Groupware and Collaborative Interaction

Distributed Interactive Systems

Technical aspects

Anastasia.Bezerianos@lri.fr

slides based on lecture by Cédric Fleury

Introduction

Technical aspects of distributed interactive systems

- Requirements common across all CSCW applications
  - Network architecture
  - Data distribution
  - Concurrency management

E.g., Collaborative virtual environments

- Strong requirements
  - Users are interacting in real-time
  - Immersion requires fast multi-modal feedback
- Lots of solutions to overcome the technical issues

Collaborative Virtual Environments

Enable users to work or have fun together

2 kinds of collaboration in virtual environment

- Co-located collaboration
- Remote collaboration

<u>Aspects of collaboration</u>

- Awareness
- Communications
- Collaborative interaction

3

Collaboration requirements

For efficient collaboration, users need to:

Have the same state of the virtual environment (virtual objects) at the same time

⇒ Consistency of the VE

Modify the virtual objects in real-time

⇒ Responsiveness of the system (interactivity)

Consistency

[Delaney et al., 2006]

Distributed virtual environment

- Distributed database of virtual objects with users modifying it in real-time

Manage the consistency

- Ensure that the database is the same for all users

Inconsistencies due to:

- Concurrent modifications
- Delay to transmit modification on the network

Responsiveness

[Delaney et al., 2006]

Responsiveness of the system

- Time required to respond to users' actions (latency during users' interaction, jitter)

Due to the time required to:

- Process and send users' actions
- Transmit actions on the network (if mandatory)
- Give a feedback to the users

Between 40ms and 300ms, under 100ms is good

Distributed Virtual Environments

Find a good trade-off between consistency and responsiveness (depending on task, application, etc.)

Technical requirements

- Connect remote computers
- Distribute data
- Share information
- Manage concurrent accesses to the data

=> Each technical choice must consider both consistency and responsiveness

Outline

Network Architecture

Data Distribution

Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

Outline

Network Architecture

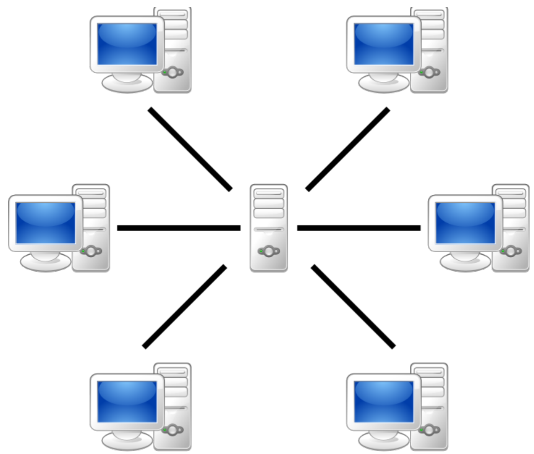
Data Distribution

Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

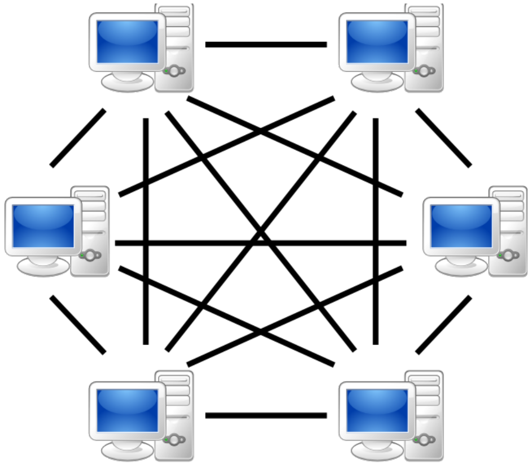


Network Architecture

Client/server architecture
[Vistel 95, RING 95, BrickNet 95, ShareX3D 08]

- All communications pass through the server
  - latency during interactions
- All nodes can be contacted quickly
  - Consistency and synchronization are easy to ensure
- A “bottleneck” can occur on the server

10



Network Architecture

Peer-to-peer architecture

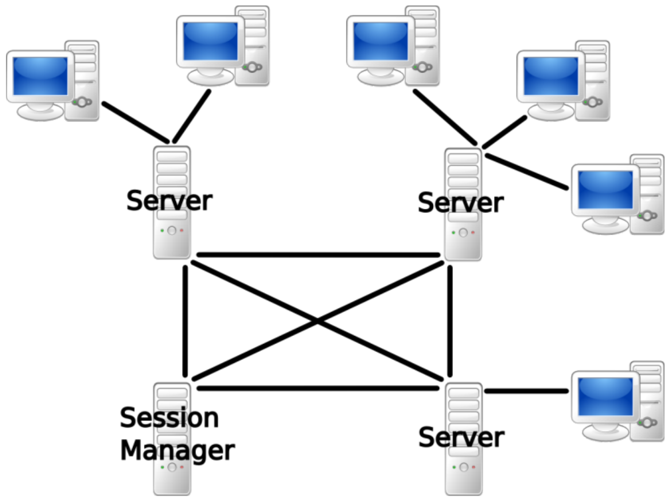
[Reality Build for Two 90, MR Toolkit 93, SIMNET 93, NPSNET 94]

- Fast communications between pairs of nodes
  - Closely coupled interactions between a few users
- Difficulties to contact all nodes at the same time
  - Consistency and synchronization are hard to ensure
  - Many messages are transmitted over the network

