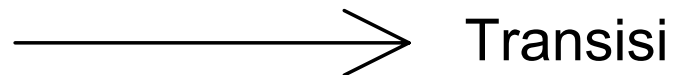
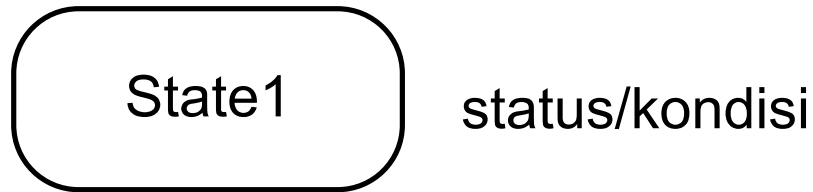
Analisa & Desain Sistem Informasi I

Rosa Ariani Sukamto, ST

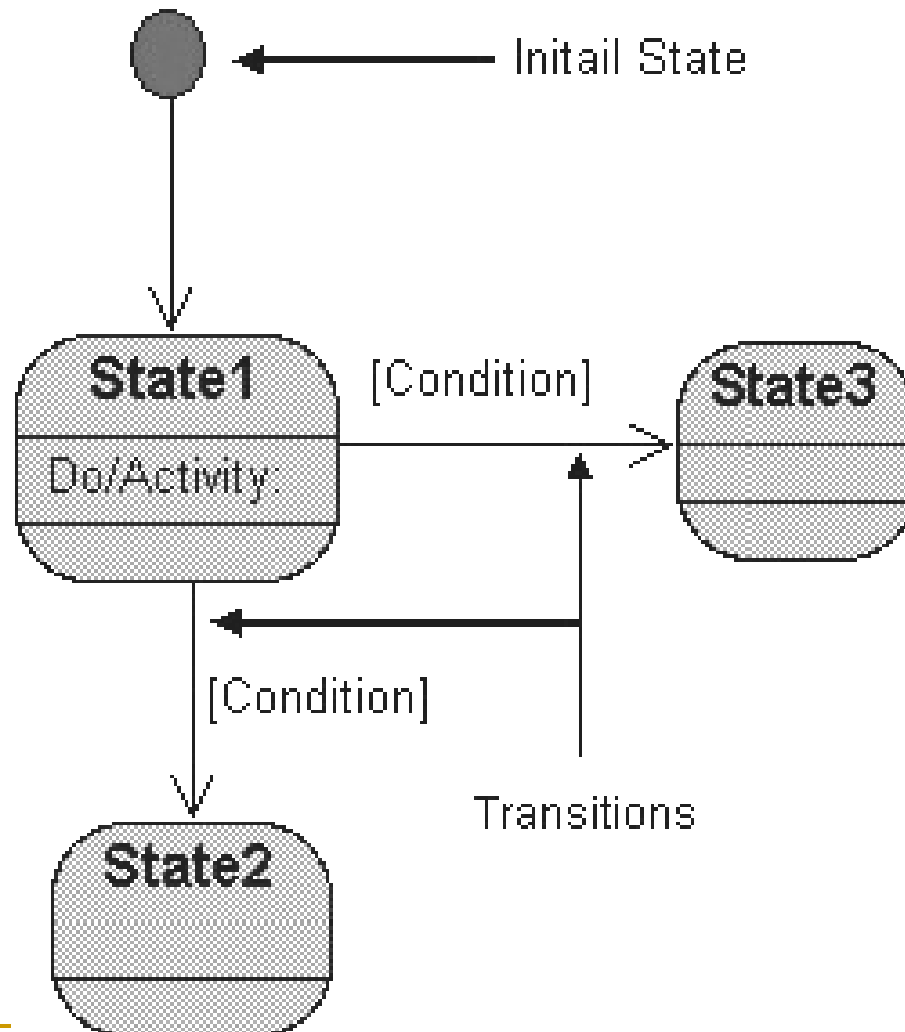
State Diagram / Statechart Diagram

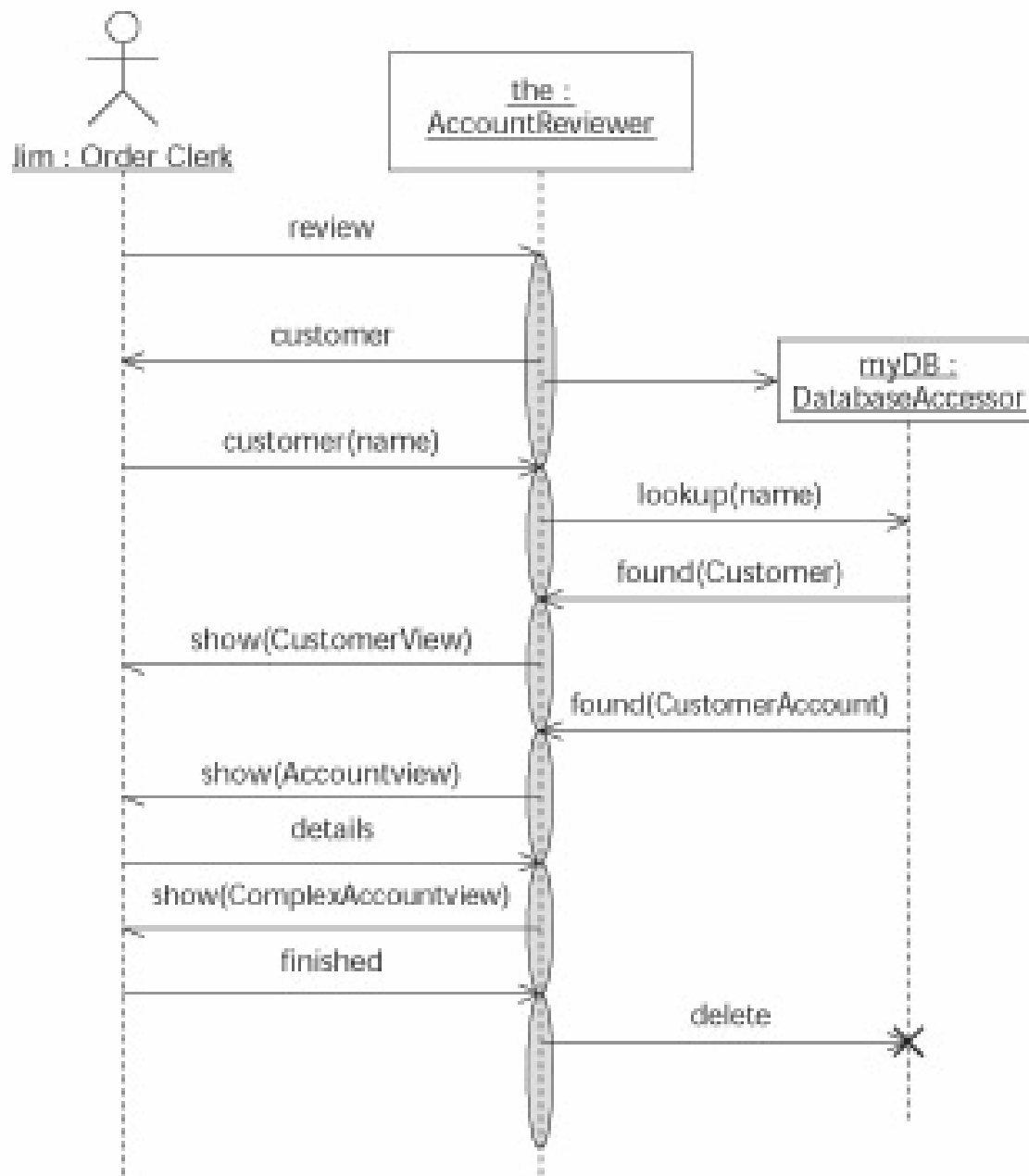
- Menunjukkan kondisi yang dapat dialami atau terjadi pada sebuah objek
- Status Awal/ Kondisi Awal
 - Kondisi awal alur hidup objek
 - Setiap state diagram hanya memiliki satu status awal
- Status Akhir/ Kondisi Akhir
 - Kondisi akhir alur hidup objek
- Transisi
 - Garis transisi antar status pada alur hidup objek

