

Collaboration Diagram

Jenis aksi

Dalam UML, kita dapat memodelkan beberapa jenis aksi, yaitu:

- **Call** : memanggil operasi yang ada pada object, object mungkin mengirim ke dirinya sendiri, menghasilkan pemanggilan lokal dari operasi.
- **Return** : mengembalikan nilai dari caller
- **Send** : mengirimkan sinyal ke object
- **Create** : membuat sebuah object
- **Destroy** : mematikan sebuah object, object mungkin saja mematikan dirinya sendiri.

Interaction diagram digunakan ketika kita ingin melihat kelakuan dari beberapa object dalam use case tunggal. Diagram ini baik saat menunjukkan kolaborasi diantara object-object, namun kurang baik dalam mendefinisikan behavior. Ada dua macam Interaction Diagram yaitu : Sequence Diagram dan Collaboration Diagram.

Collaboration Diagram

- CD merupakan cara alternatif untuk menggambarkan suatu skenario dari sistem
- CD juga menggambarkan interaksi antar objek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message.
- Setiap message memiliki sequence number.
- Collaboration Diagram berisi :
 - Obyek, yang digambarkan dalam segi empat/rectangle
 - Hubungan/Link antar obyek, diperlihatkan sebagai garis yang menghubungkan dengan obyek lain.
 - Pesan/Message ditunjukkan sebagai teks dan panah yang mengirim pesan ke penerima pesan

Collaboration vs Sequence Diagram

- **Collaboration Diagram**

- Menunjukkan hubungan disamping interaksi
- Lebih baik untuk memvisualisasikan pattern of collaboration
- Lebih baik untuk memvisualisasikan semua efek dari object yang diberikan
- Lebih mudah digunakan untuk sesi brainstorming/fase desain

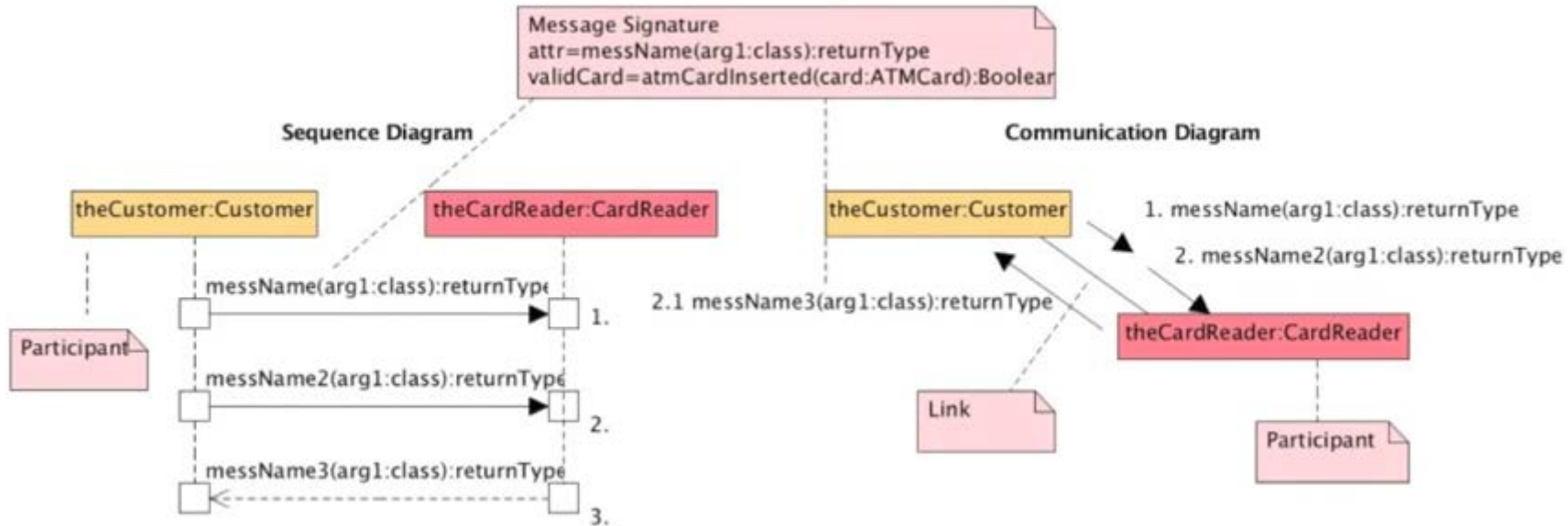
- **Sequence Diagram**

- Menunjukkan urutan message secara eksplisit
- Lebih baik dalam memvisualisasikan keseluruhan aliran
- Lebih baik dalam memvisualisasikan spesifikasi yang real time dan skenario yang kompleks
- Cocok untuk fase analisa sistem