Network Architecture

Hybrid architecture

- Servers connected with peer-to-peer connections [SPLINE 97]
  - Avoids the “bottleneck” on a single server
  - Connects nodes with specific requirements
  - Increases system latency



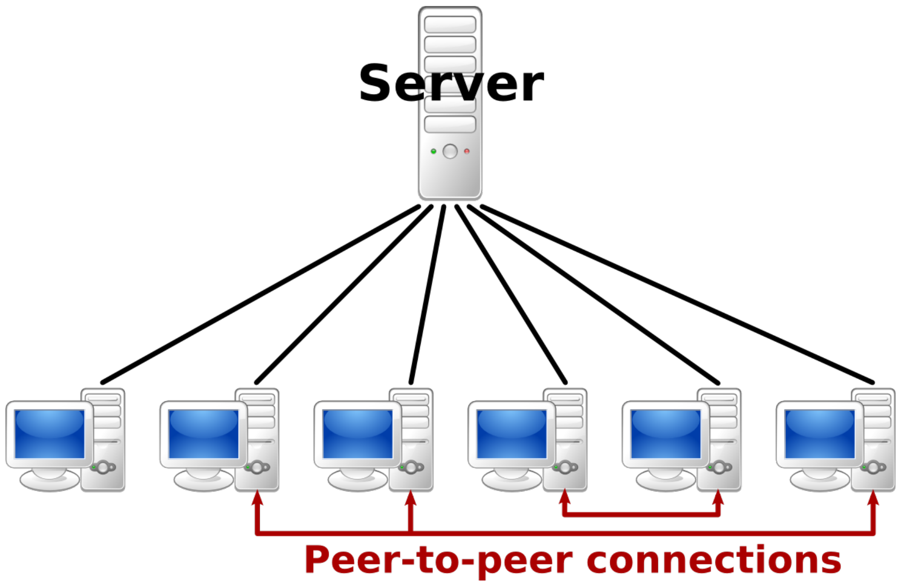
12

Network Architecture

Hybrid architecture

Temporary peer-to-peer connections [Anthes et al., 04]

- Are established according to users' locations in the VE
- Increase CVE consistency between nearby users



Outline

Network Architecture

Data Distribution

Communication Protocols

Communication Reduction Mechanisms

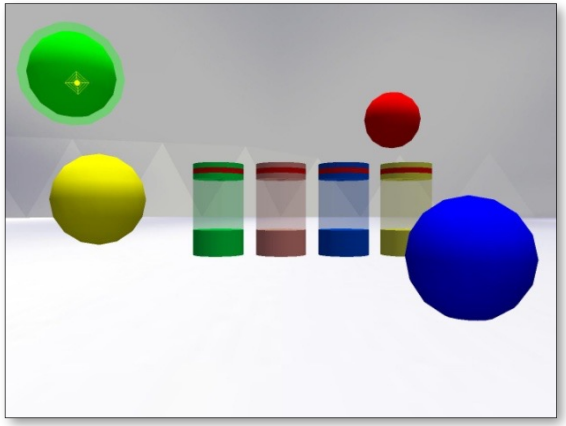
Consistency Management Mechanisms

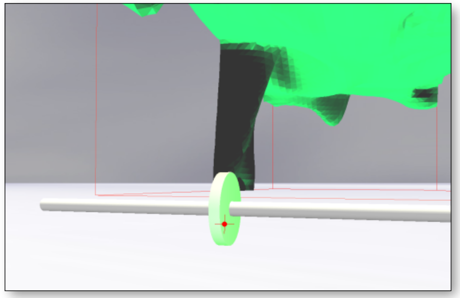
Software architecture

Data Distribution

A virtual object

- A set of parameters (data)
  - Identifier
  - Attributes (position, orientation, etc.)
  - User access rights
  - Geometry, and possibly textures
- A behaviour
  - Only reactive (responding to user actions)
  - Continuous (evolving in the time)





⇒ Which computers store the data?

⇒ Which computers manage the data modification?

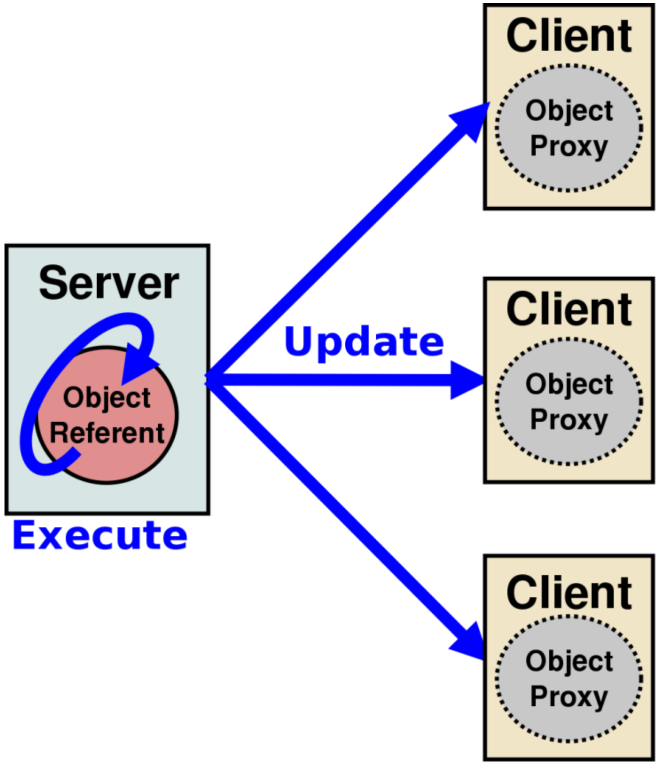
⇒ Which computers execute the behaviour?

