

## **10.5 Implementasi Groupware**

### **10.5.1 Umpan balik dan jeda pada jaringan**

Sistem groupware biasanya membutuhkan beberapa komputer yang terhubung melalui jaringan. Jika pengulangan umpan balik mencakup transmisi melalui jaringan, akan sulit mencapai waktu respon yang dapat diterima. Untuk melihat apa yang terjadi bila user menuliskan sebuah karakter :

1. Aplikasi user mengambil kejadian event dari window manajer
2. User memanggil sistem operasi ...
3. Yang mengirimkan pesan melalui jaringan, sering melalui serangkaian protokol.
4. Pesan akan diterima oleh sistem operasi pada remote machine
5. Yang memberikan remote application untuk memproses
6. 6-8 mengulangi proses yang sama pada langkah (2-4)
7. 9 umpan balik akan diberikan pada layar user

Pada proses diatas membutuhkan sedikitnya dua pesan atau lebih pada jaringan ditambah empat switch penghubung dan empat protokol. Rata-rata waktu yang dibutuhkan 1-2 detik untuk tiap proses.

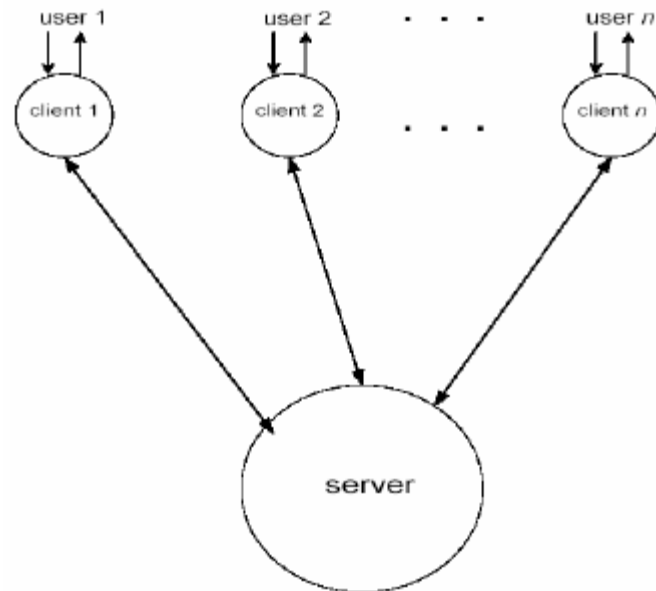
### **10.5.2 Arsitektur groupware**

Terdapat dua arsitektur groupware yaitu :

1. Centralized (client-server architecture) Arsitektur terpusat memiliki dua bentuk yaitu :
  - a. Client-server, setiap workstation partisipan memiliki program minimal (client) yang menangani layar dan menerima input partisipan. Dalam aplikasi sebenarnya dijalankan oleh server yang bekerja pada komputer pusat dan menangani semua data aplikasi.
  - b. Master-slave, server bekerja pada salah satu workstation dan memasukkan client (user yang pertama meminta aplikasi bersama). Master menjadi gabungan server client dan slave menjadi client. User dari master akan memiliki respon yang cepat dibandingkan dengan user lainnya.
2. Replicated

Pada arsitektur ini masing-masing workstation user menjalankan salinan aplikasi. Salinan ini berkomunikasi dengan yang lain dan berusaha membuat struktur datanya konsisten dengan

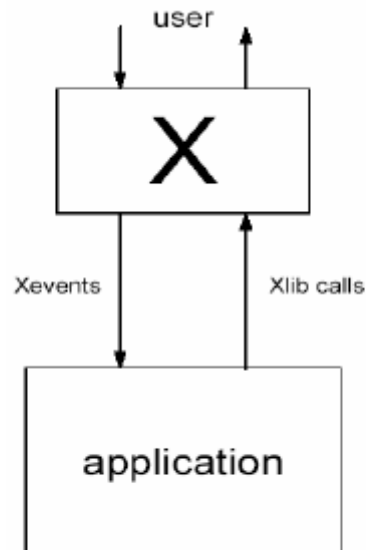
yang lain. Setiap replikasi menangani respon usernya masing-masing dan harus mengupdate layar dalam merespon pesan dari replikasi lainnya. Kelemahan arsitektur ini sulit untuk diprogram. Untuk mengatasi masalah tersebut dengan adanya rollback dari satu replika ke replika lain dan mengeksekusi ulang perintah. Keuntungan utamanya pada umpan balik lokal.