State Diagram / Statechart Diagram

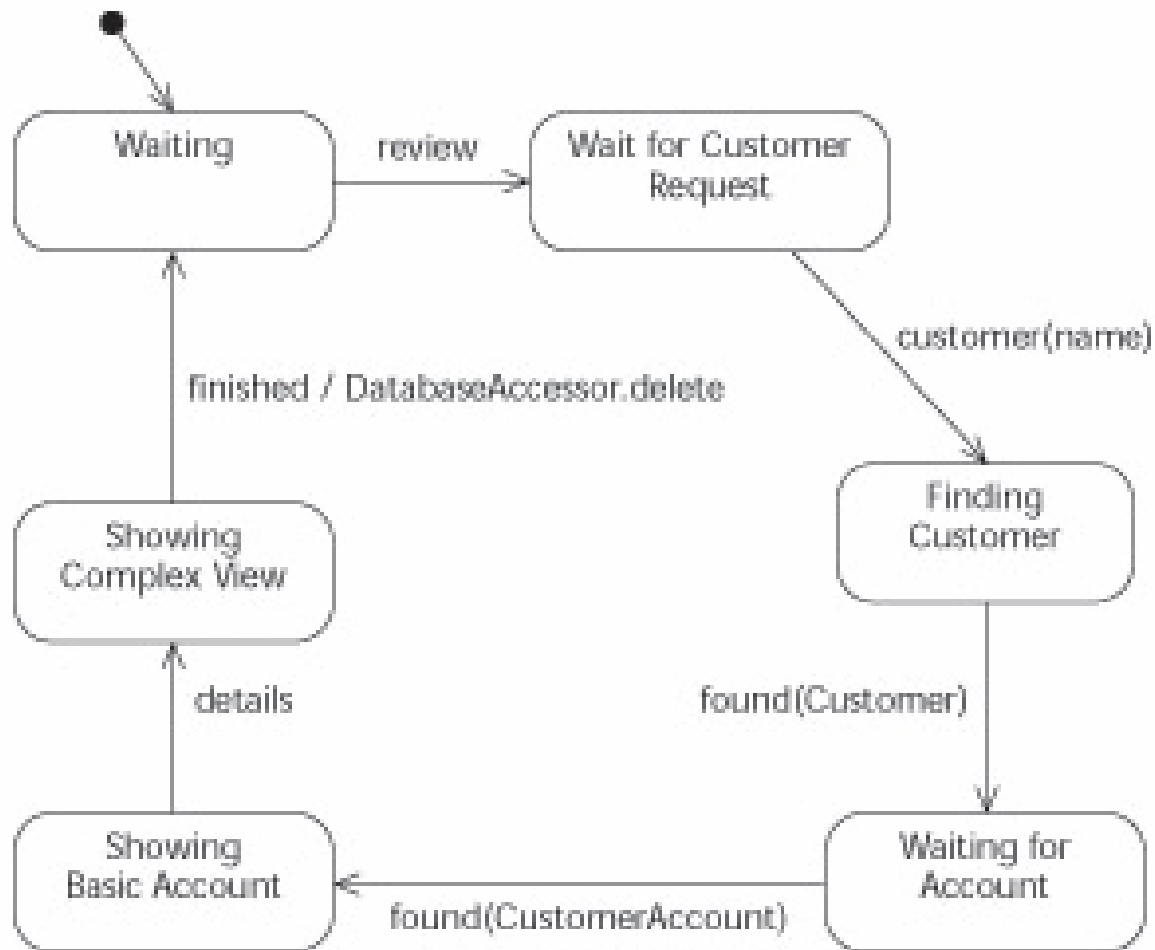


State Diagram / Statechart Diagram

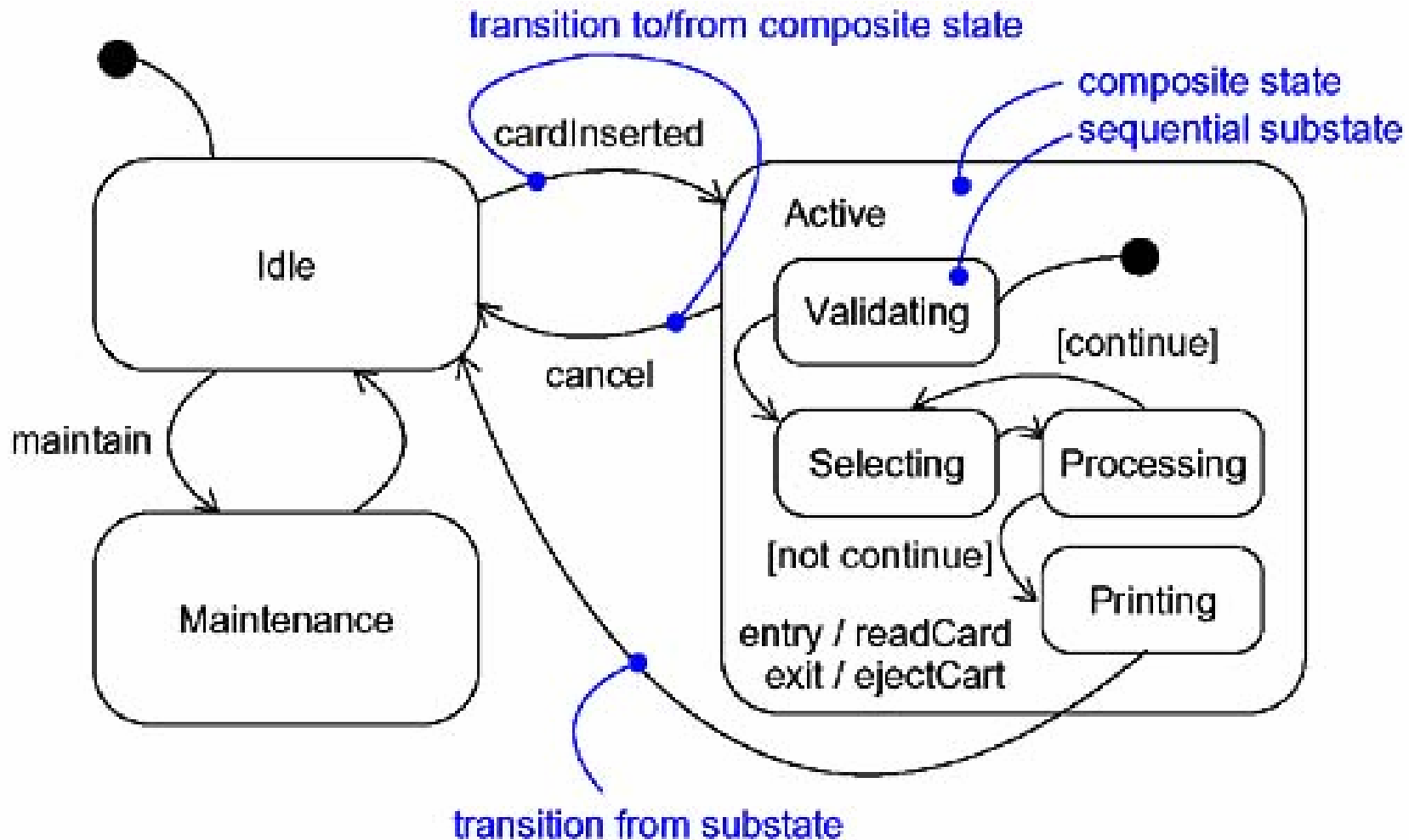




State Diagram Objek the sebagai AccountReviewer

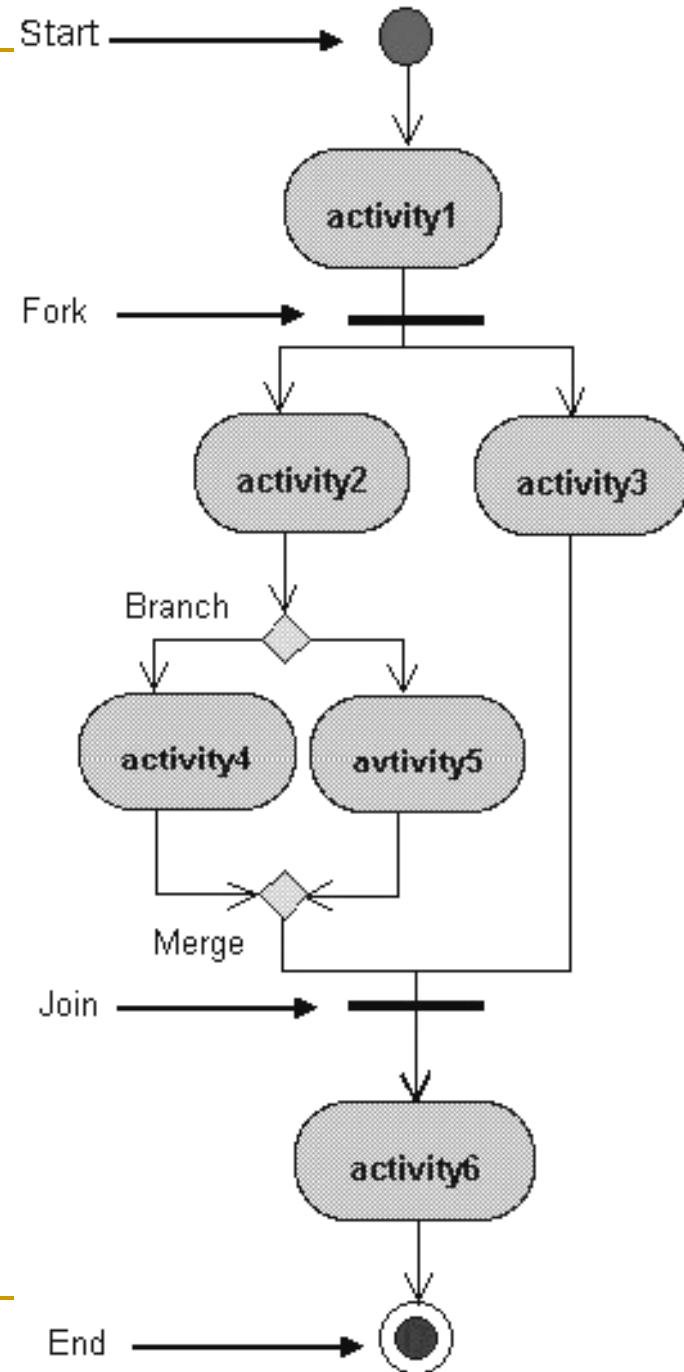


State Diagram / Statechart Diagram



Activity Diagram

- Menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis



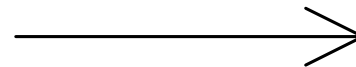
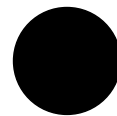
Activity Diagram