Komponen CD

- Collaboration Diagram mendeskripsikan pola interaksi antar object. Diagram ini menunjukkan object-object yang terlibat dalam interaksi melalui link dan message yang dikirim antar object
- **Object** dapat direpresentasikan:
 - a) NamaObject:>NamaClass
 - b) NamaObject
 - c) :NamaClass
- **Link** adalah hubungan antar object yang dapat digunakan untuk mengirim message. Link digambarkan dengan garis solid antar dua object
- **Message** : pengertiannya sama dengan message pada sequence diagram

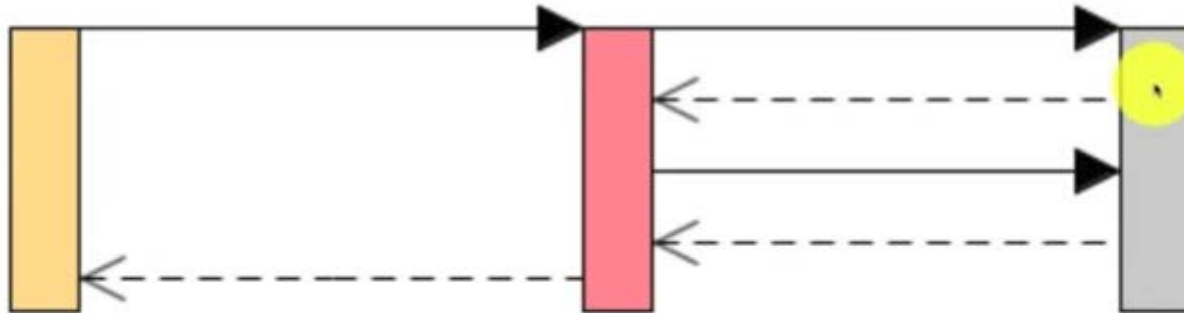
Sequence Diagram vs Collaboration Diagram



Sequence Diagram vs Collaboration Diagram

Nested Method

Sequence Diagram
Nested Method

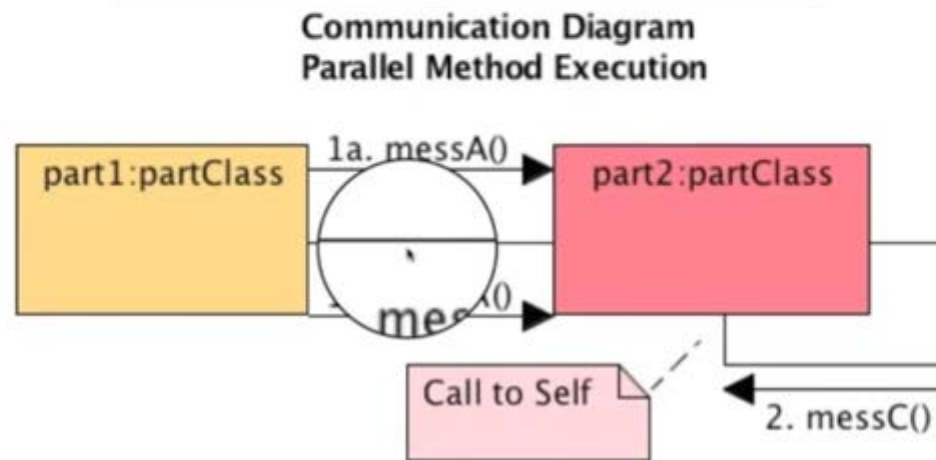
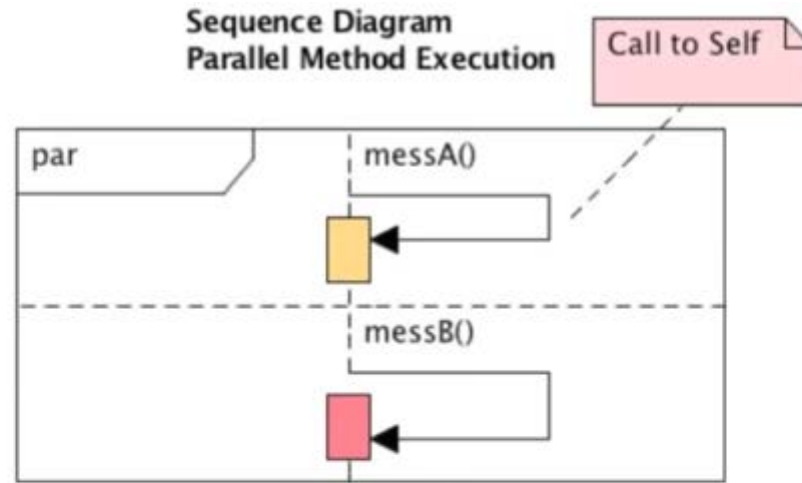