Data Distribution

Centralized [Vistel 95]

Data is stored on the server

Behaviours are executed on the server



16

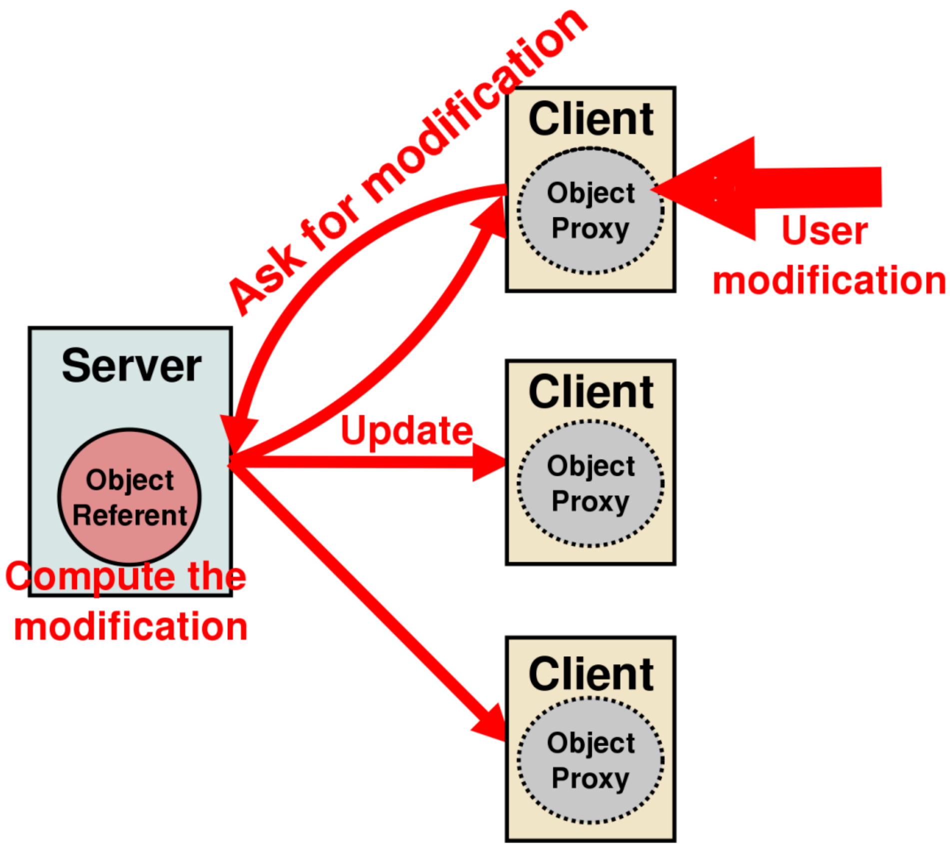
Data Distribution

Centralized [Vistel 95]

Data is stored on the server

Behaviours are executed on the server

Modification requests are processed on the server



Data Distribution

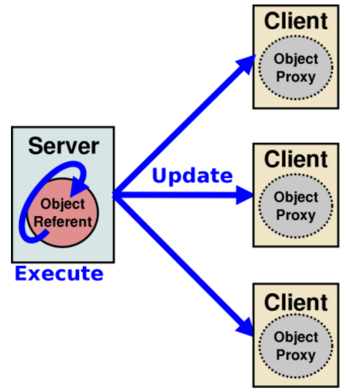
Centralized [Vistel 95]

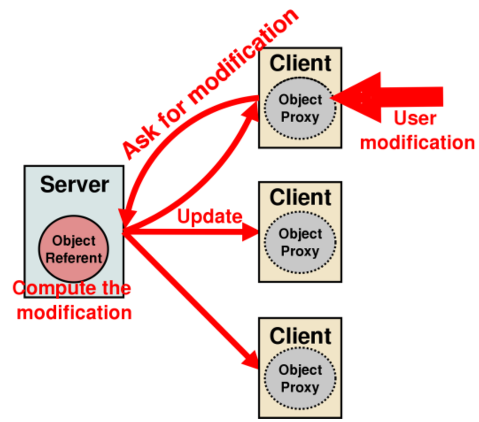
Advantages

- Ensures a global consistency
- Avoids data replication
- Avoids behaviours processing on the clients

Drawbacks

- Introduces latency during interactions
- Transmits many messages over the network





