



Kuliah

***Rekayasa Perangkat Lunak
(Software Engineering)***

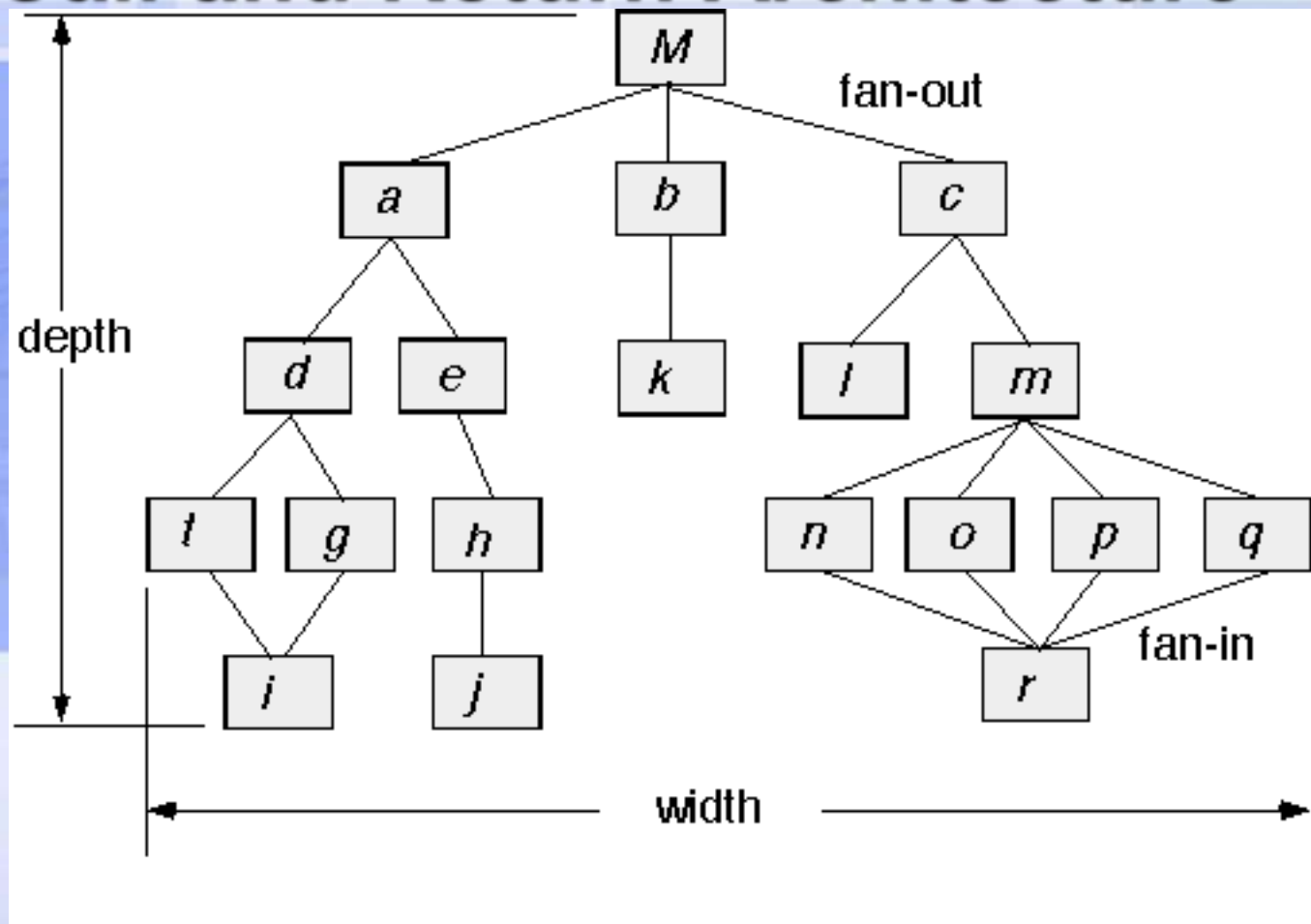
Bagian 2

Software Engineering: A Practitioner's Approach, 6/e

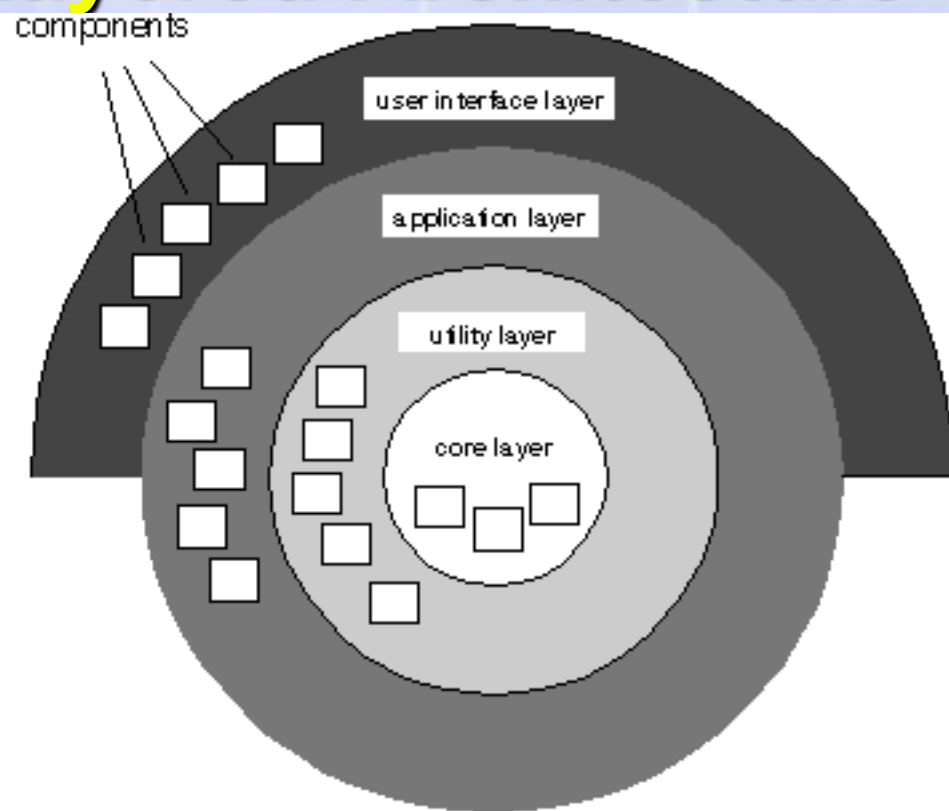
Bab 10

Desain Arsitektur

Call and Return Architecture



Layered Architecture



Pattern Arsitektural



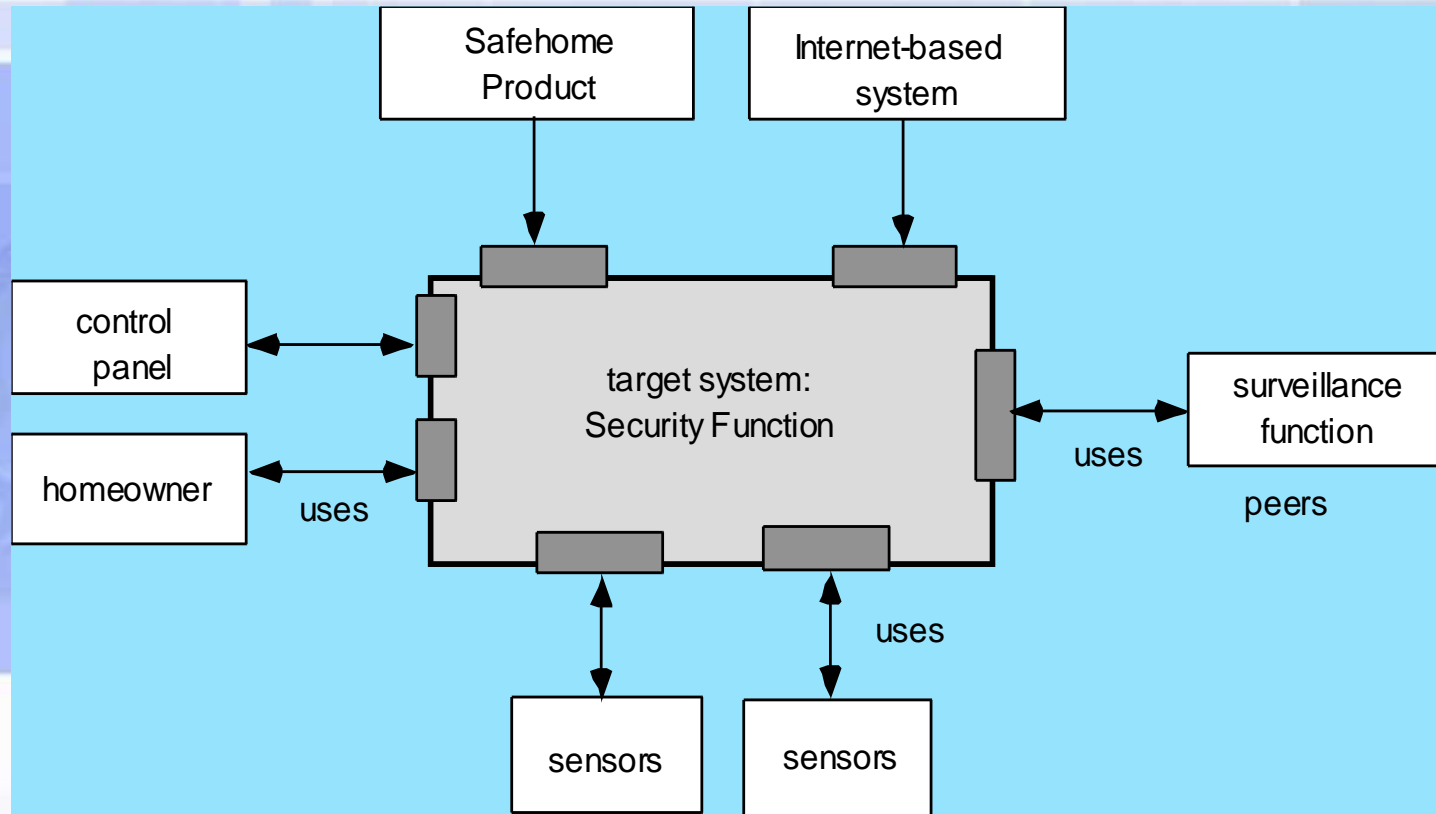
- **Concurrency**—aplikasi harus menangani banyak tugas dalam pola yang mensimulasikan paralelisasi
 - *operating system process management* pattern
 - *task scheduler* pattern
- **Persistence**—Data ada jika dia bertahan setelah eksekusi proses yang membuatnya. Ada dua pattern umum ::
 - *database management system* pattern yang menerapkan penyimpanan dan pengambilan dari DBMS kepada arsitektur aplikasi
 - *application level persistence* pattern yang membangun fitur persistence pada arsitektur aplikasi
- **Distribution**— pola dimana sistem atau komponen2x di antaranya berkomunikasi dalam lingkungan terdistribusi
 - *broker* bertindak sebagai orang di tengah antara komponen klient dan komponen server.