Communication Diagram
Nested Message



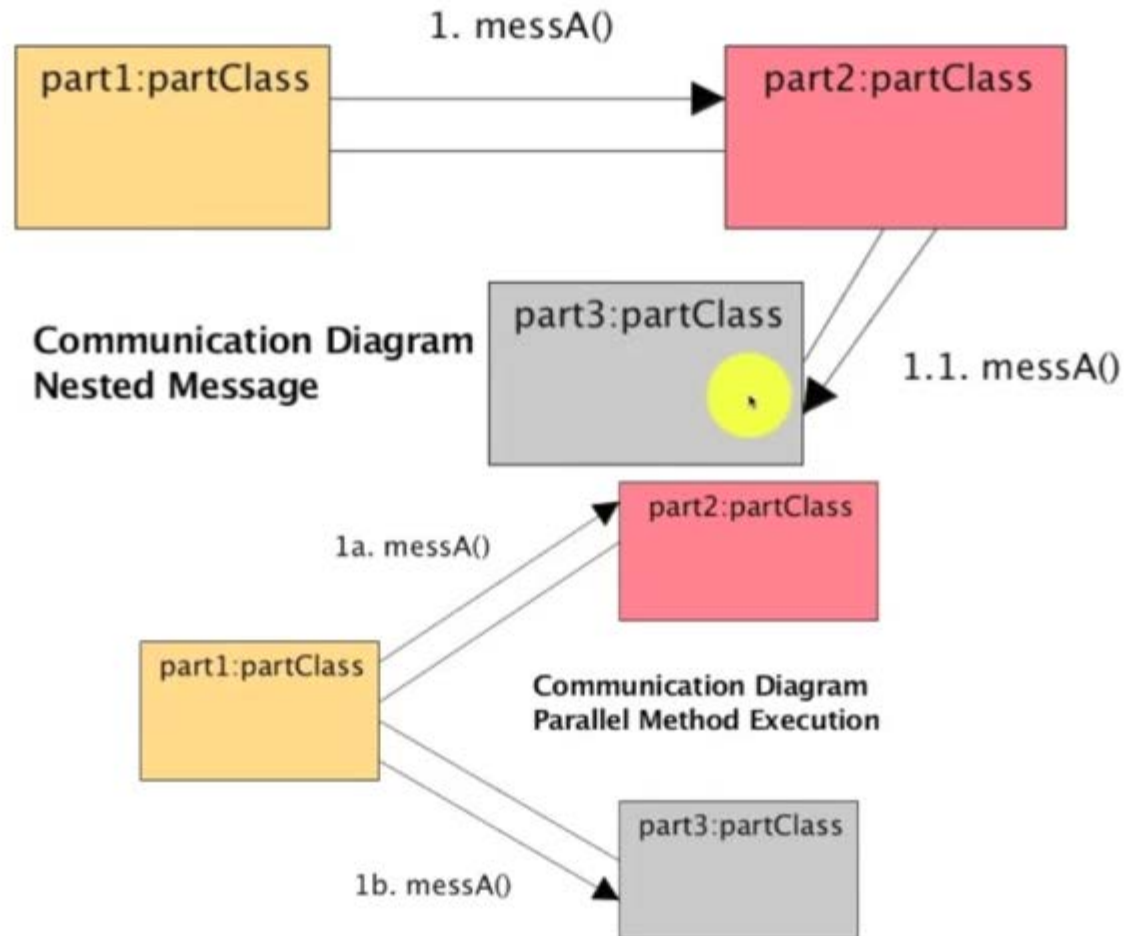
Sequence Diagram vs Collaboration Diagram

Parallel Method Execution



Sequence Diagram vs Collaboration Diagram

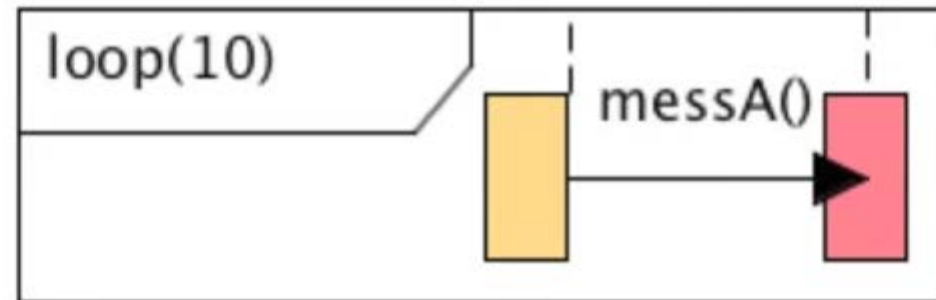
Nested Message



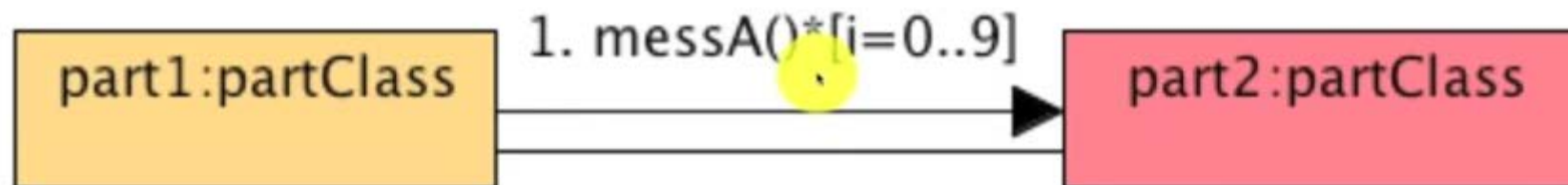
Sequence Diagram vs Collaboration Diagram

