GROUPWARE

Ifeoluwa Idowu

GROUPWARE

What is Groupware?

Definitions of Groupware

- Computer-based systems that support groups of people engaged in a <u>common task</u> (or goal) and that provide an interface to a <u>shared environment</u>.

- <u>Secure, dynamic collaboration</u> solution that offers both traditional and mobile support for <u>any communication</u> over intranets, extranets and internet.

- A combination of technologies enabling an organization to <u>create</u>, <u>share</u>, and <u>leverage</u> an <u>accumulated knowledge base</u>.

- Commercial CSCW.

Little on the History of Groupware

- Plato Notes (1973), a research prototype at UIUC. Runs on numerous systems

- Iris Associates pioneered the concept of groupware based on Plato and released the product Lotus notes in 1989.

- The founders of Iris were the students that created Plato.

- Iris was later bought out by IBM. Good Fortune?

Classification of Groupware

- According to time-space matrix.

- More generally – real-time vs. non-real-time implementation.

	Same Time	Different Times
Same Place	face-to-face Interaction	asynchronous Interaction
Different Places	synchronous distributed interaction	asynchronous distributed interaction

Systems the Implement Groupware Functionalities

- Diff Time/Diff Place

  Email

  Calendaring and Scheduling

  Workflow

- Same Time/Diff Places

  Chat room

  Gaming

  Video Conferencing

  Shared Workspace

- HYBRIDS:

- Multi-user Editors

- Bulletin boards

- Content exchange

- Blackboard

- Online Classrooms

- Group Decision Support Systems and Electronic meeting rooms

What do they all have in common?

Generic Architecture
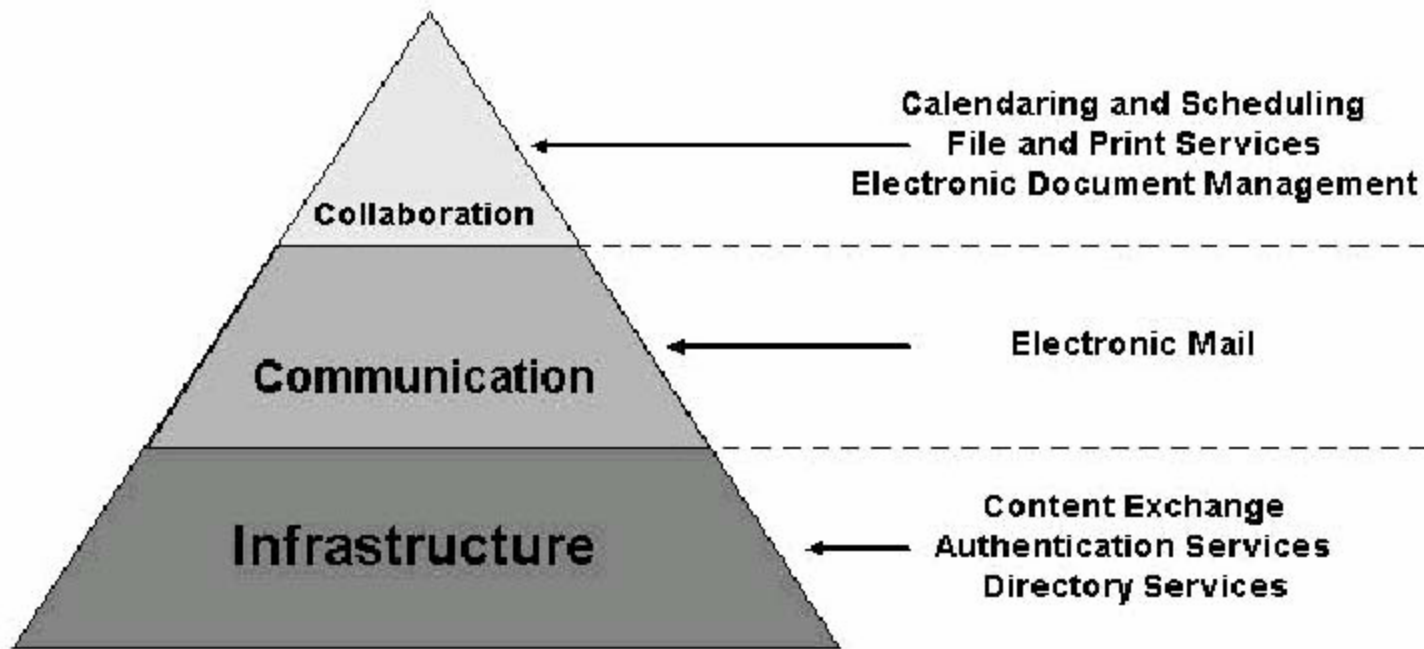
Do you now see what they have in common?

