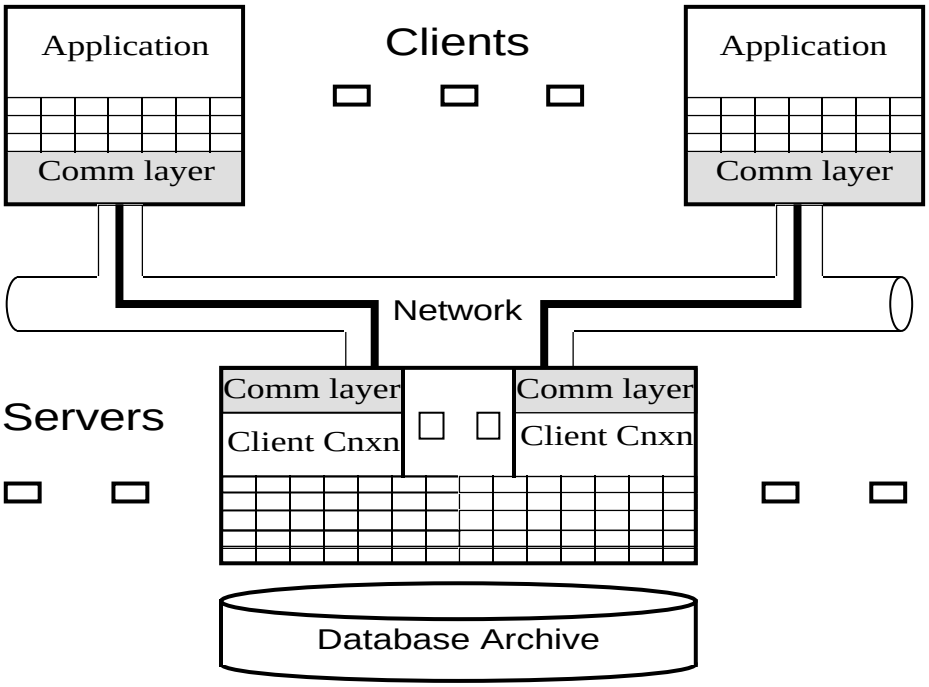
Shared Distributed Memory : the Workspace Model

Didier DONSEZ, Liming CHEN, Pascal FAUDEMAY†

Laboratoire HEUDIASYC - Université de Technologie de Compiègne
†Laboratoire MASI- Université Pierre et Marie Curie (Paris VI)

e-mail : {donsez,lchen}@hds.univ-compiegne.fr
e-mail : faudemay@masi.ibp.fr

The OO-DBMS Architecture



Client / Server Model fits well to LANs with powerful Workstations

But there are some limitations ...