Execute Multiple Times

Sequence Diagram
Execute Multiple Times

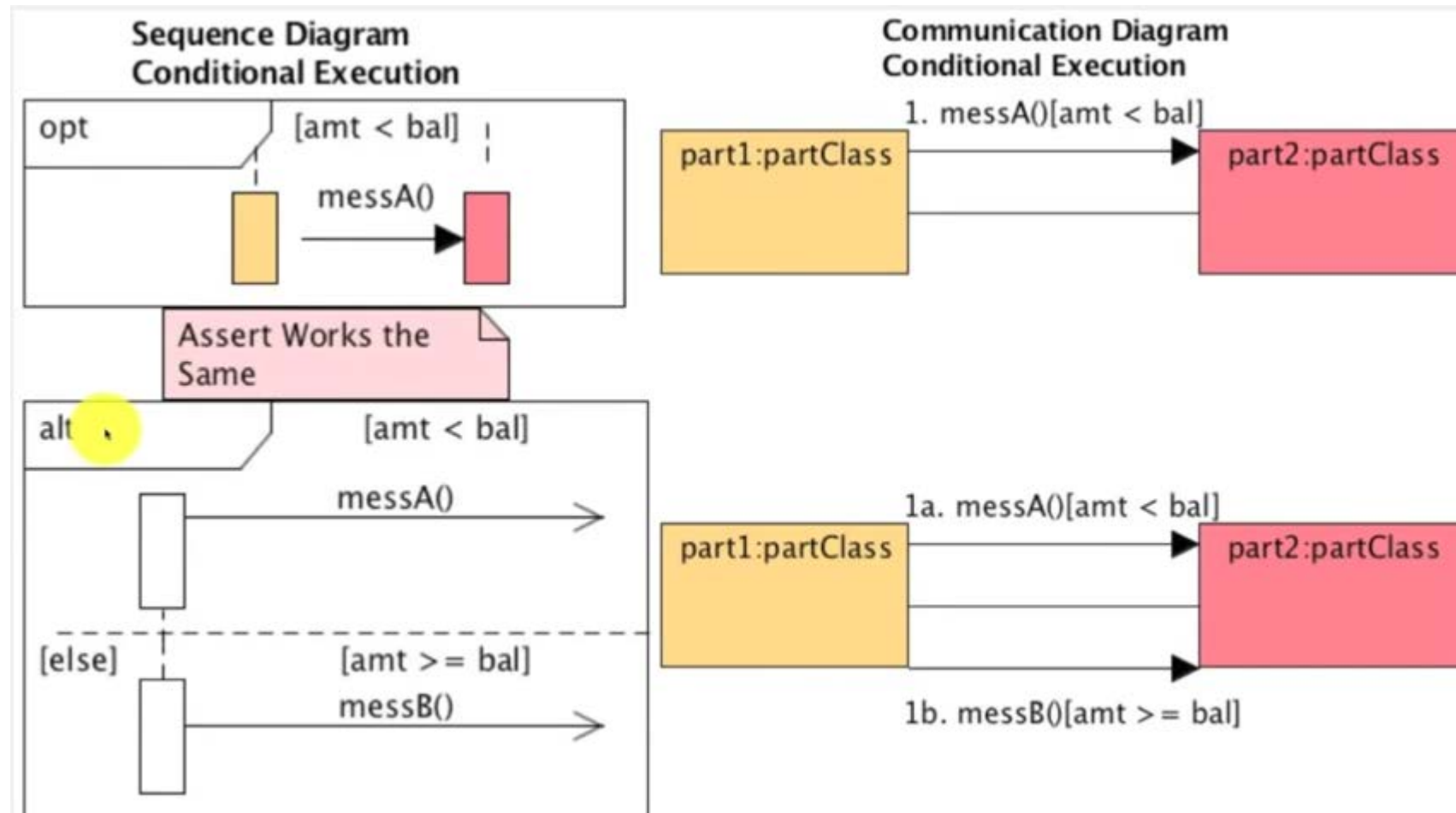


Communication Diagram
Execute Multiple Times