**Gambar 10.7** Arsitektur Client Server

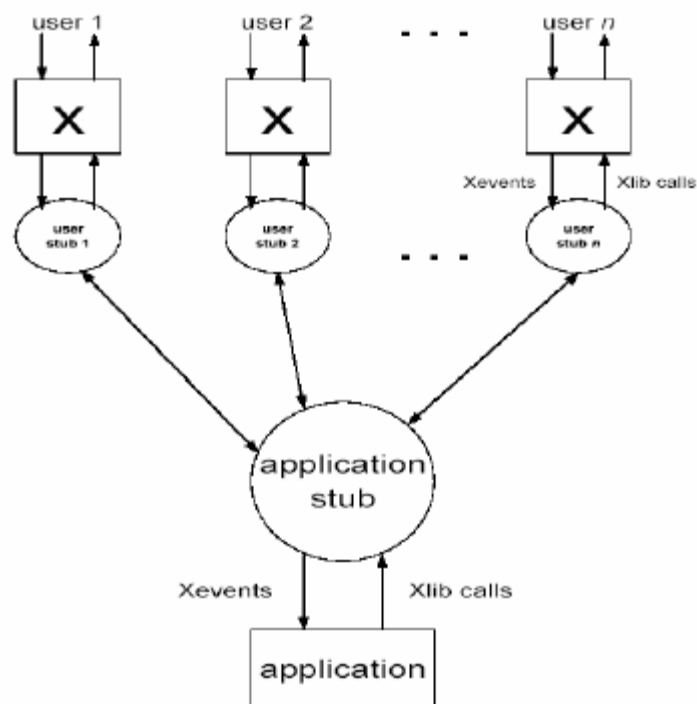
### 10.5.3 Shared windows architecture

Pada gambar di bawah ini merupakan aplikasi pada single user yang normal berinteraksi melalui window manager (misalnya X). Window manager bekerja bersama dengan menangkap panggilan call antara aplikasi dan X.



Gambar 10.8 Single user application

Ketika aplikasi mengirimkan panggilan grafik ke X, ia akan masuk ke potongan stub aplikasi khusus. Kemudian melewati call grafik ke stub user pada setiap workstation partisipan. Salinan dari X akan tereksekusi pada setiap workstation dan stub user akan melewati call grafik ke salinan lokal X. Secara bersamaan maka keystroke pengguna dan beberapa tindakan menyebabkan X melewati stub user kemudian melewati stub aplikasi ke aplikasi.



Gambar 10.9 Shared window architecture

#### **10.5.4 Feedthrough dan Network traffic**

Feedthrough ini merefleksikan tindakan seorang user pada layar user yang lain sehingga dapat mengurangi trafik suatu jaringan. Hanya sedikit jaringan yang mendukung pesan broadcast sehingga  $n$  partisipan  $n-1$  network message! Solusinya dengan meningkatkan granularitas dan mengurangi frekuensi umpan balik dengan mempertimbangkan timelines vs lalu lintas.

#### **10.5.5 Graphical toolkits**

Widget pada graphical toolkit atau window manager seperti: menu, tombol button, dialogue box, text dan graphic region berguna untuk membuat interface single user. Salah satu dari widget tersebut dapat digunakan untuk membentuk sistem groupware. Beberapa widget dapat menangani kontrol aplikasi, misal pada menu pop-up berikut :

```
sel = do_pop_up("new","open","save"."exit",0);
```

Secara fundamental fungsionalitas dari widget toolkit tidak mencukupi untuk groupware.

#### **10.5.6 Robustness dan skalabilitas**

Groupware merupakan sesuatu yang kompleks yang terdiri dari gabungan elemen jaringan, grafik dan lain sebagainya. Skala user yang besar, groupware perlu lebih sering dilakukan pengujian dan debugging untuk meminimalkan terjadinya kesalahan. Dibawah ada beberapa tip yang berhubungan dengan groupware, yaitu;

##### **1. Network atau server fail**

Masalah yang paling besar pada sistem client-server adalah jika terjadi tabrakan server, baik pada perangkat lunak maupun pada perangkat keras. Masalah tersebut bisa diatasi dengan memiliki multiserver dan salinan data sehingga server backup dapat mengambil alih setelah terjadi tabrakan pada server utama.