Does not fit to Heterogeneous LAN / WAN Networks

Use of Check In / Check Out Mechanisms

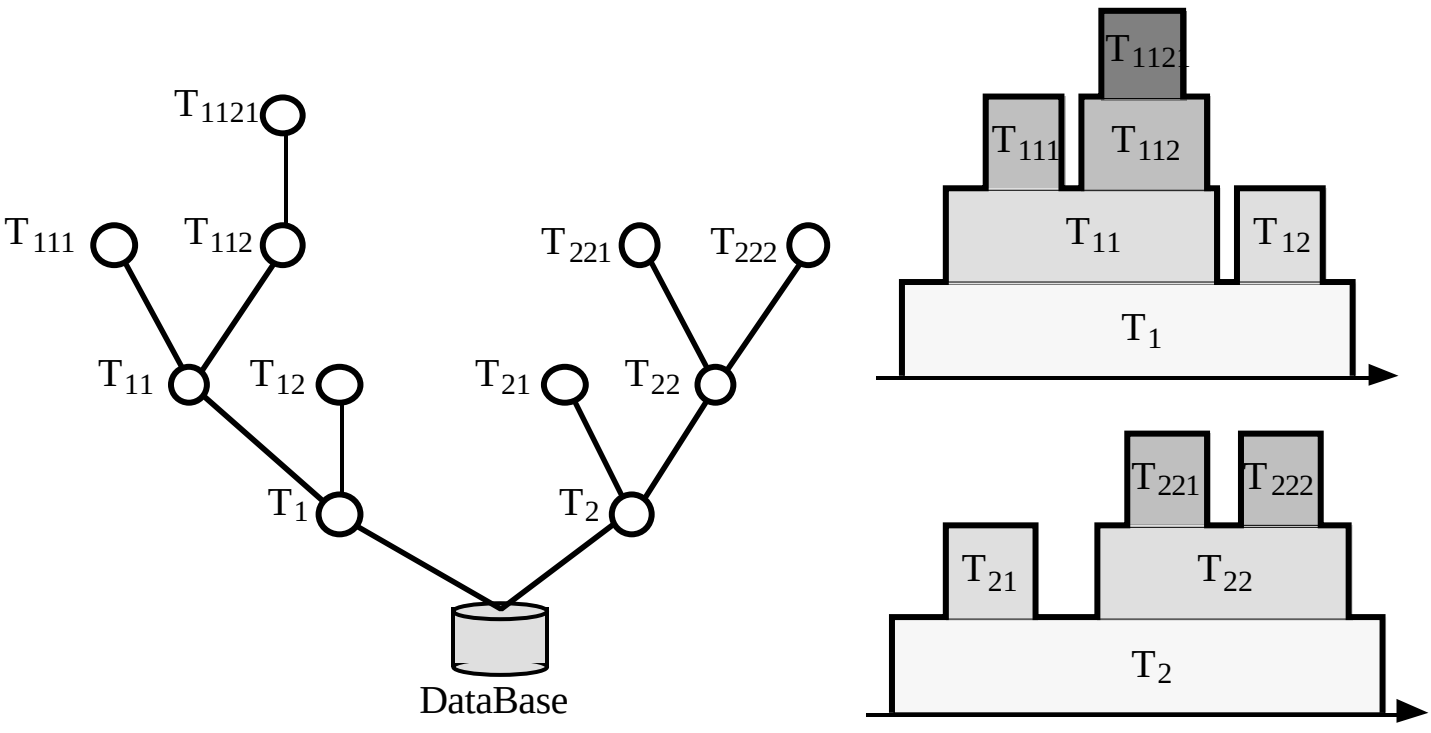
Need for Query Shipping

- **Weak Clients**

  Nomadic PCs, Walkstations

- **Security and Confidentiality**

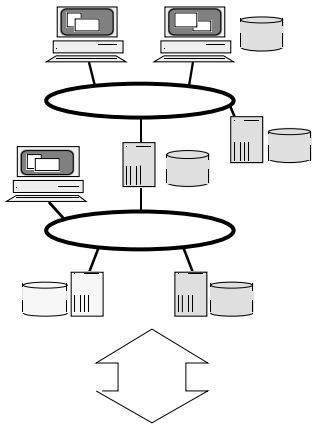
The Nested Transaction Model [Moss85]



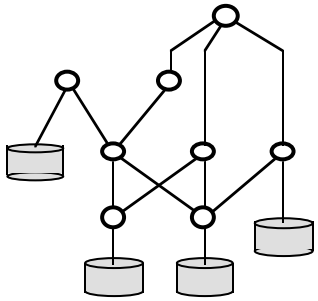
The Workspace Model

Goal :

- **Extend Moss' Model**
- **Map Client and Server Processes on a graph of "extended" nested transactions**