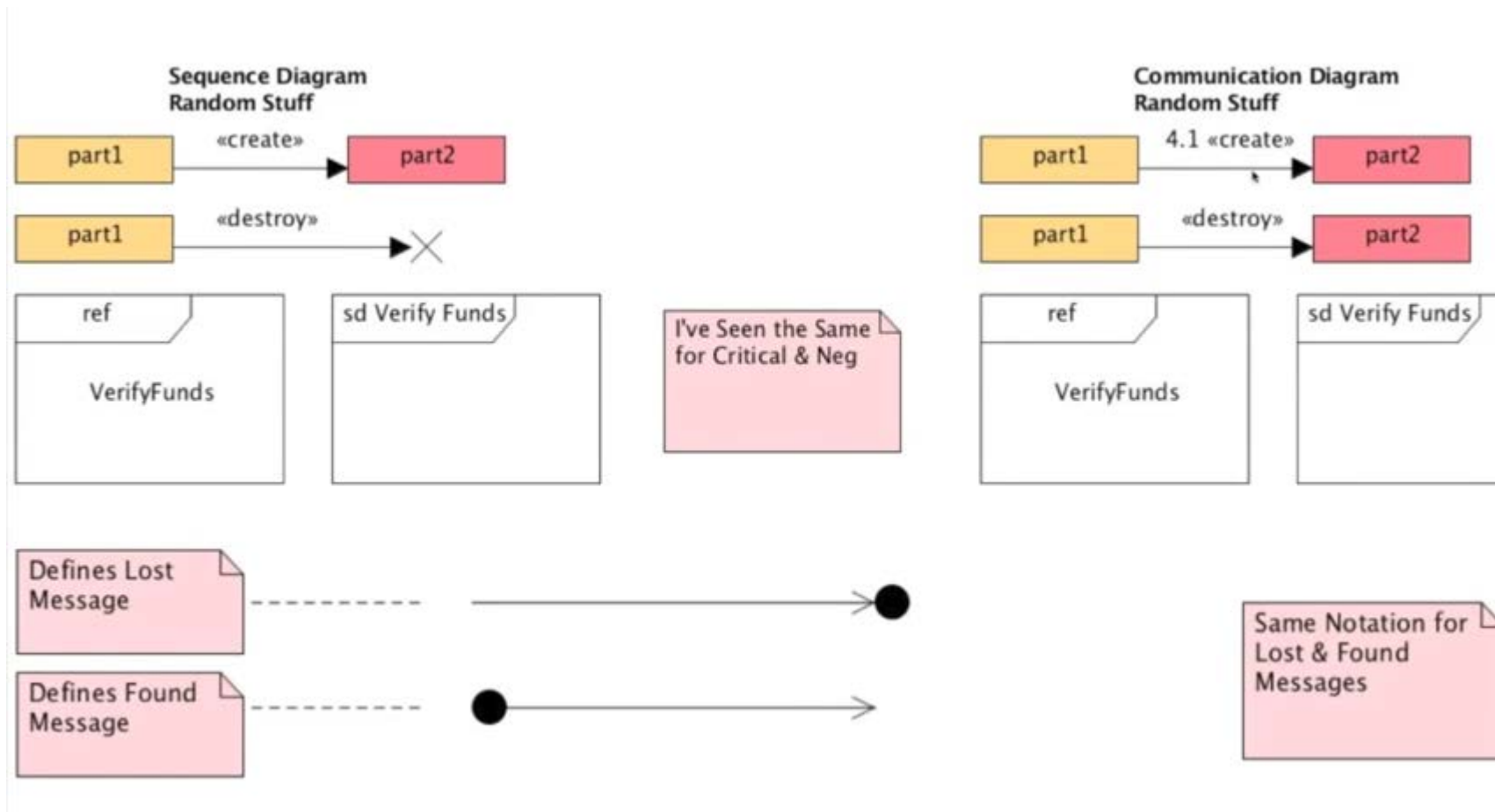
Sequence Diagram vs Collaboration Diagram

Conditional Execution



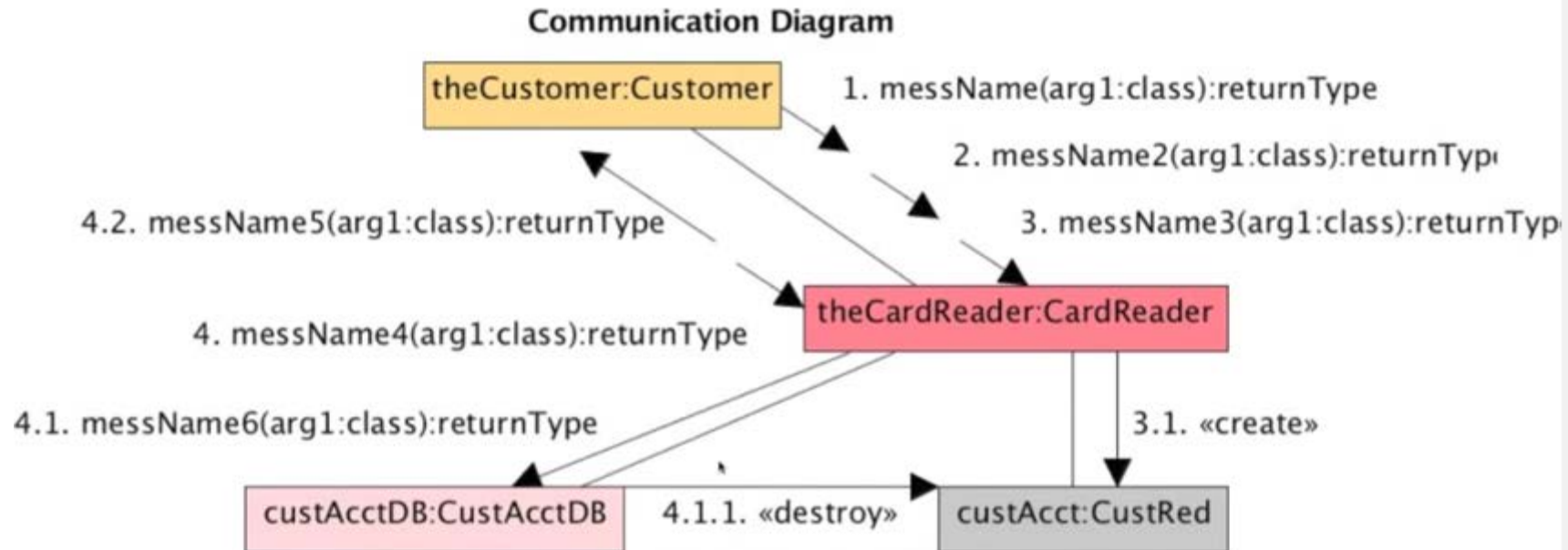
Sequence Diagram vs Collaboration Diagram