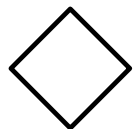
aktivitas



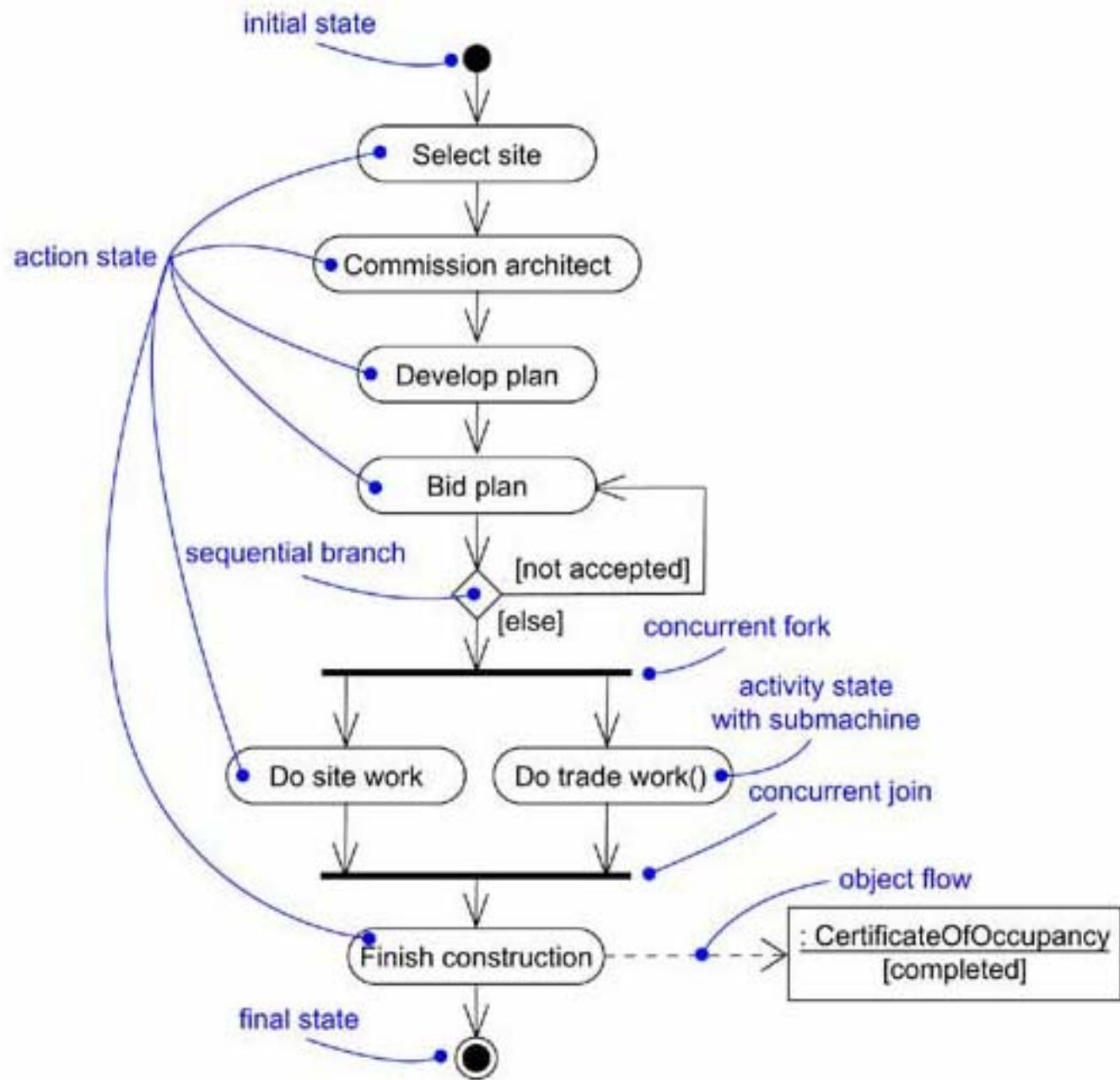
Final State



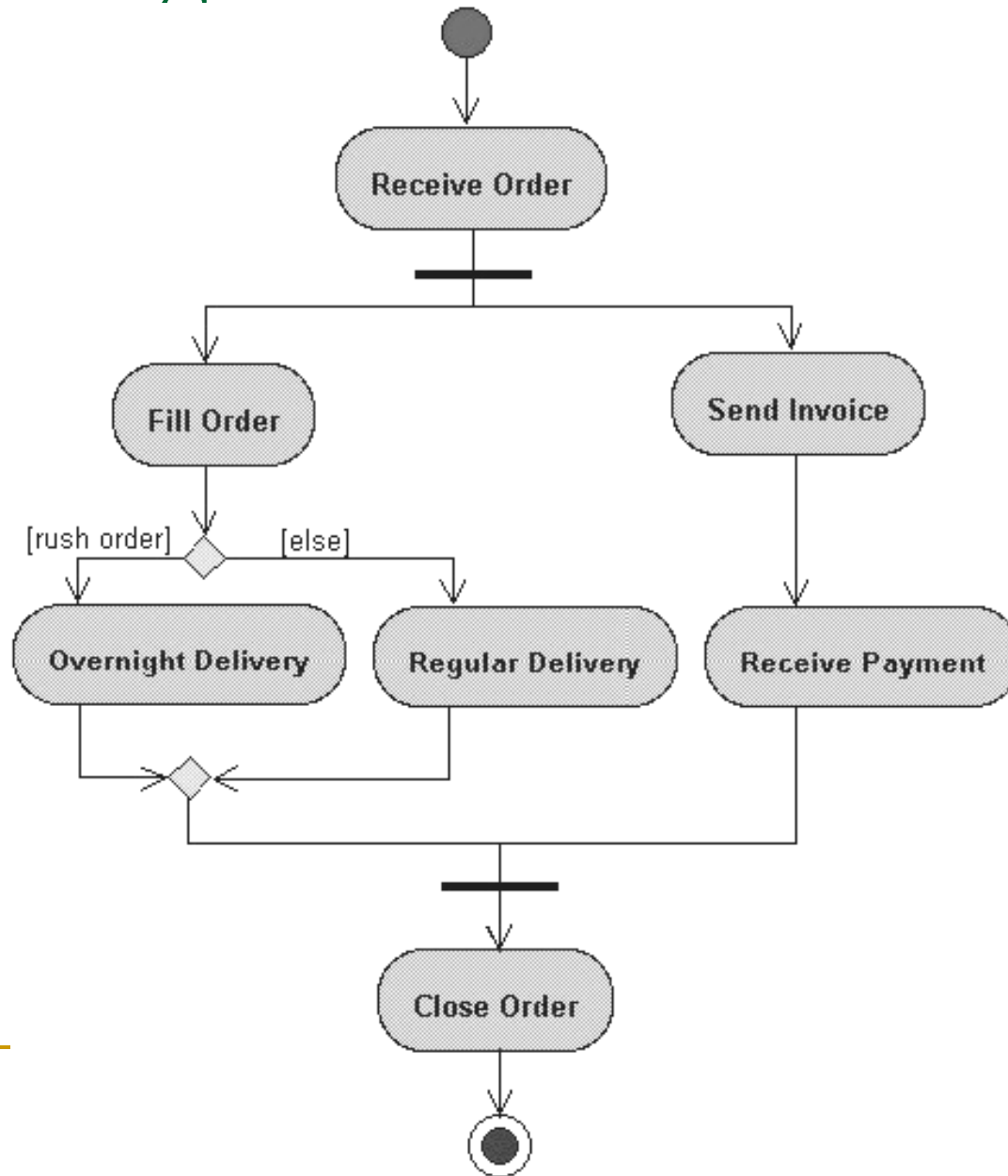
Transisi



decision

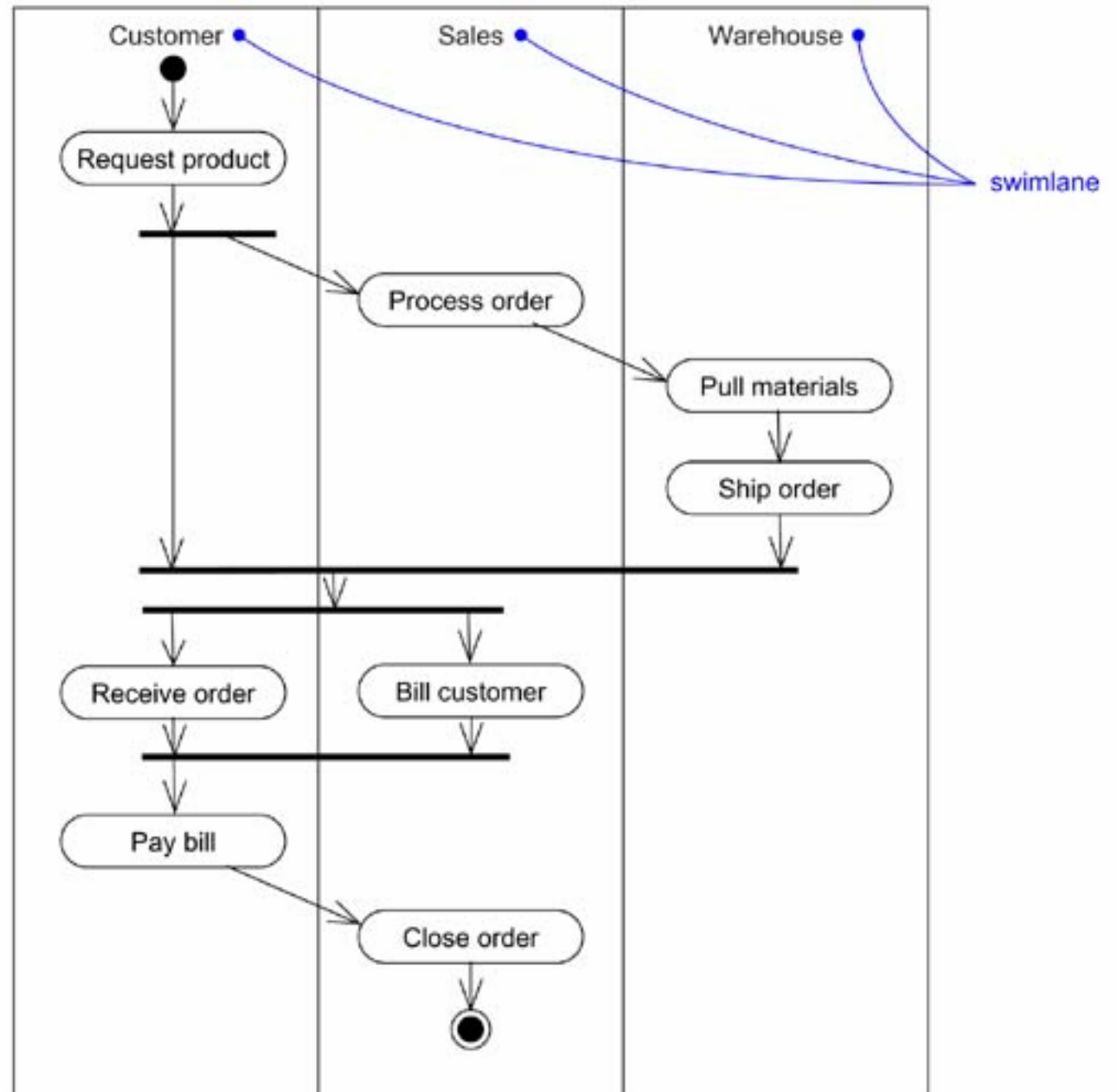


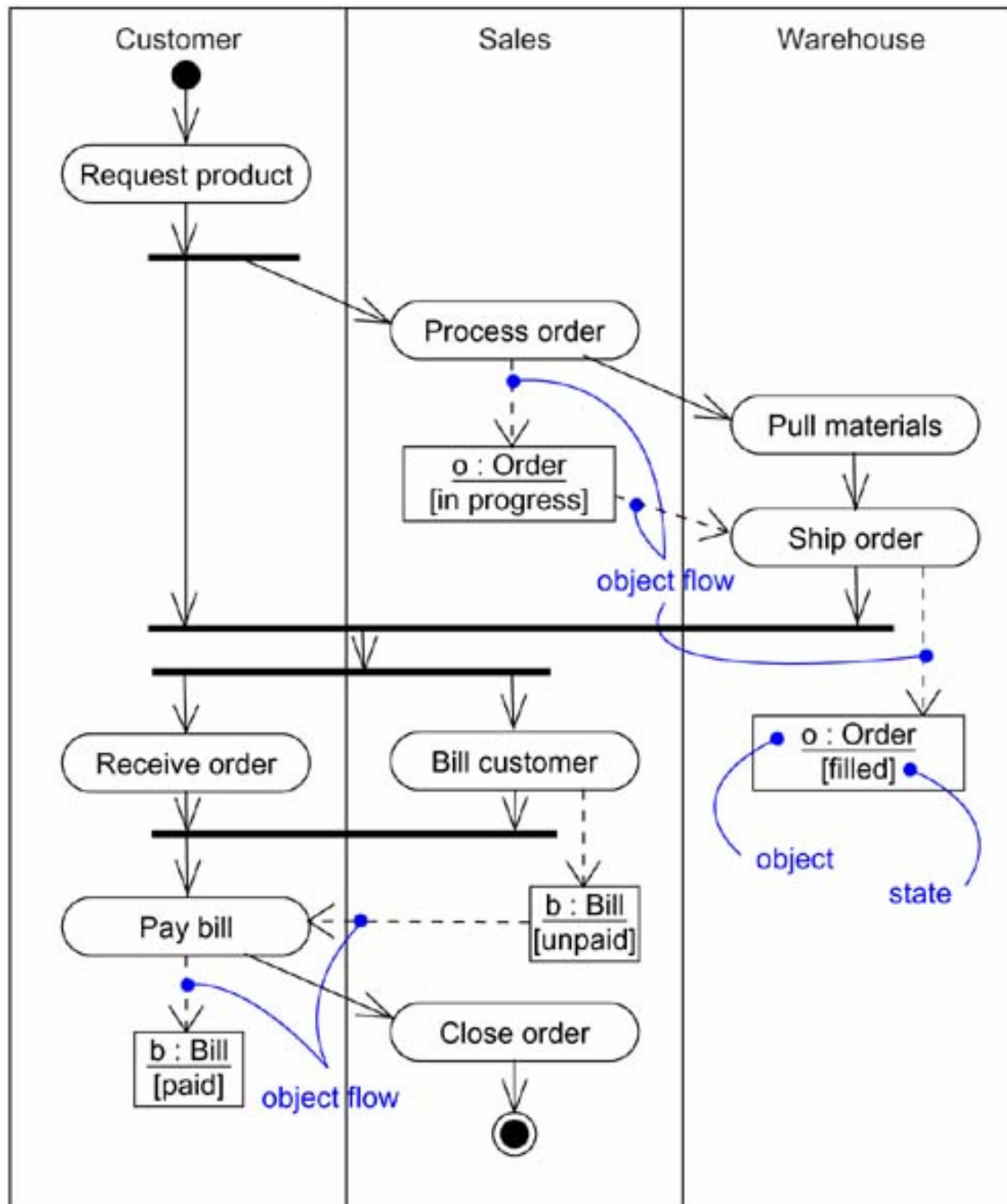
Activity Diagram



Swimlane:

Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