18

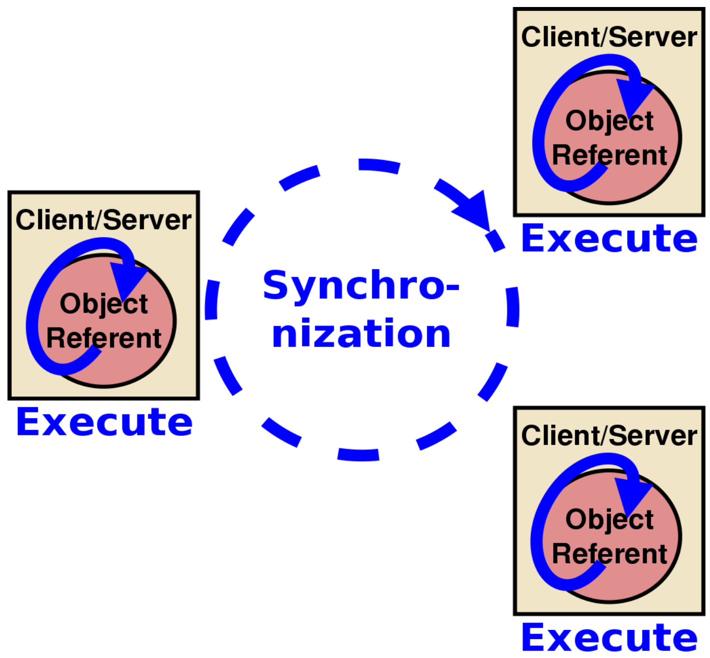
Data Distribution

Replicated [SIMNET 93, MR Toolkit 93]

Data is replicated on each node

Synchronization between nodes can be achieved

Behaviours are executed on each node



19

Data Distribution

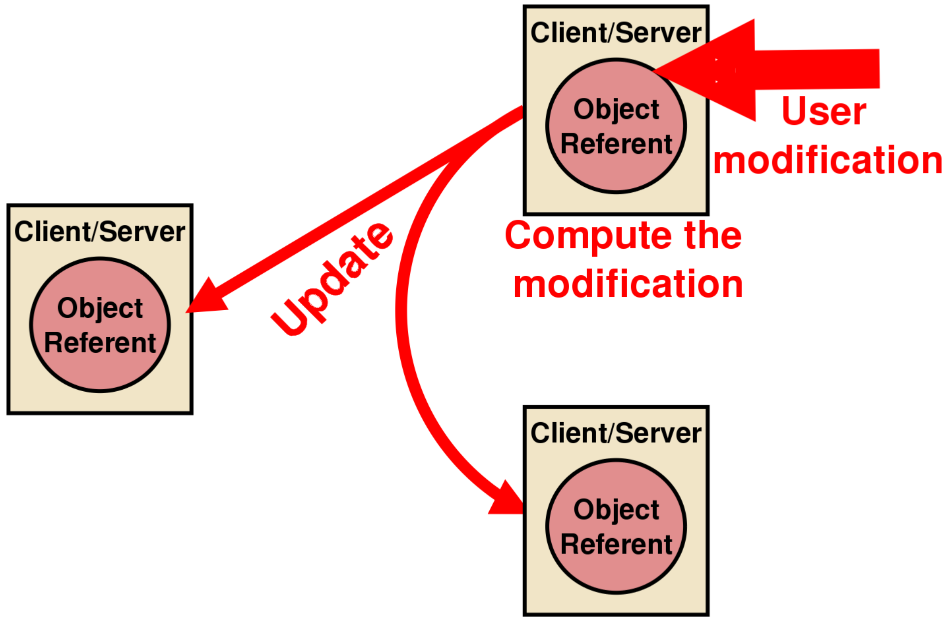
Replicated [SIMNET 93, MR Toolkit 93]

Data is replicated on each node

Synchronization between nodes can be achieved

Behaviours are executed on each node

Modification requests are processed locally



20

Data Distribution

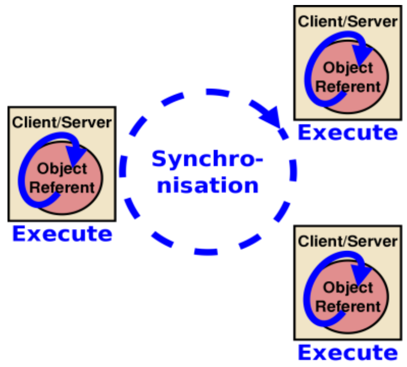
Replicated [SIMNET 93, MR Toolkit 93]

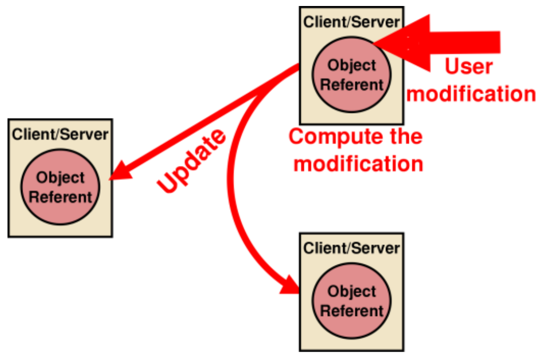
Advantages

- Low-latency interactions
- Few messages transmitted

Drawbacks

- Data replication
- Behaviours processed on each node
- Inconsistencies due to transmission delay of update messages
- Additional mechanisms for managing concurrent accesses