Random Stuff



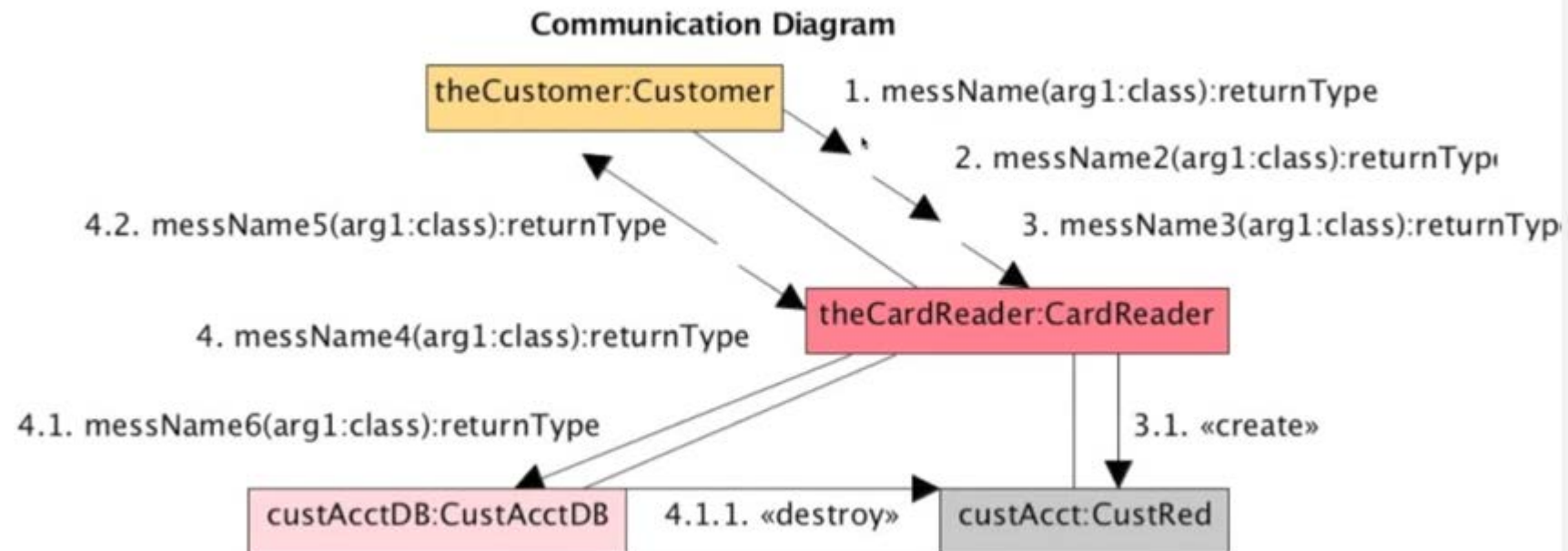
Sequence Diagram

Contoh(1)