What they have in common

- Front-End/ Client side
  - Interface
  - Protocols
- Back-End/Server side
  - Protocols
  - Database/Storage
  - Other protocols for coordination

Communication Front-End

- Mail User Agent (MUA)

- Compose, Send, Receive, and Manage email messages

- Embedded services include addressing, packaging, signing, and encrypting outgoing messages; decrypting and displaying incoming ones

- May be bundled into a user-end software (ex. MS Outlook) or embedded into a word processor

- Polling server for updates

Communication Back-end

- Server – Message storage and Directory User Agent (DUA)

- Directory Services – manage names, email addresses and distribution lists

- Message Transfer Agents (MTA) – email delivery program. Server to server or server to user

- Email Application Programming Interface (API) – A series of subroutines that enable programs to access various components of the email system. Common API's include MAPI, VIM, and CMC

Communication Back-end Contd

- Email Gateways – transferring messages between incompatible email systems
- Protocols – the methods of communication that enables component to speak to one another: Transport Protocols (server to server), Access protocols (MTA to MUA), Directory Access protocol, Directory service protocol (oversees directories)

Collaboration-Calendaring and Scheduling

- Schedule events and access calendar for people, facilities or equipments.

- Applications usually have notification functionality built-in.

- Organizer notifies others and receives a reject or accept response.

- Communication between C&S servers is needed if implementing across a WAN.

- C&S servers store the calendars, manages them and generates notifications.

- Calendars reside on server and on users' systems.

Collaboration

- Can be real-time or appear to be real-time
- Needs synchronization
- Needs Order, access permissions, edit permissions, replication and more

Communication Layer Issues

- A standard email support system much be capable of handling text, images, tables, video, and sound

- None of this has the same format

- Email systems are not required to be compatible between organizations – gateways are employed to solve this problem

- Storage of the above unstructured files must be addressed
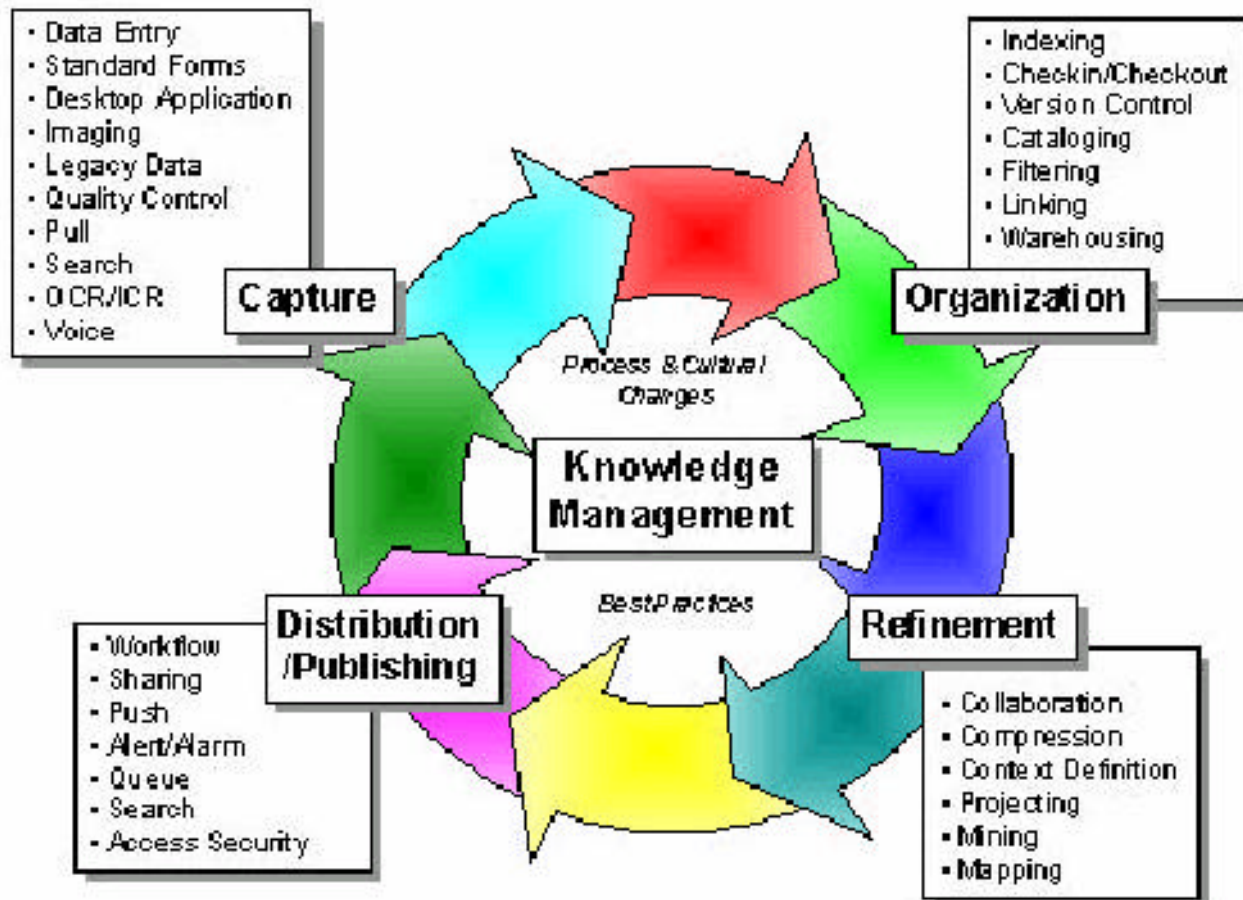
Collaboration Layer Issues