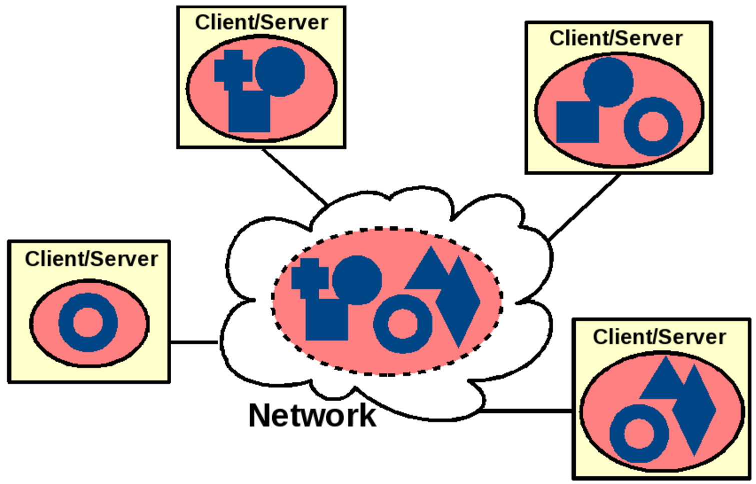
21

Data Distribution

(other models) Hybrid [DIVE 98] [BrickNet 98]

Only the necessary objects are replicated

A server saves the whole VE state



Advantages

- Reduction of data replication
- Less processing on each node

Drawbacks

- Difficulties to ensure consistency and manage concurrency
- Many messages transmitted over the network
- Dynamic downloads of additional objects

Data Distribution

(other models) An adaptive data distribution [Fleury et al., 2010]

- Three modes of data distribution
  - Centralized
  - Replicated
  - Hybrid
- Chosen independently for each object
- Changed dynamically during a working session

Data distribution ≠ network architecture

Network architecture: how msgs are transmitted physically

Data distribution: how msg flow logically

Does not have to be the same !

- Centralized data dist. with peer-to-peer connection
- Replicated data dist. can be used with client/server arch.
  - Ex: video games

Outline

Network Architecture

Data Distribution

Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

Communication Protocols

Classic protocols (TCP, UDP)

- UDP: faster, simpler (but may have packet loss)
- TCP: more reliable but a bit slower

(also) Virtual Reality dedicated protocols

- [RTP\I 99]: adapt RTP for interaction
- [VRTP 97]: support VRML (virtual reality modelling language)
- Some others [DWTP 98, DIS 93, HLA 97, ISTP 97]

Communication Protocols

Specific protocols in industrial environment

- Deal with:
  - Standard Internet access
  - Firewalls that support only HTTP and HTTPS protocols

More generic standards are emerging

- VRPN (Virtual-Reality Peripheral Network)
- OSC (Open Sound Control)

27

Outline

Network Architecture

Data Distribution

Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

Communication Reduction Mechanisms

Avoid to overload the network

- Big number of users
- Low bandwidth network

Reduce the number of messages transmitted on the network without:

- Reducing the consistency
- Increasing the latency during interactions

Message filtering

Send only the updates to the concerned users

- Avoid overloading the network
- Reduce the processing time of the messages

Reduce the # of shared objects between users

Filter according the area of interest of users

- Objects close to a user [Waters et al., 1997]
- Objects in the field of view of a user [Funkhouser, 1995]

Compression & Aggregation

Compression

- Not relevant for position/orientation [Joslin et al., 2004]
- But when data starts to be complicated
  - Joints of a virtual avatar, physical simulation data
- Load new virtual objects (geometry, level of details)

Aggregation

- Send all the object updates in one message
- Can introduce delay in message transmission

Outline

Network Architecture

Data Distribution

Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

Consistency Management Mechanisms

Inconsistencies due to

Network delay

Concurrent modifications

2 ways to address them

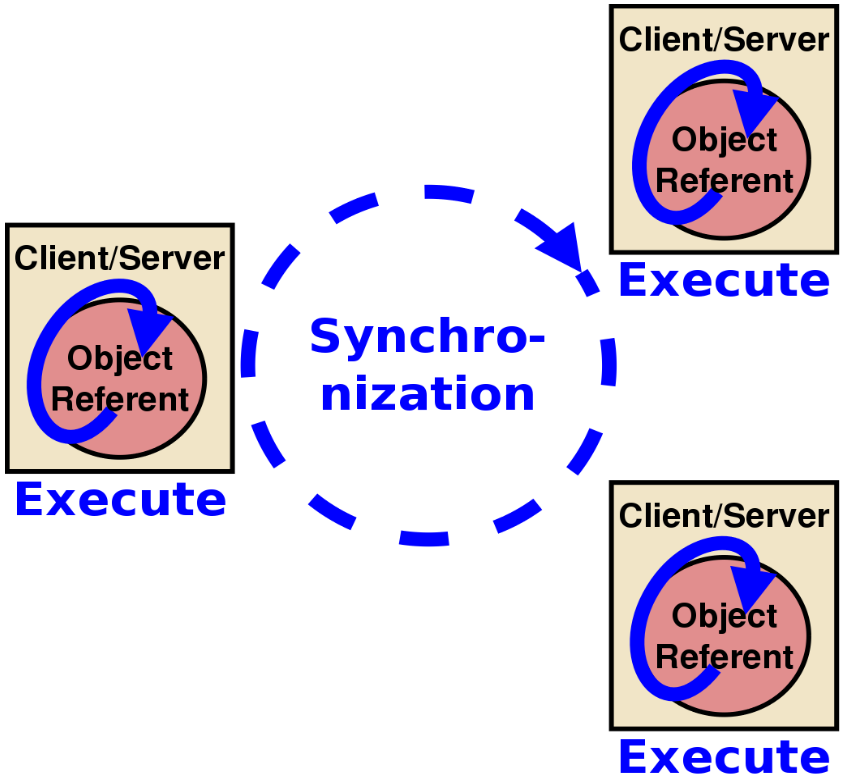
Synchronization

Concurrency control

Synchronization

Ensure that each user has the same state of the virtual env.