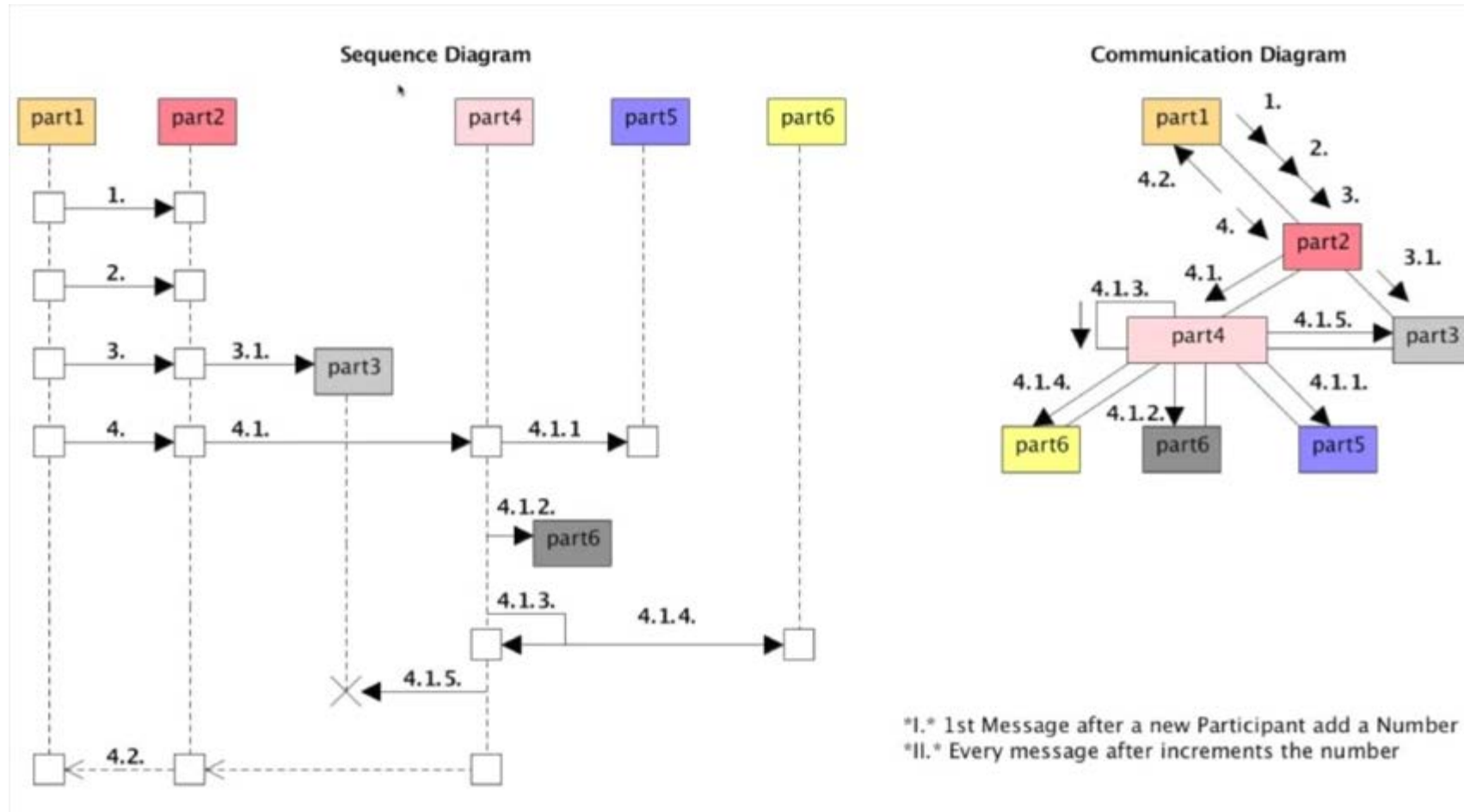
Collaboration Diagram

Contoh(1)



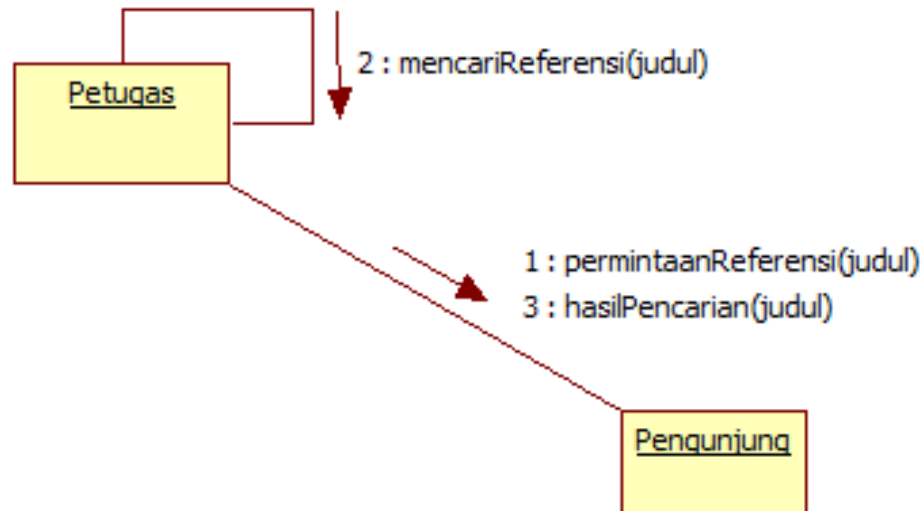
Sequence Diagram vs Collaboration Diagram