- Creation of Different types of Electronic document
- Sharing of these documents
- Storing and locating the files
- Edit permissions
- Replication and synchronization
- Granularity

General Implementation Issues

- Collection
- Organization
- Management
- Processing
- Location
- Distribution
- File Permissions
- Coordination

- Transparency of physical location to users and search tools
- Universal Access
- Searching and obtaining results
- Indexing of documents
- Concurrency of docs
- Workflow

Technology Components

- Document Capture/Creation

- Indexing

- Storage

- Routing and Distribution

- Search and Retrieval

- Display and View

- Output and Publishing

- Replication

- Data Entry
- Standard Forms
- Desktop Application
- Imaging
- Legacy Data
- Quality Control
- Pull
- Search
- OCR/ICR
- Voice

Capture

- Indexing
- Checkin/Checkout
- Version Control
- Cataloging
- Filtering
- Linking
- Warehousing

Organization

Process & Cultural Changes

Knowledge Management

Best Practices

Refinement

- Workflow
- Sharing
- Push
- Alert/Alarm
- Queue
- Search
- Access Security

Distribution /Publishing

- Collaboration
- Compression
- Context Definition
- Projecting
- Mining
- Mapping

Document management

Architecture of Groupware

- Generic – a general view of what a groupware system will look like if I built it.
- Proprietary – Consider systems that are already commercially used including Lotus, MS Exchange, GroupWise and Grove.
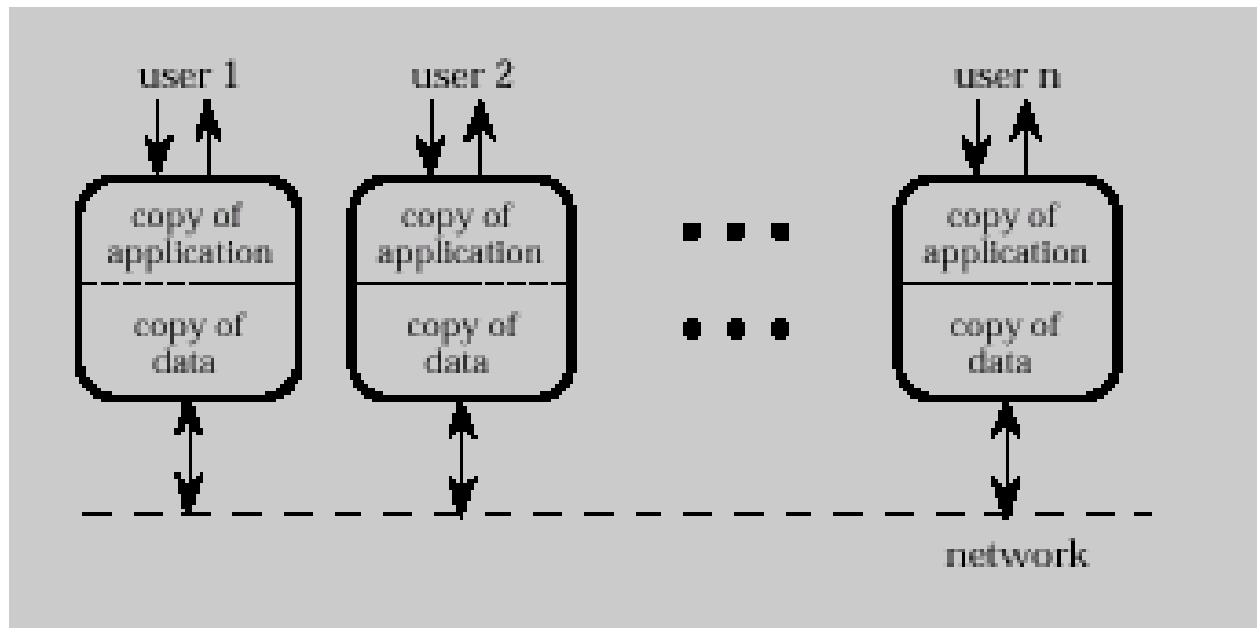
Architecture I - Centralized
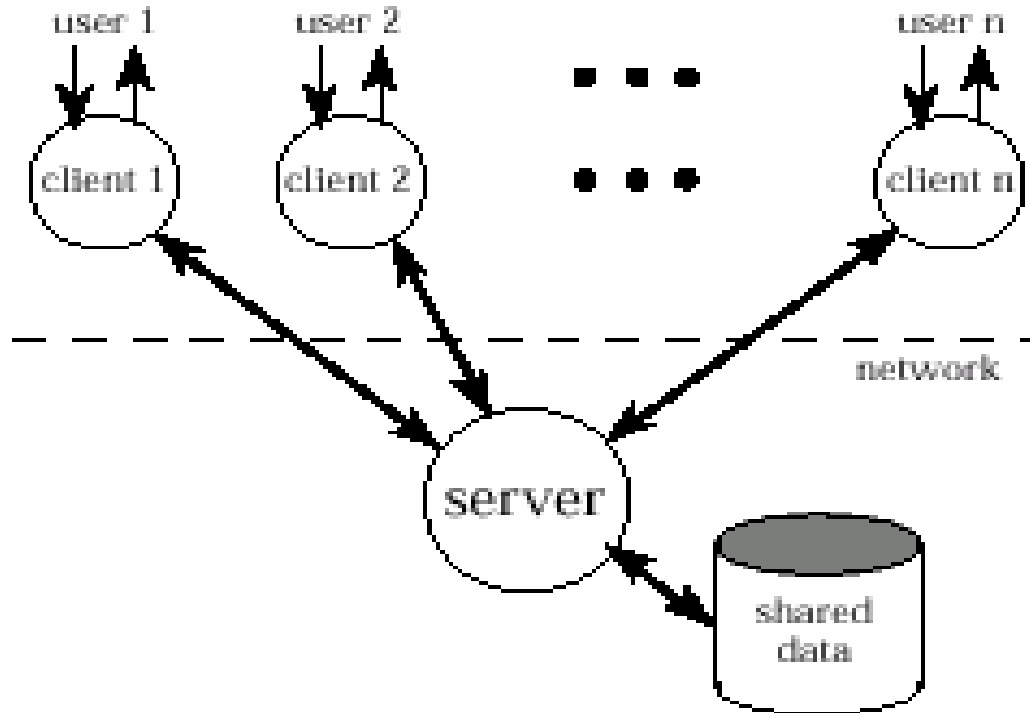
Single copy of application and data

Client-server – simplest case

Architecture II - Replicated

Also known as peer-peer

Copy on each workstation – consistency issue

Architecture III-Hybrid/Half-way

- Local copy of application
- Central database
- Local cache of data for fast feedback