Especially in replicated mode



34

Synchronization

Time is a fundamental element of CVE

- Absolute time: synchronized clock (UTC)
- Logical or virtual time: logical clock
  - Ordered sequence of events
  - Uses timestamp

Some examples of synchronization …

35

Synchronization

Lockstep synchronization [Ring 95, OpenMASK 02]

- Waits for all nodes before computes the next step
  - Each node sends acknowledgements to the system
  - Then, the system allows nodes to process the next step
- Advantages
  - Perfect synchronization
  - Events are processed in the correct order
- Drawbacks
  - Real-time is not guaranteed
  - One node can slow down all the others

Synchronization

Delayed global consistency [Delaney et al 06]

Mark events with a timestamp using a logical clock

Execute events following the correct timestamp order

Advantage

Causality is ensured

Drawback

No time synchronization

37

Synchronization

Time warp synchronization [Jefferson 85]

- Events are marked with a timestamp
- Events are processed as soon as they arrive
- "Rollbacks" are used to solve causality errors
  - Incoming event older than the event already processed
- Advantage
  - No Latency during interactions
- Drawback
  - "Rollbacks" are very annoying for the users (feedback)
  - "Rollback propagation"

Synchronization

Server synchronization [ShareX3D 08]

- Server manages a “state number” for each object
  - Increments the “state number” for each modification
- Server sends the last received update to nodes if they are not up-to-date
- Advantages
  - Ensures that nodes are up-to-date
  - Reduce the number of sent messages
- Drawback
  - No causality and no time synchronization

Synchronization

Trade-off Consistency <-> Speed

Important for real-time / synchronous collaboration

e.g., collaborative virtual environments, collocated collaboration

Other solutions also exist, such as predictive time management [PARADE 97], or hybrid approaches like imposed global consistency [Delaney et al 06]

Concurrency control

(easier) Centralized mode or hybrid mode with 1 referent

Server/referent can handle concurrent modification requests

(harder) Replicated mode (several referents / nodes)

Virtual objects can be modified locally on several nodes at the same time

Concurrency control is required across nodes

Concurrency Control

3 main modes of concurrency control

- Pessimistic mode [BrickNet 98]
  - Only one user can modify an object at the same time
- Optimistic mode [Delaney et al 06]
  - No concurrency control during interactions
  - A <u>correction</u> is necessary when conflicts occur
- Prediction based mode [PARADE 97, ATLAS 07]
  - Predict which users will probably modify an object
  - Give priority to the users according to the prediction

Users' Access Rights

Give different access rights to users

- Protect virtual objects (confidential data, non modifiable objects, etc.)
- Assign some role to users

3 criteria

- Right to see an object
- Right to modify its parameters
- Right to create/delete objects

Use a scale of access level from 0 to N
(0 is the most restrictive)

Thought Exercises

Text Editor

(Network Architecture?)

Data Distribution?

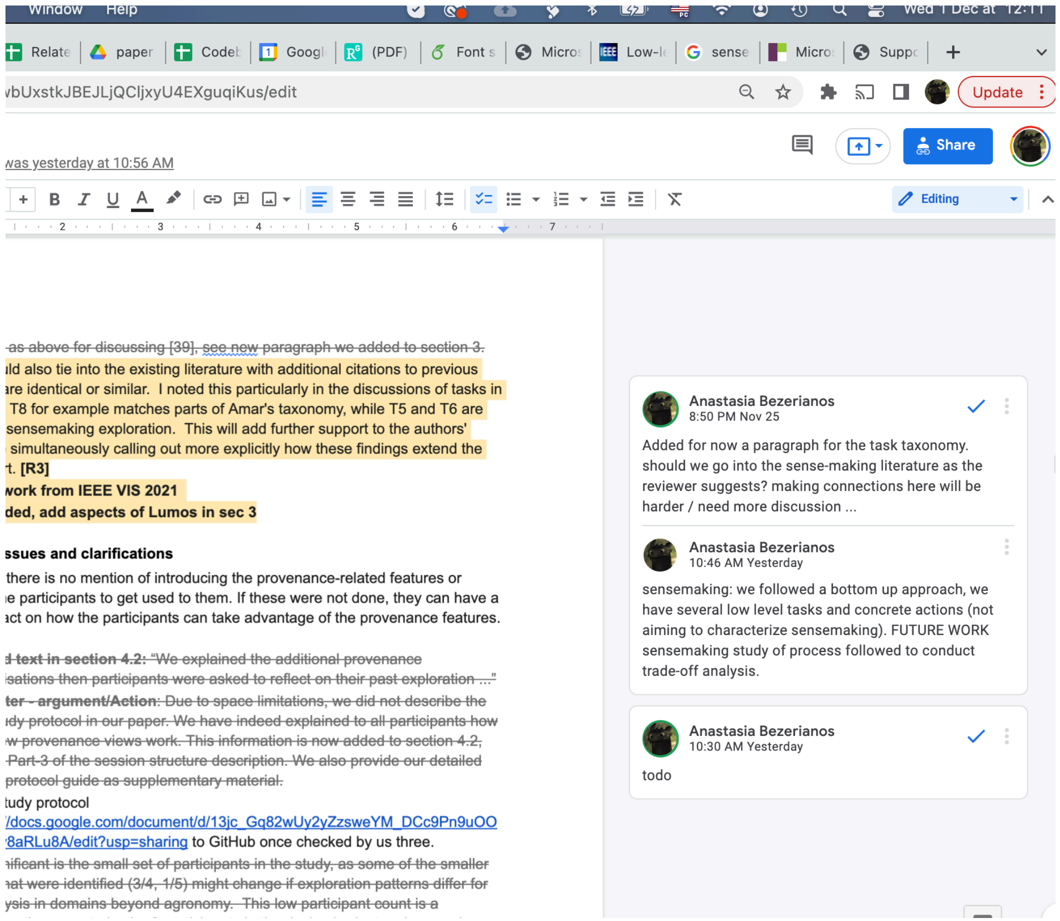
(centralized or replicated)