Desain Arsitektur

- PL harus ditempatkan pada konteks
 - Desain harus menentukan entitas eksternal (sistem lain, piranti, orang) dimana PL berinteraksi dengannya
- Sekumpulan arsitektur archetypes harus diidentifikasi
 - *archetype* adalah abstraksi (mirip dengan class) yang menampilkan satu elemen dari perilaku sistem
- Desainer menentukan struktur sistem dengan memilih komponen PL yang mengimplementasi masing2x archetype

Architectural Context



Archetypes

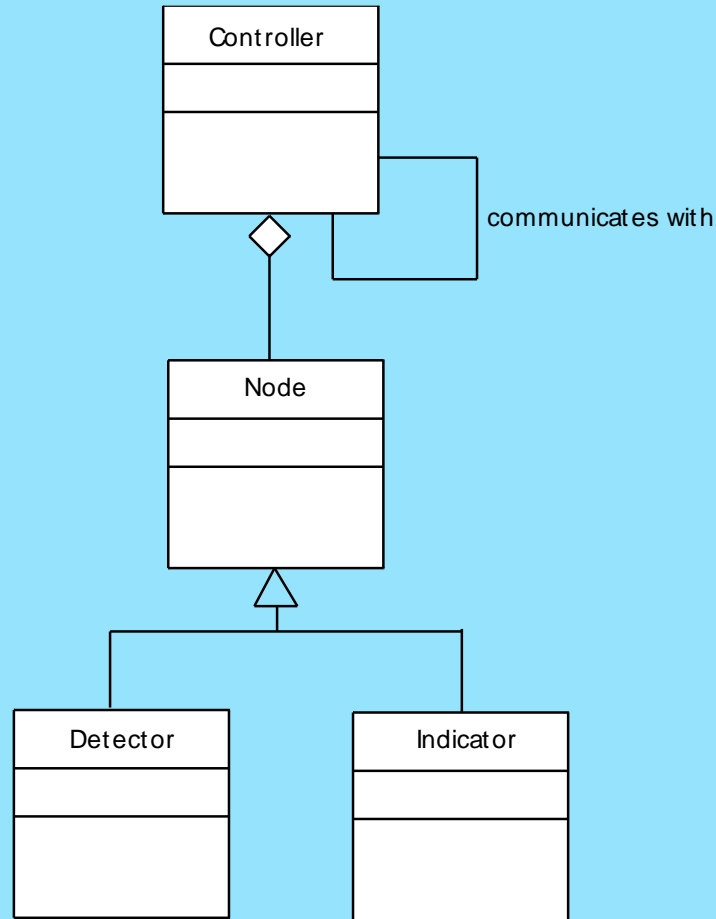
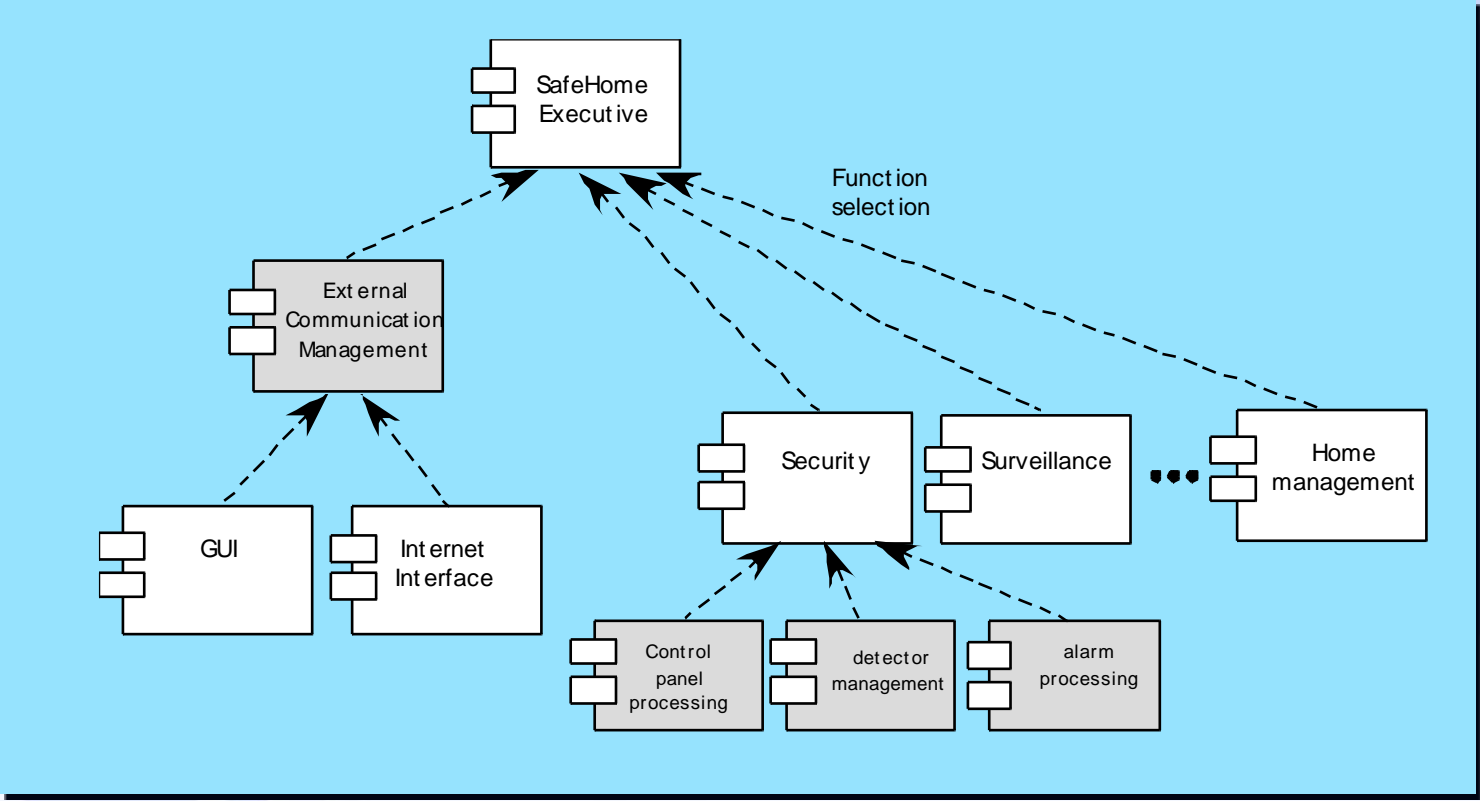


Figure 10.7 UML relationships for SafeHome security function archetypes (adapted from [BOS00])

Component Structure



Refined Component Structure