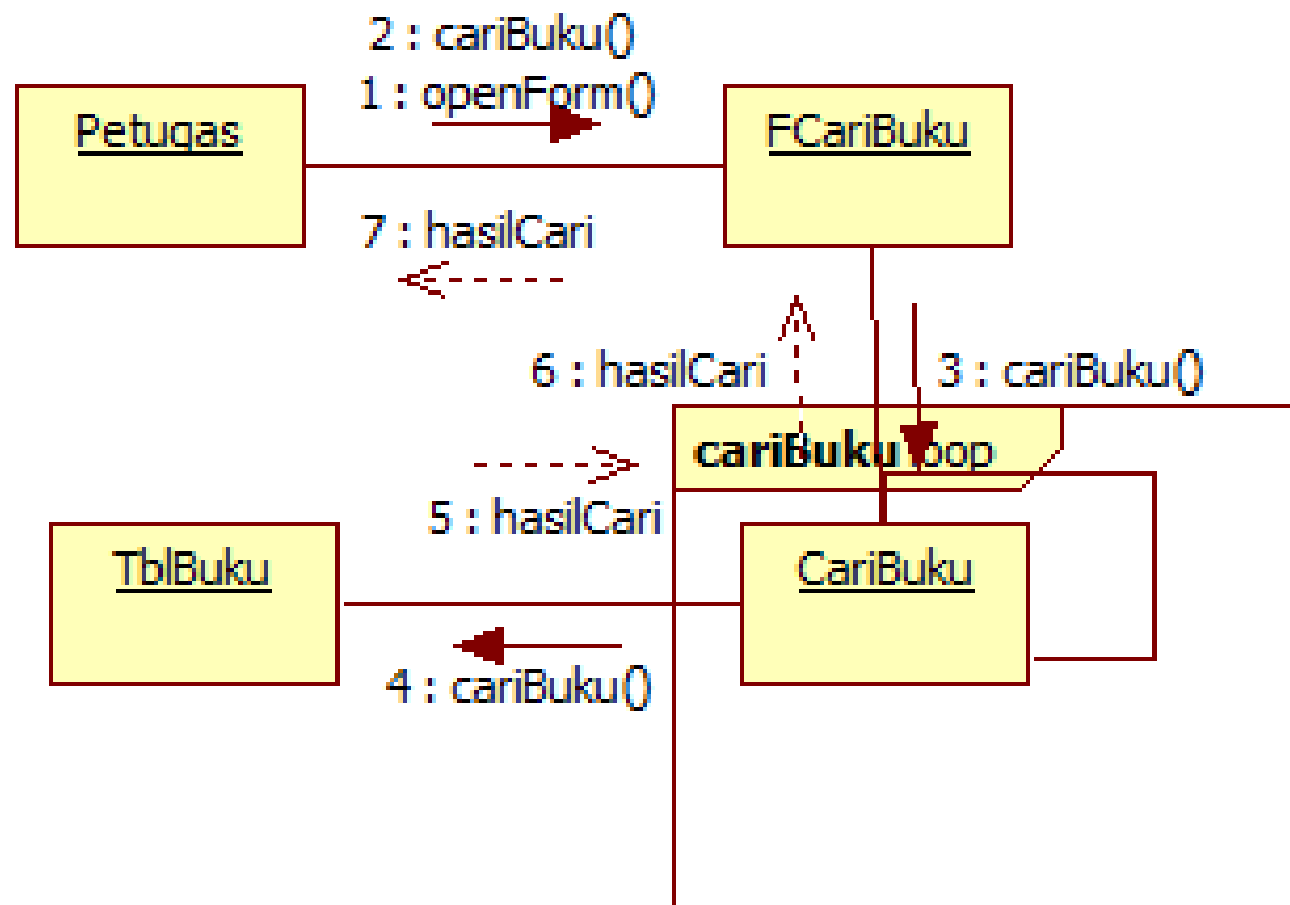
Perbandingan



Latihan 1

- Di sebuah perpustakaan, petugas perpustakaan mengambil permintaan referensi dari pengunjung, mencari informasi referensi tersebut dari database, kemudian memberi jawaban kepada pengunjung
- Bagaimana collaboration diagram-nya ?





Latihan 2

- Di sebuah perpustakaan, pengunjung akan meminjam buku, petugas perpus mengambil data peminjam dari pengunjung, menyimpan data peminjaman ke database
- Bagaimana collaboration diagram-nya ?

Latihan 3

- Berikut ini adalah kegiatan penjualan Buku “SiBook”
 - Proses penjualan dimulai saat seorang *customer* menanyakan tentang stok buku kepada admin.
 - Selanjutnya admin akan mengecek di sistem dan memberitahukan hasilnya kepada *customer*
 - Jika stok buku tidak ada, admin akan bertanya kepada *customer*, apakah diganti dengan buku yang lain
 - Apabila stok buku ada maka transaksi dilanjutkan, dan petugas akan menginputkan data buku yang dipesan *customer*
 - Admin melakukan perhitungan pembayaran dan memberikan nota pembayaran
- Bagaimana collaboration diagram-nya ?