Consistency Management?

Synchorization or Concurrency

Communication

Reduction?



Online Multiplayer Game

(Network Architecture?)

Data Distribution?

(centralized or replicated)

Consistency Management?

Synchorization or Concurrency

Communication Reduction?

Chat Application

(Network Architecture?)

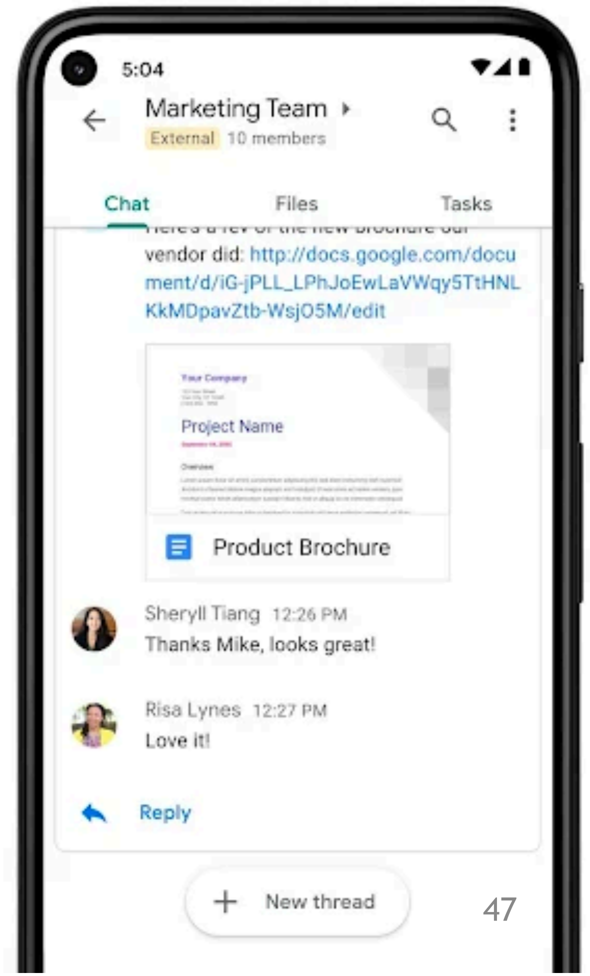
Data Distribution?

(centralized or replicated)

Consistency Management?

Synchorization or Concurrency

Communication Reduction?



47

Collaborative Coding Application

(Network Architecture?)

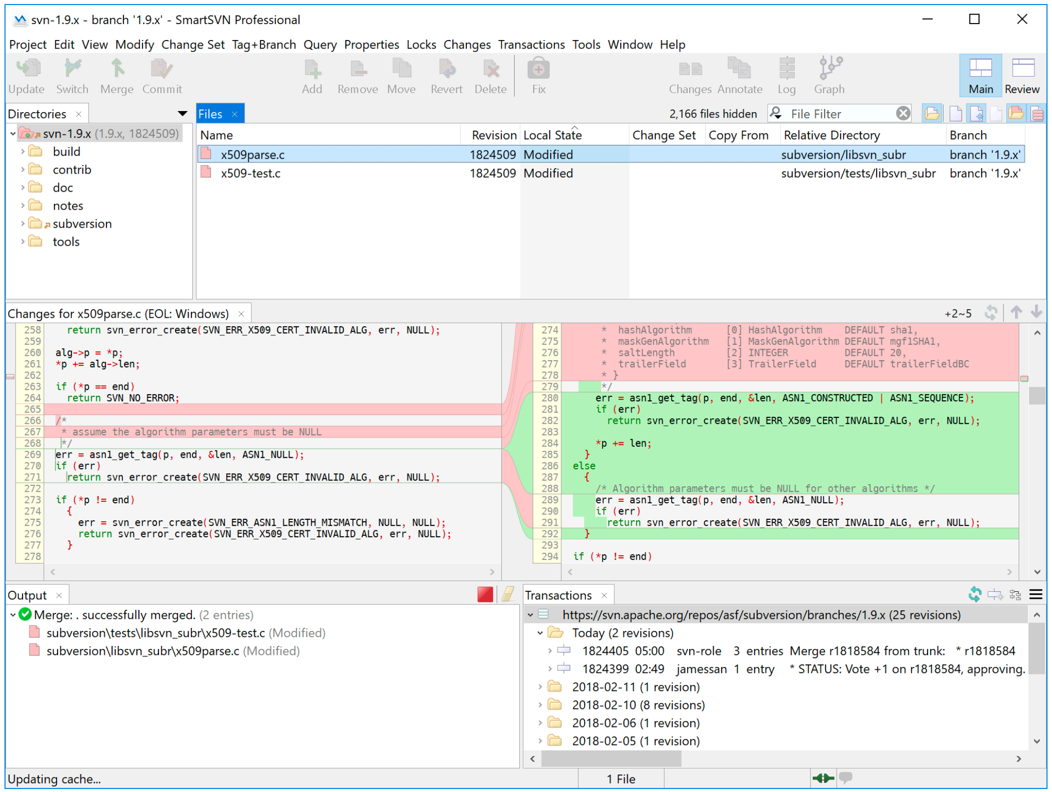
Data Distribution?