##### **2. Client Fail**

Kerusakan ini berupa kesalahan kode karena sangat kompleks. Bila menggunakan arsitektur client server pencegahannya adalah;

a. Robust

Server harus dapat mempertahankan client yang tabrakan dan kerusakan client tidak menyebabkan server *hang*.

b. Reconfigure

Server harus mendeteksi kesalahan yang terjadi pada client dan mengkonfigurasi ulang atas keseluruhan sistem.

c. Resynchronise

Ketika workstation / client me-recover maka server harus mengirimkan informasi yang cukup untuk dikirimkan ke client yang diperbaiki.

### 3. Kesalahan dalam pemrograman

Beberapa aplikasi yang crash tidak menyebabkan aplikasinya menjadi rusak dan kemungkinan akan menyebabkan lebih sulit lagi mendeteksinya.

Contoh : struktur data antara client dan server kemungkinan akan menyebabkan tidak konsisten. Hal ini tidak mungkin terjadi jika algoritma yang diterapkan benar.

### 4. Rangkaian kejadian tidak terlihat

Pemrograman terdistribusi banyak mempunyai masalah yang dikenal dengan *deadlock* . Hal ini terjadi jika terdapat dua atau lebih proses, masing-masing menunggu untuk melakukan sesuatu. Kemungkinan *deadlock* sering tidak terdeteksi selama pengujian sistem operasi dan *buffer* jaringan. Karena beban meningkat menyebabkan *buffer* penuh dan *deadlock* tidak bisa dihindari. Untuk mengatasi masalah *deadlock* ini yaitu jangan pernah menghalangi input atau output, tetapi gunakan pewaktuan. Asumsi umum pada program groupware adalah pesan yang dikirim dari satu komputer akan tiba dalam waktu yang sama pada komputer lain. Ini bergantung pada protocol yang digunakannya.

#### 10.5.7 Skala dan pengujian

Cara sederhana dapat mencegah kesalahan algoritma, seperti : menggunakan tabel daripada struktur data yang rumit, ukuran panjang field yang tetap untuk nama-nama dan pesan. Ini dapat mengurangi beberapa kesalahan sebelumnya dan merupakan teknik yang direkomendasikan untuk prototipe aplikasi. Pada proses pengujian terkadang menguji fungsi dari suatu aplikasi menggunakan beberapa window pada workstation yang sama, yang masingmasing bertindak sebagai user yang berbeda. Cara yang lain adalah simulasi untuk 'race condition' dan urutan yang ganjil dengan menjalankan sistem diantara 2 workstation kemudian tekan kunci panel secara simultan. Jika skala dan pengujian telah dilakukan dengan baik, sebuah software perlu diluncurkan di pasaran untuk memberi kesempatan bagi orang lain untuk menemukan bug atau kesalahan lain. Hal ini sangat penting demi perbaikan di kemudian hari. Coba kita amati berapa banyak versi dari sistem operasi Microsoft? Itu menandakan bahwa Microsoft terus melakukan perbaikan dari waktu ke waktu, baik dari segi interface, algoritma, skala penggunaan maupun ketahanannya. Perbaikan Microsoft tersebut seiring dengan perkembangan perangkat keras.

## 10.6 Bahan Diskusi

Dalam implementing synchronous groupware, jelaskan tentang :

- Architect for groupware
- Server faults
- Workstation faults
- Unforeseen sequences of events

## 10.7 Rangkuman

- *Computer-Supported Cooperative Work (CSCW)* adalah bidang studi yang berfokus pada perancangan dan evaluasi teknologi baru untuk mendukung kerja kelompok.
- Groupware merupakan jenis *software* yang mendukung kerja kelompok (*workgroup*) yang terhubung ke jaringan dengan kebutuhan bekerja sama.
- Bentuk-bentuk groupware yaitu asinkron tersebar, sinkron tersebar, tatap muka dan interaksi asinkron
- 

## 10.8 Latihan Soal-soal

1. Jelaskan sistem groupware?

2. Sebutkan hal-hal yang perlu diperhatikan dalam mengirim email?
3. Jelaskan tentang meeting rooms, berkenaan dengan meeting dan DSS?
4. Jelaskan fenomena yang mempengaruhi penggunaan CMC?
5. Sebutkan 4 tipe komunikasi tekstual dalam groupware?
6. Apa yang dimaksud dengan komunikasi *face to face*?