- Centralized support

  locking – explicit or implicit

  update propagation

Generic Architecture-Infrastructure

- The foundation that enables communication and collaboration across an organization
- Includes content exchange, authentication services, and directory services
- File formats are described here – standard formats are preferred to facilitate exchange
- Access to the system functions through mobile devices, internet browsers, or dedicated client-side interface/software

Database Topics - Indexing

- B^+-Indexing. Why?
- Indexing must be created at initial storage
- Index are created by assigning keywords to the documents before storage
- Local versus universal index
- Documents can be scanned by appropriate software for content indexing. Video too?

Structured and unstructured data

- Techniques

  - Individual files

  Problems: Search and retrieval takes too long; space consumption; no portability; machine dependent; still being used in Exchange email system. Why? How?

  - BLOB

Binary Large Objects (BLOBS)

- Application dependent not machine
- Portable between copies of application of different platforms
- All files are stored in single database file and indexed
- Search is done on B+ tree and pointer points to the beginning of file
- Info about file(Type, App) is stored at the beginning of storage location
- Not adopted. Commercial systems are proprietary

Database type – Shared

- Database must be able to support features like transparency, isolation and locking which are not supported standard databases.

- Standard databases also do not support notification when data changes, a feature that is central to groupware.

- Replicated copies can be controlled by the use of constraints and triggers.

- Replication across a network or in a cluster may require the use of a log file.