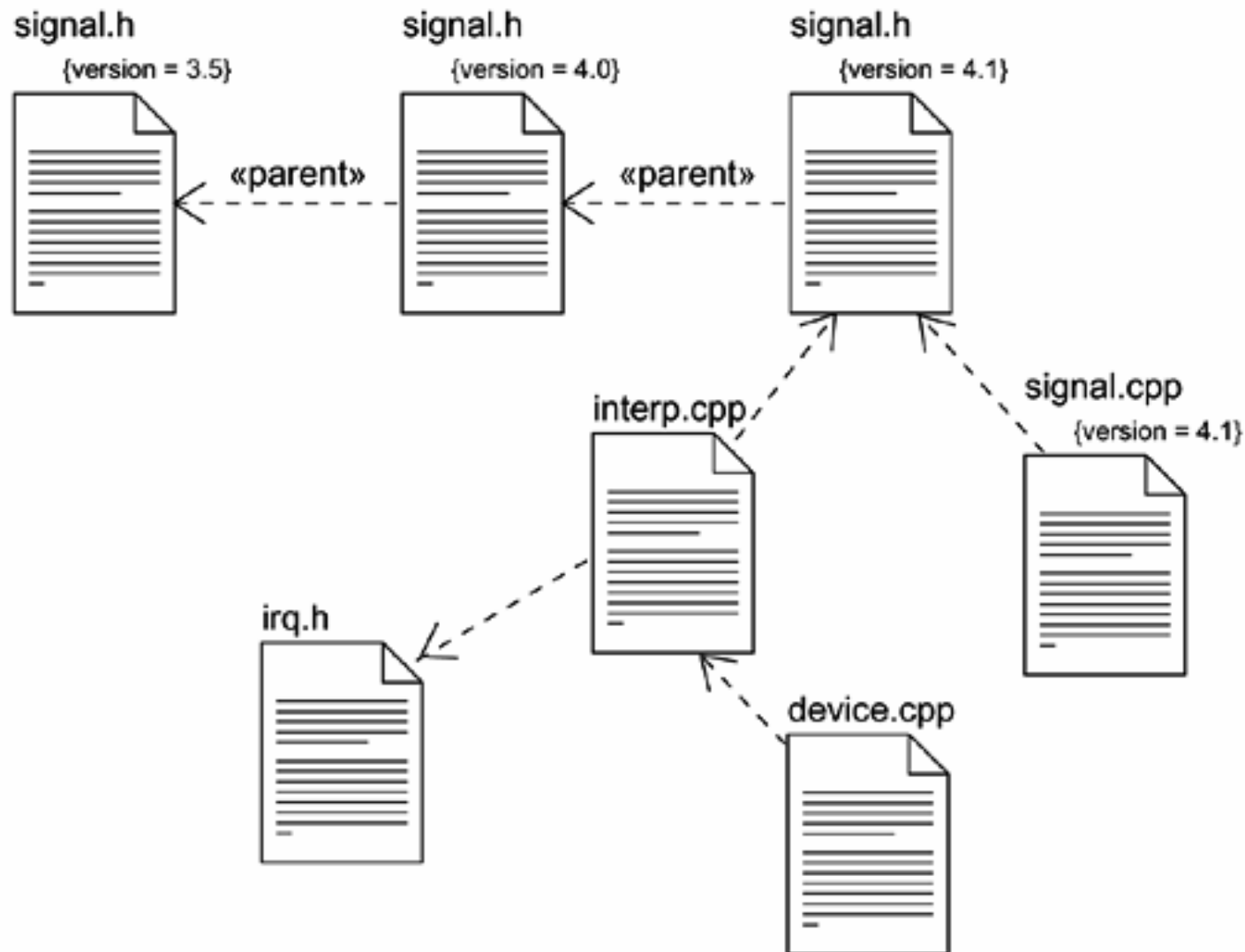




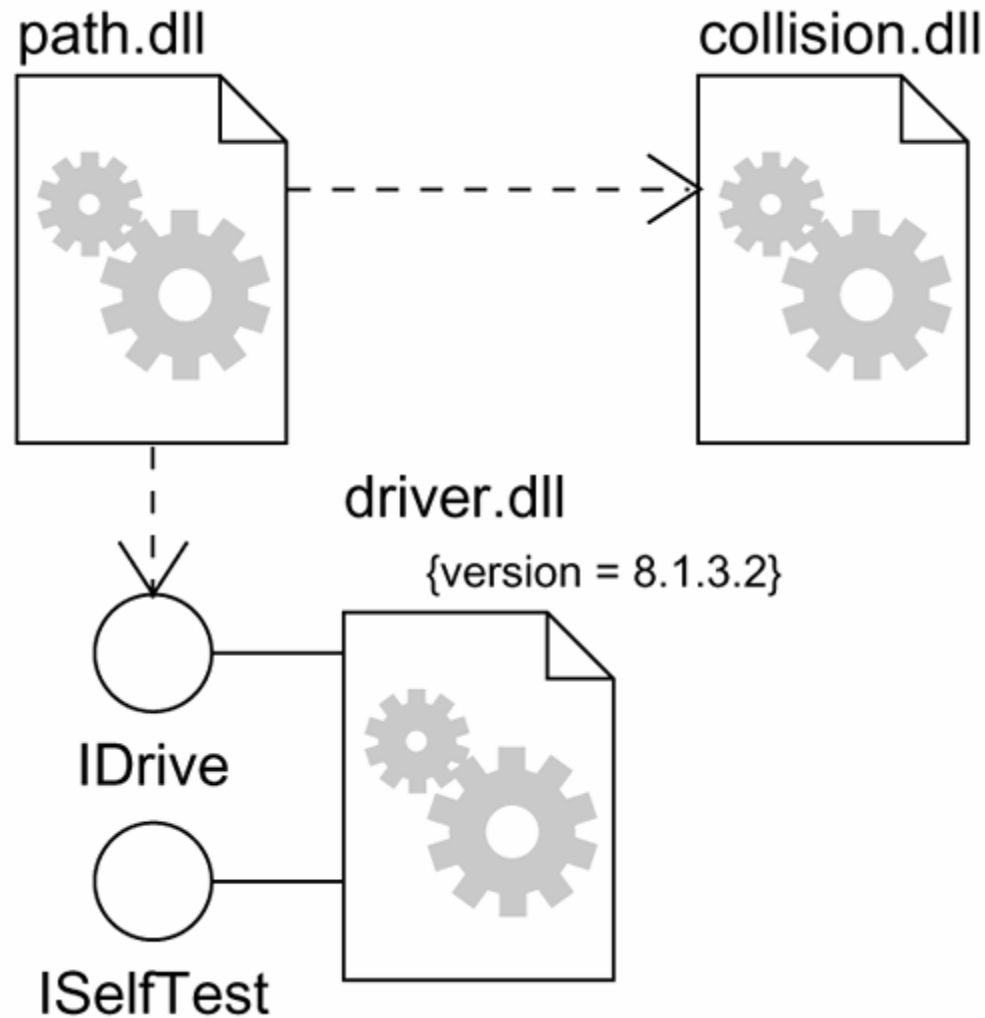
Component Diagram

- Menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem
- Fokus ke komponen sistem
- Digunakan untuk memodelkan
 - Source code
 - Komponen executable yang dilepas ke user
 - Basis data secara fisik
 - Sistem yang harus beradaptasi dengan sistem lain