(centralized or replicated)

Consistency Management?

Synchorization or Concurrency

Communication Reduction?



Outline

Network Architecture

Data Distribution

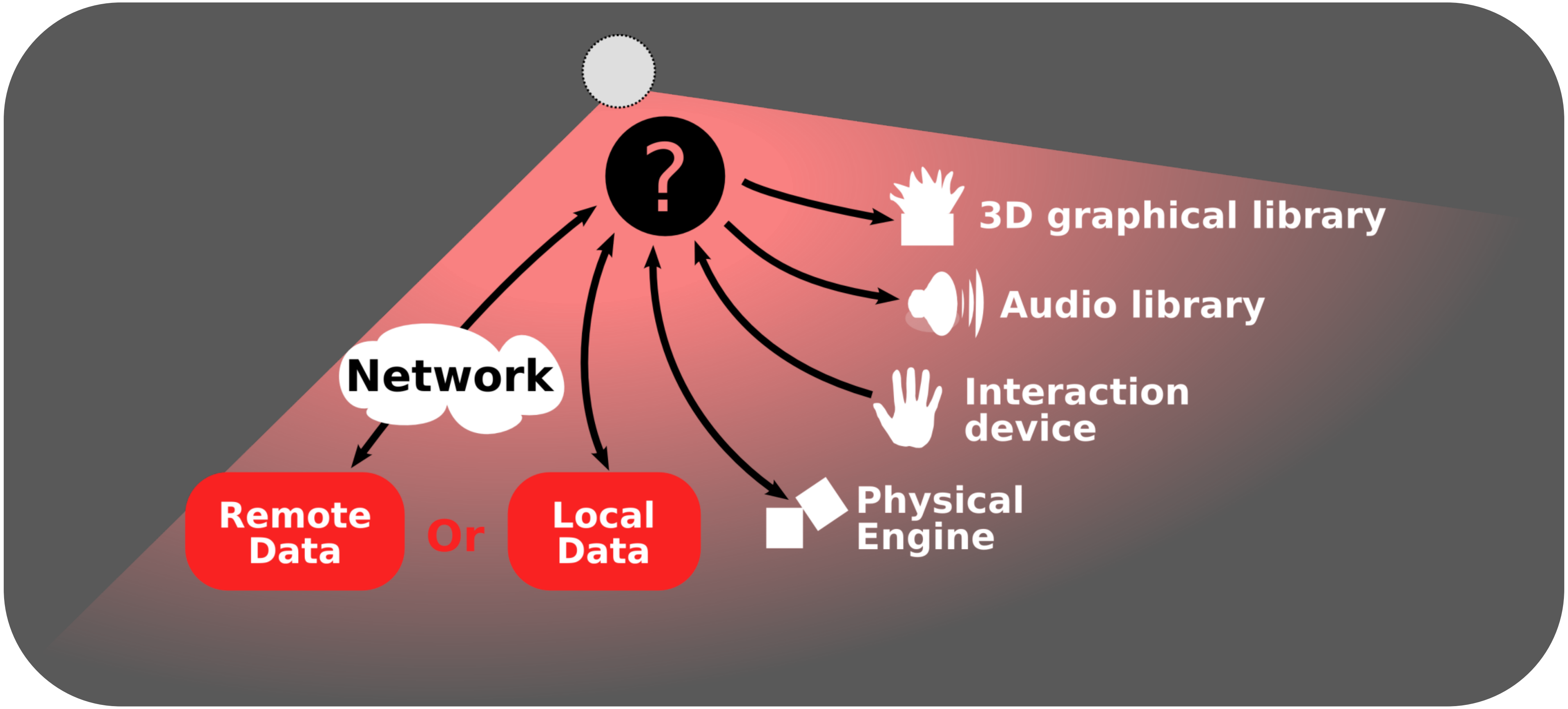
Communication Protocols

Communication Reduction Mechanisms

Consistency Management Mechanisms

Software architecture

Software Architecture



⇒ How to design virtual objects in order to ensure a good separation between data distribution and multiple representations?

Models for Interactive System

Application can be decomposed in 3 parts

- Core components
  - Store data
  - Execute behavior
  - Process users' modification requests
- Interface component
  - Make the link with the users
    - Display the object
    - Register the action of the users
- A link between the Two components

Conclusion

Common issues of CSCW applications

- Trade-off between consistency and responsiveness
  - Network architecture and data distribution
  - Consistency management mechanisms

⇒ No solution which fits all application requirements, so an adaptive solution might be a good solution

- Software architecture has to deal with
  - Data distribution over the network
  - Various software libraries and materiel devices

⇒ Make a clear separation between core application part, data distribution part, interface with the users