Database - Storage

- Structure and unstructured data are stored
- Generally used solution is to convert the files to binary large objects (BLOB) with the information about each file preceding it
- This binary format is independent of platform thereby promoting replication and portability. MS Exchange defies this rule

Updating Active Values

- An active value is data that is being viewed by more than one user while being editable by one
- Each user's view must be updated as the edit is done
- Applications of this include blackboard
- This can be implemented using constraints

user 1

1 7 9

user 2

1 7 9

change event

add 1

1 8 0

active value
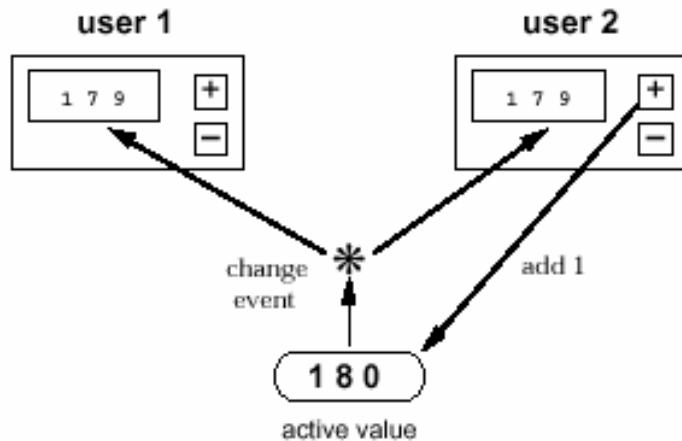
Concurrency Constraints

- Equality constraint enforces that users' view is constantly similar. Simplicity of algorithm and network connectivity play a big role

- Floor control is also used to order user input

- What about real-time synchronous multi-user editors?
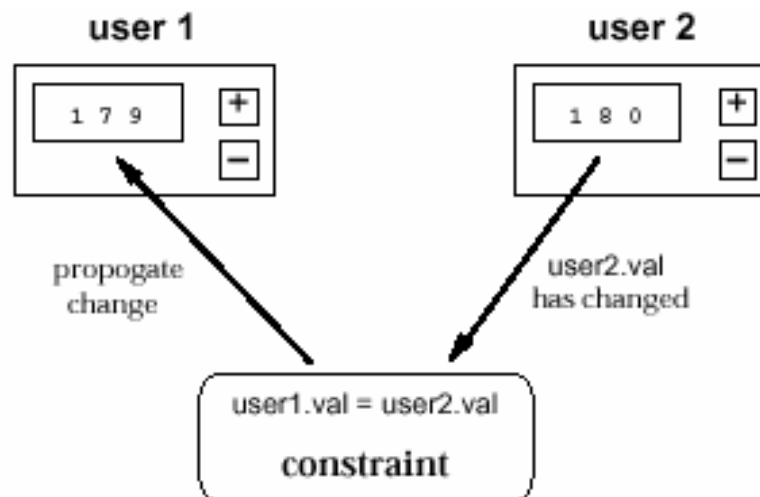
Multi-user editor

- Same document is opened for editing by multiple users. Ex. Grove
- Locking granularity – page, paragraph/function, line, word
- Lock request, allocate and release
- Update propagation – character, word ?
- Frequency of write-backs, manual or auto

Database Availability

- Not an issue in single server. There or not.
- Cluster and distributed systems must provide fail over support.
- Virtual server in Exchange.
- State preservation.
- Recoverability.

Recoverability

- Another advantage of Uncentralized architecture

- Generally log-based

- Concurrency and consistency also log-based

- Can they be state-based like Ben's? No

  - Files with the same keyword-index may have only time differences. Ex. Email

What we know thus far

- Different data types – text, image, video, database; structured and unstructured
- Asynchronous and synchronous
- Replicated, central and hybrid DB
- Several backend and front-end operations
- DB must be current, updated, indexed, replicated among servers or nodes, secure, operations have to be coordinated
- How is this done? Lets ask the Pros!

Commercial Implementation

- GROVE (GRoup Outline Viewing Editor)
- Lotus Notes/Dominos (How the pros do it)
- Novell GroupWise (http://www.novell.com/products/groupwise)
- Groove (http://www.groove.net) - Full package
- Microsoft Exchange (Most Complex)

Building your own Groupware

- Groupware Toolkits

  - Suite

  - Mead – Groupware specific

    Developed in U. of Lancaster by Richard Bentley

  - Rendezvous

- Programming - Hard

Causes Of Failure

- Hardware Failures – Too many components
- Programming error – Complex Algorithms
- Unforeseen sequence of events – Interleaving and delays
- Scalability