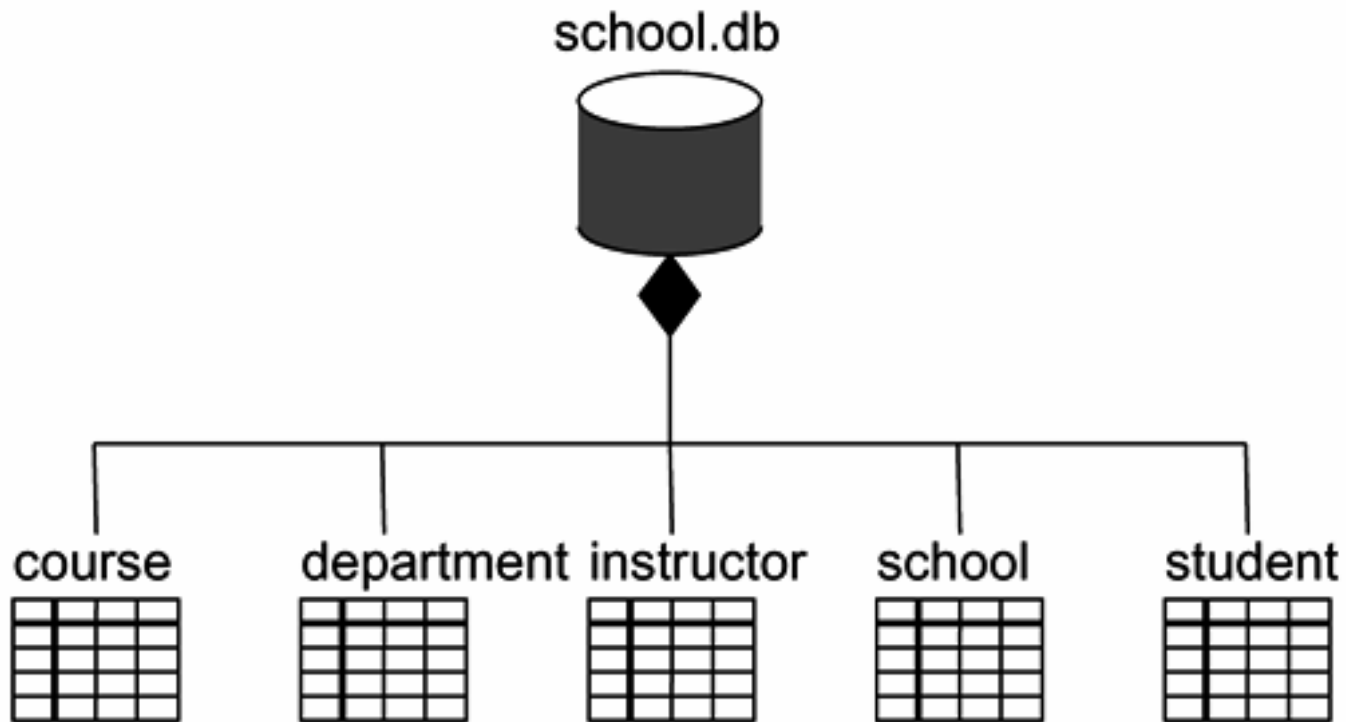
Component Diagram-Source Code



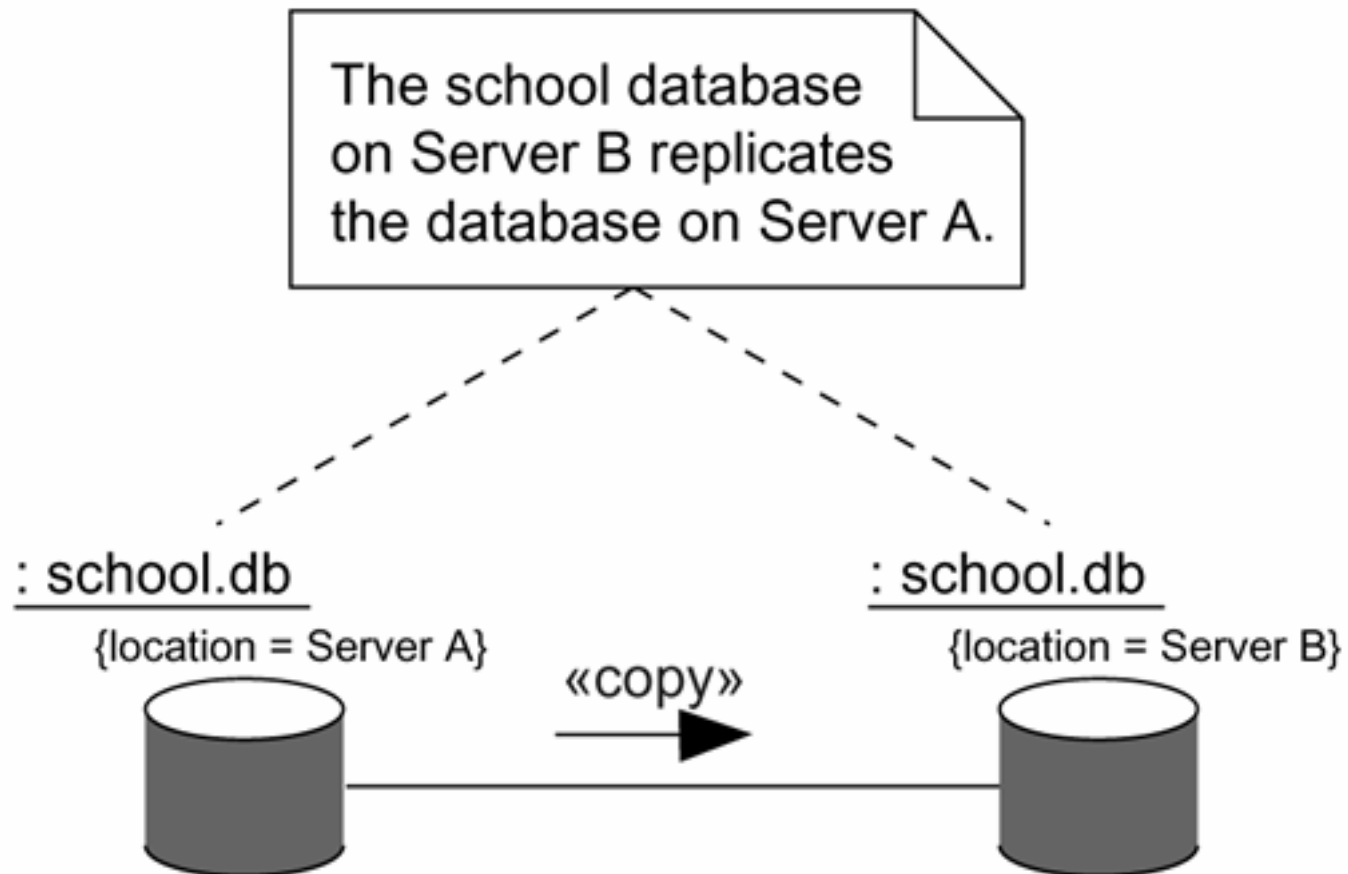
Component Diagram-Komponenten Executable



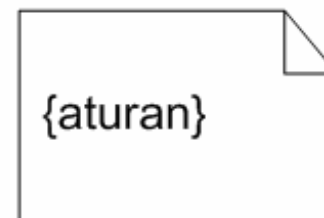
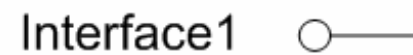
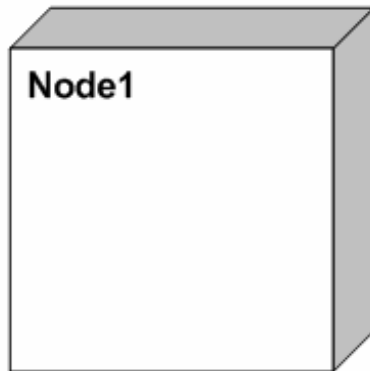
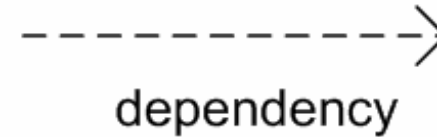
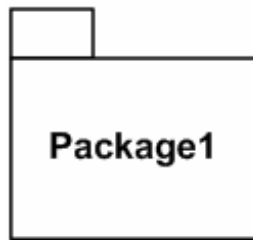
Component Diagram-Basis Data Fisik



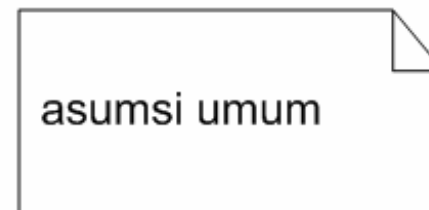
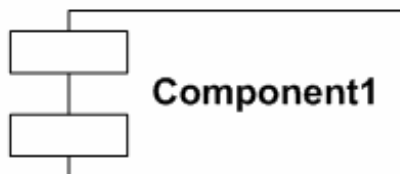
Component Diagram-Adaptasi Sistem



Component Diagram-Simbol



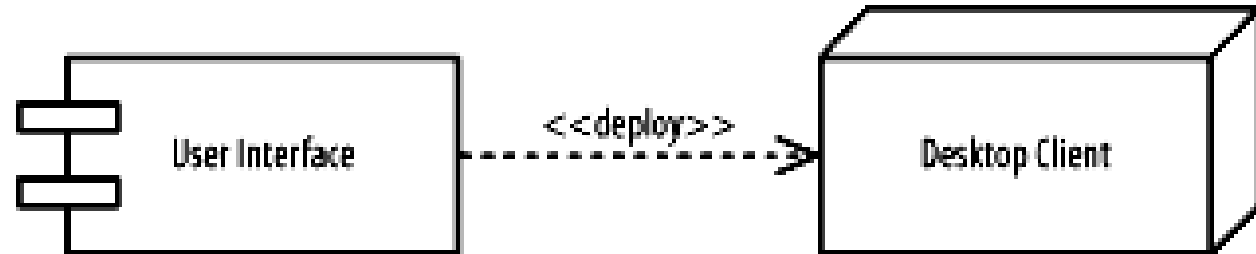
constraint



note

Component Diagram-Symbol

■ Node



- Sumber daya yang digunakan pada saat aplikasi dijalankan, misal:

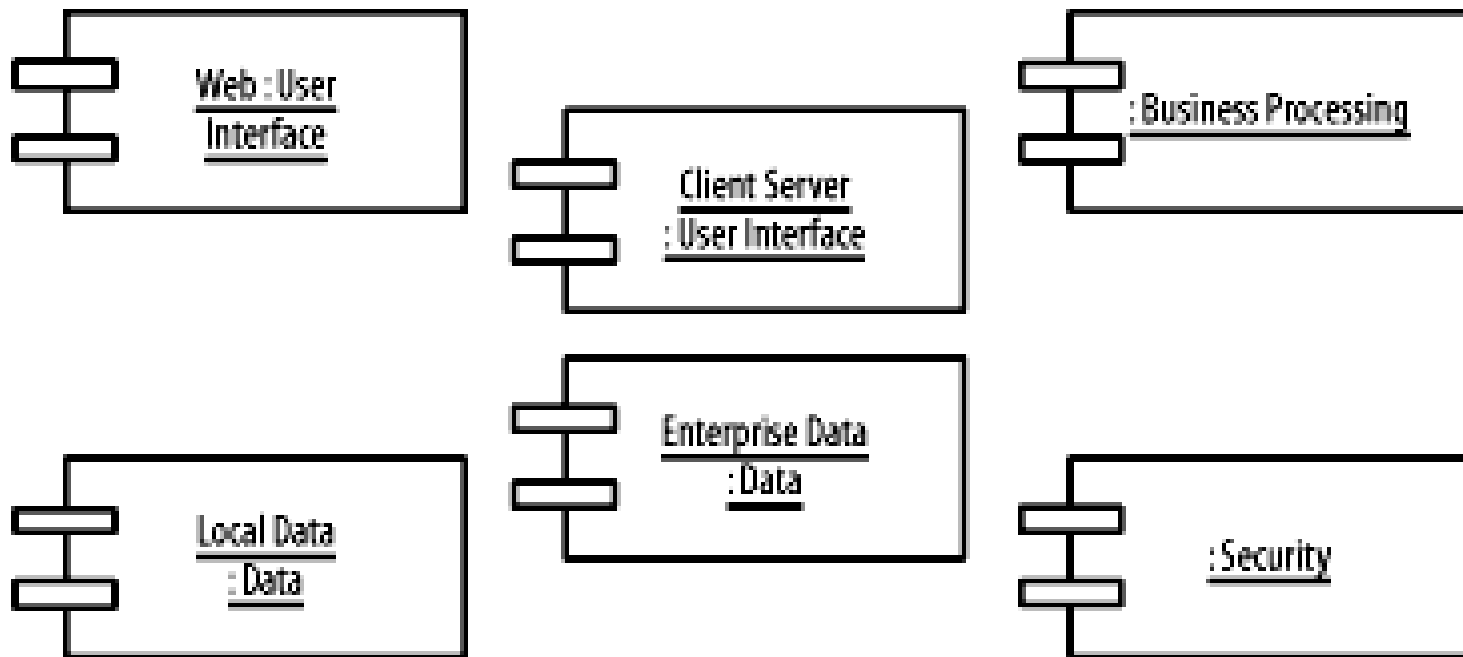
- Desktop client (tempat user interface dijalankan)
- Printer
- Business Process Server (tempat komponen business process dijalankan)
- Database server (tempat komponen data dijalankan)

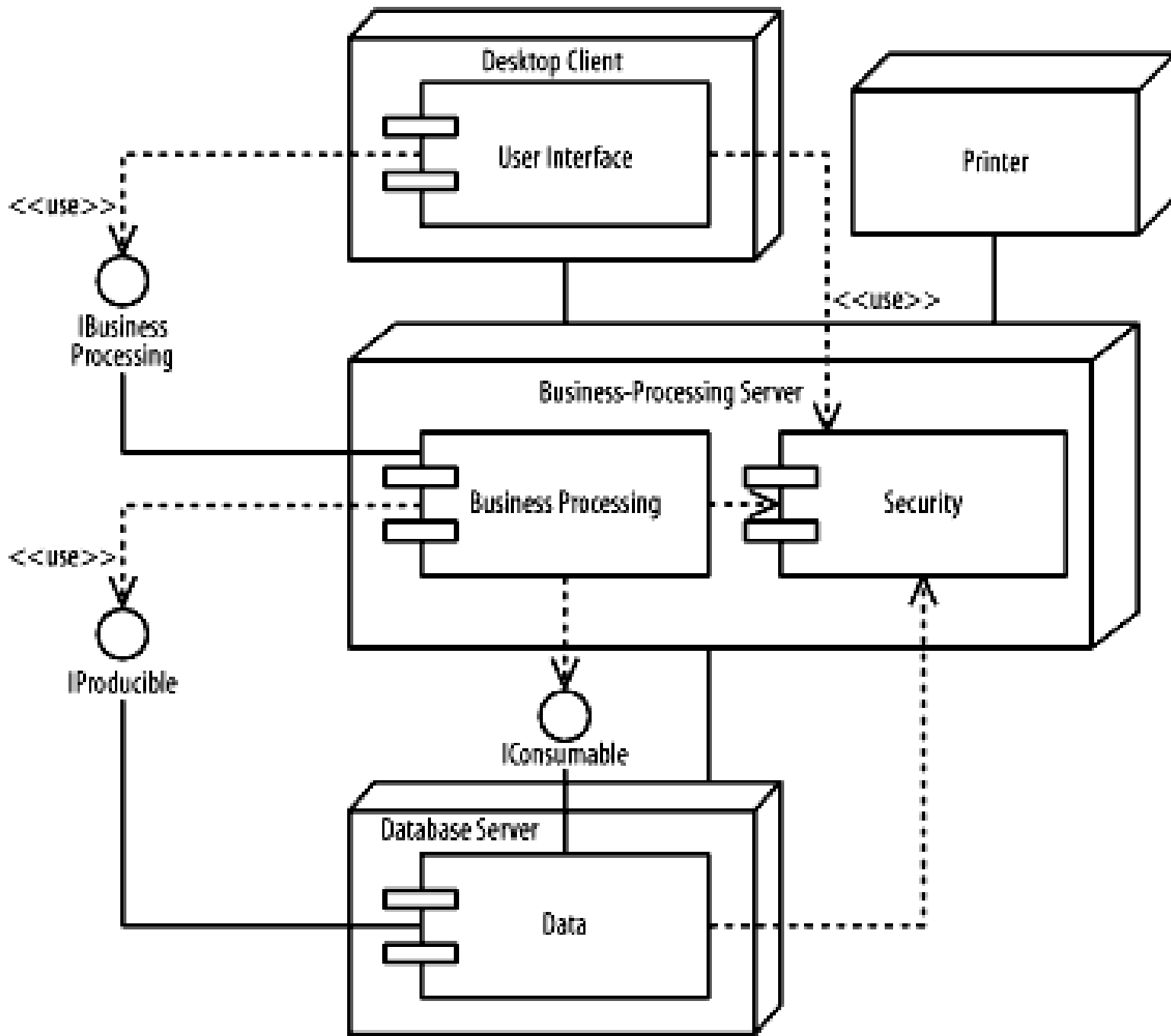
■ Dependency

- Menunjukkan asosiasi antar komponen, misalnya <<use>> dan <<deploy>>

Component Diagram

- Komponen pada Sistem Manajemen Proyek
 - User Interface
 - Business Processing
 - Data
 - Security

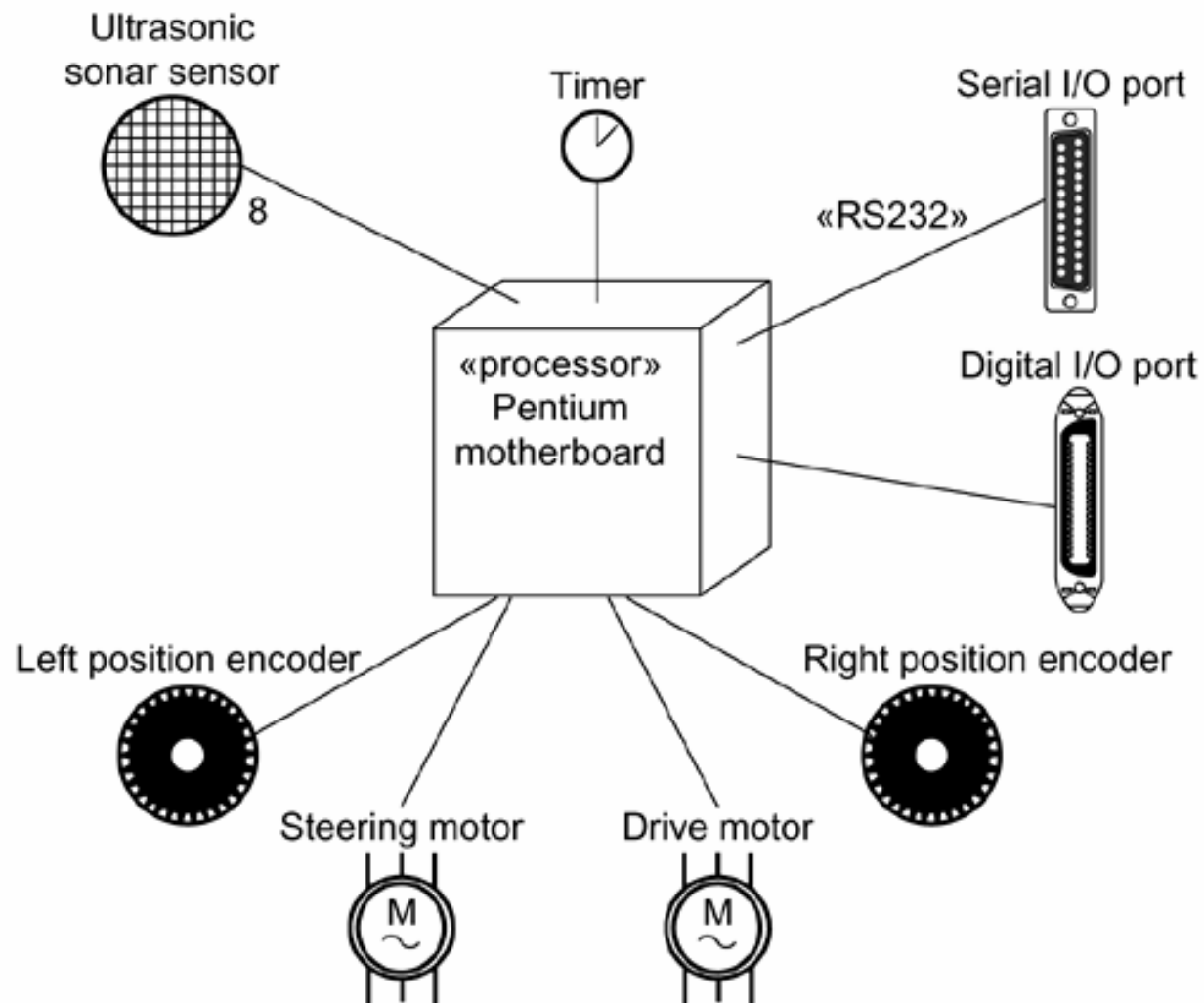




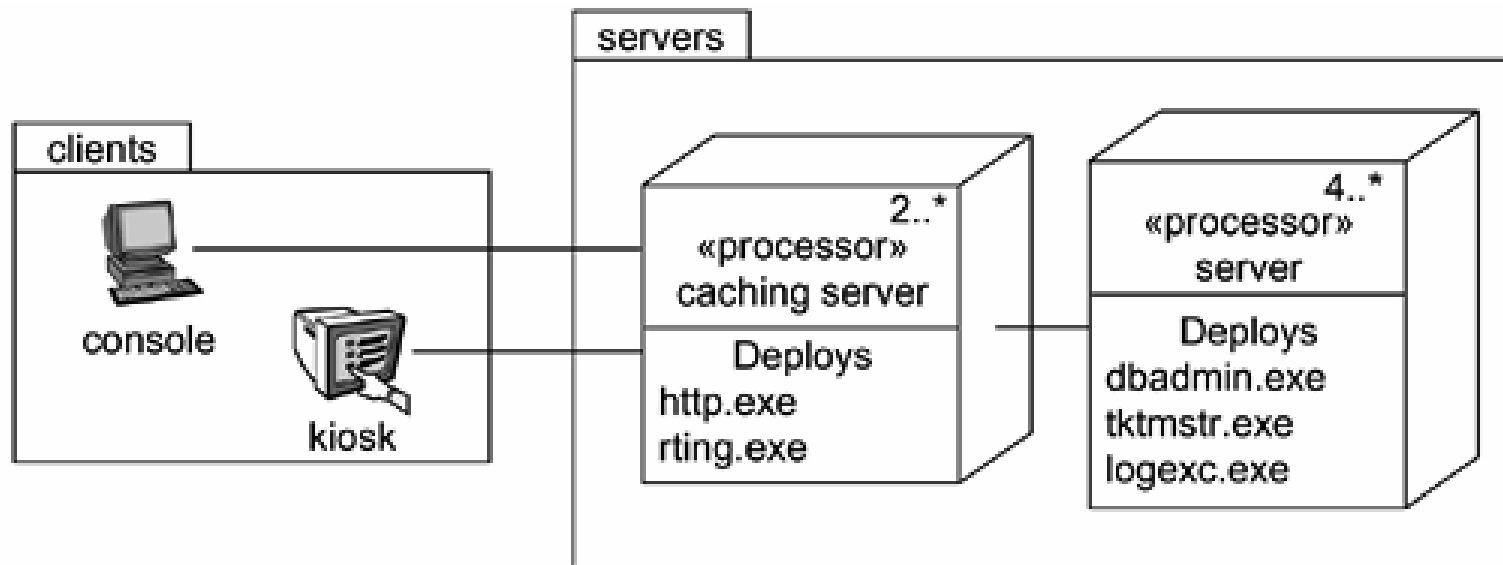
Deployment Diagram

- Menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi
- Digunakan untuk memodelkan
 - Sistem tambahan (*embedded system*)
 - Device, node, dan hardware
 - Sistem client/server
 - Sistem terdistribusi murni
 - Rekayasa ulang aplikasi

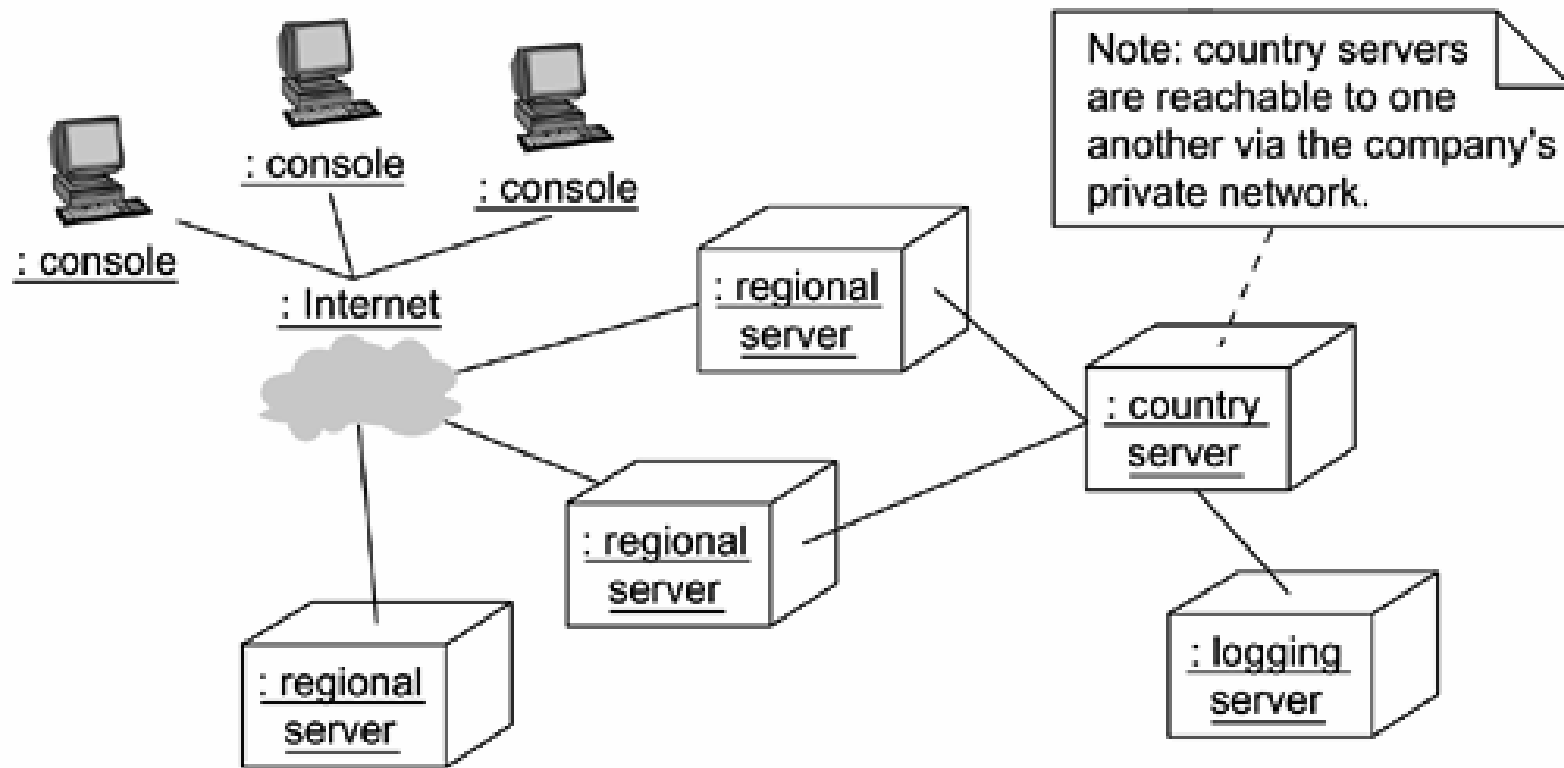
Deployment Diagram-Embedded System



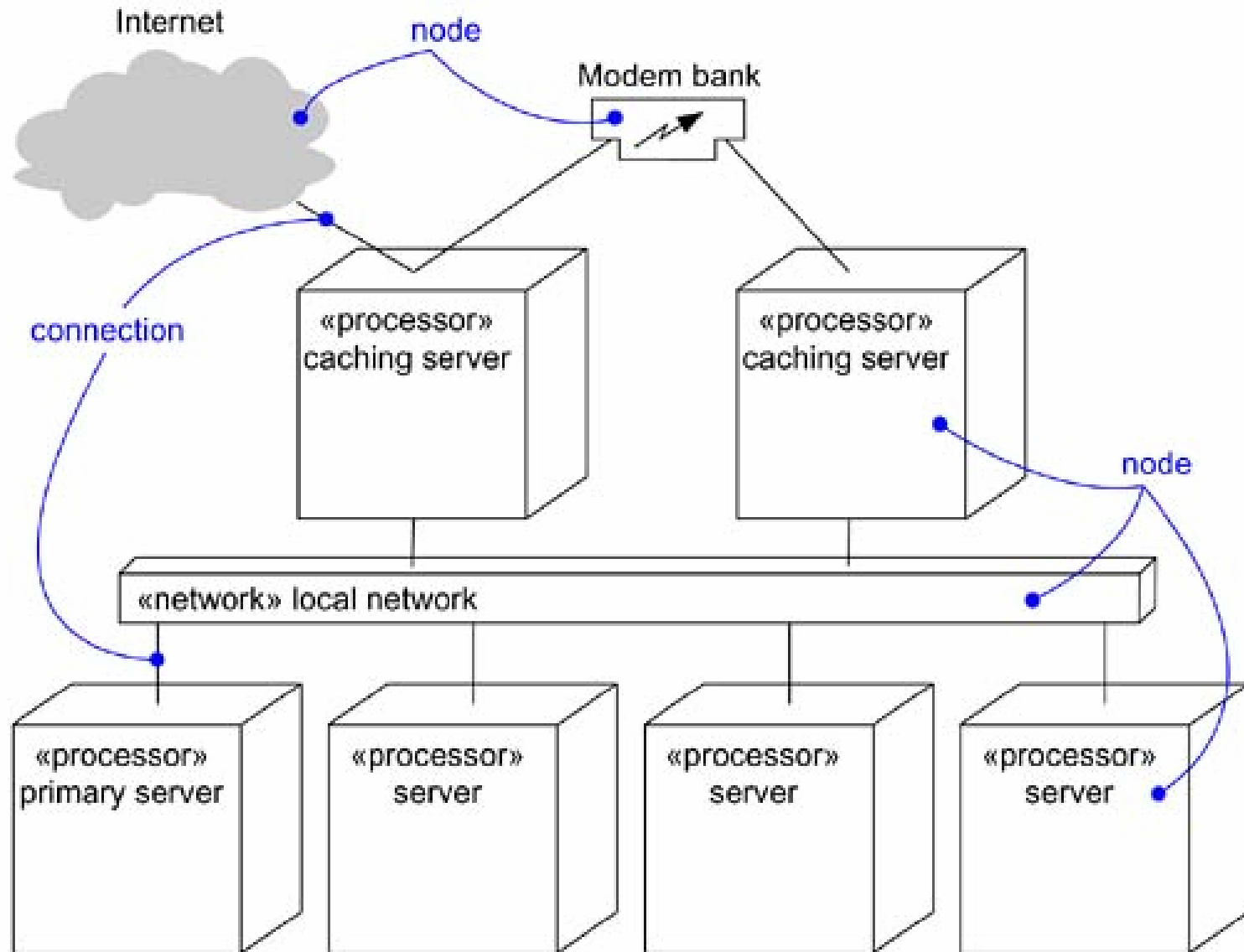
Deployment Diagram-Client/Server



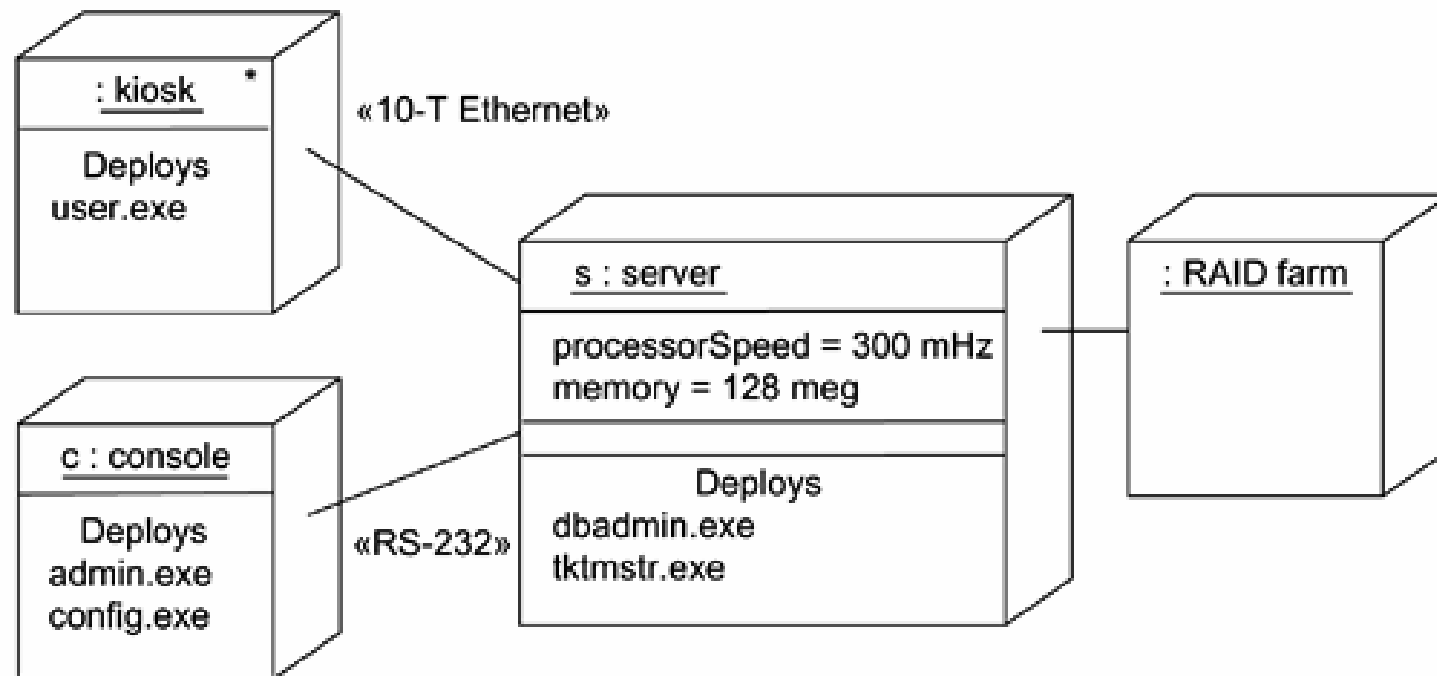
Deployment Diagram-Distributed System

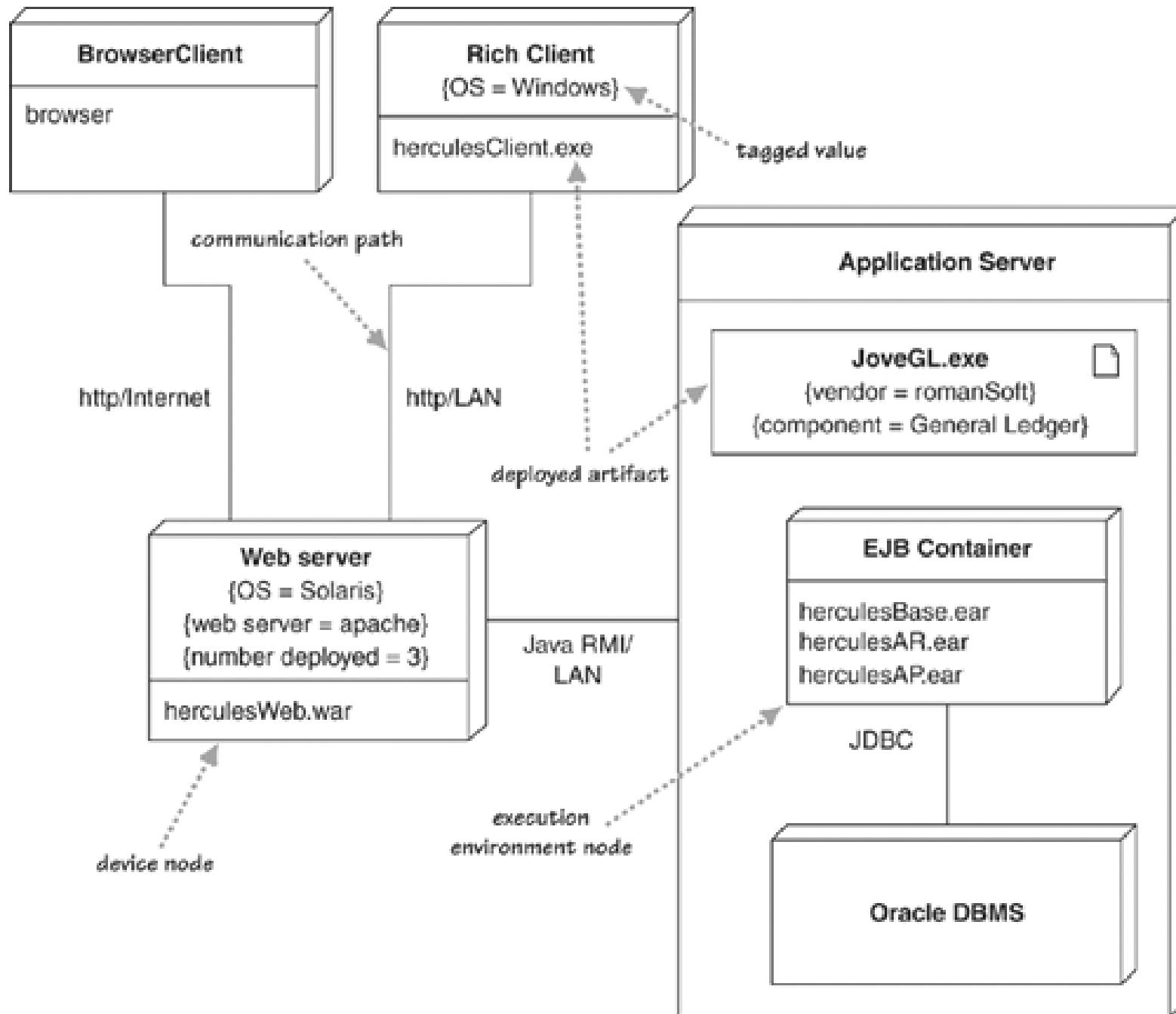


Deployment Diagram-Distributed System



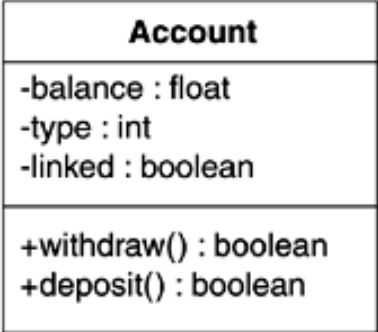
Deployment Diagram-Distributed System





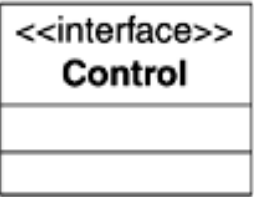
Implementasi Kelas dari UML

Dari Diagram Kelas

Java	UML
<pre>public class Account { private float balance; private int type; private boolean linked; public boolean withdraw(float amount) { } public boolean deposit(float amount) { } }</pre>	 <p>The UML class diagram for the Account class is shown in a rectangular box. It is divided into three horizontal compartments. The top compartment is the class name 'Account'. The middle compartment lists the private attributes: '-balance : float', '-type : int', and '-linked : boolean'. The bottom compartment lists the public methods: '+withdraw() : boolean' and '+deposit() : boolean'.</p>

Implementasi Interface dari UML

Dari Diagram Kelas

Java	UML
<pre>public interface Control{ ... }</pre>	

Java	UML
<pre>public interface Control{ ... }</pre>	