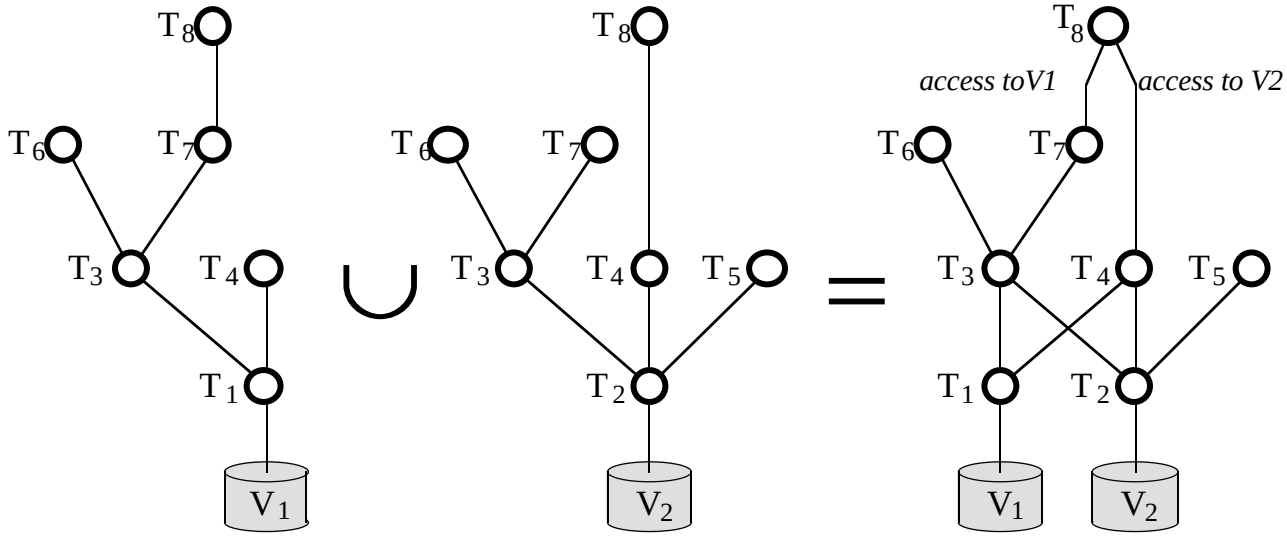
the Workspace Model = Transaction Model + Process Model



WS Model : The Transaction Model

Moss' Model +

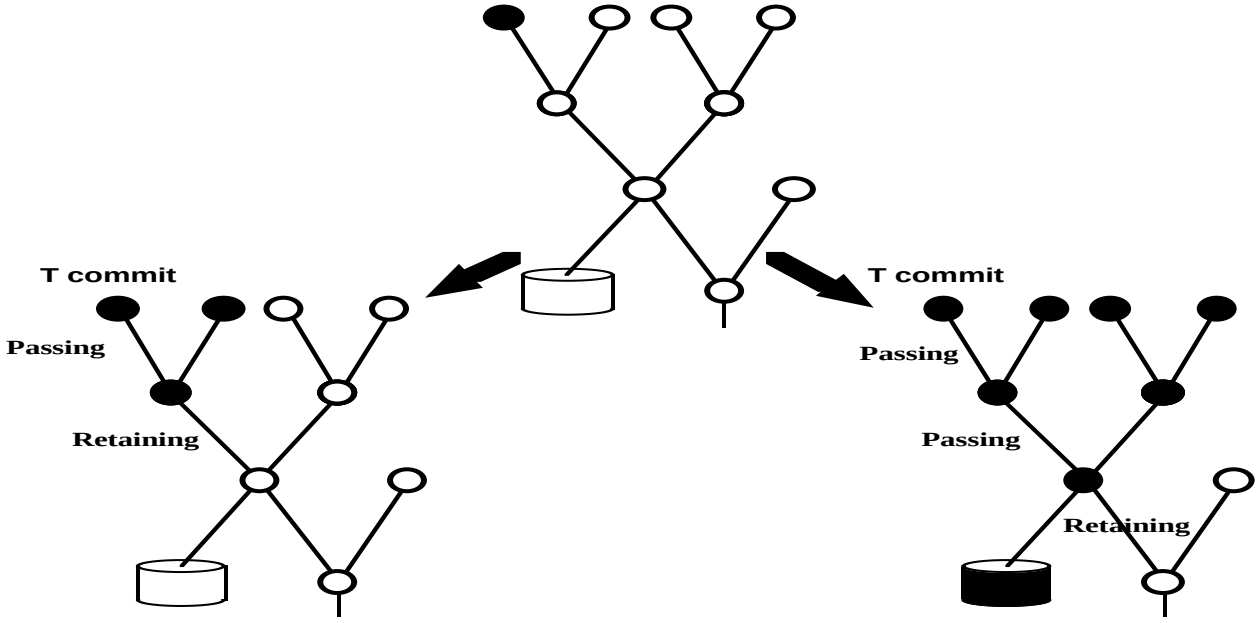
- **simultaneous "sister" transactions**
- **multiple parents transactions**
- **retaining or passing export**



WS Model : Retaining Transaction / Passing Transaction

Retaining Export — **Moss' Commit**

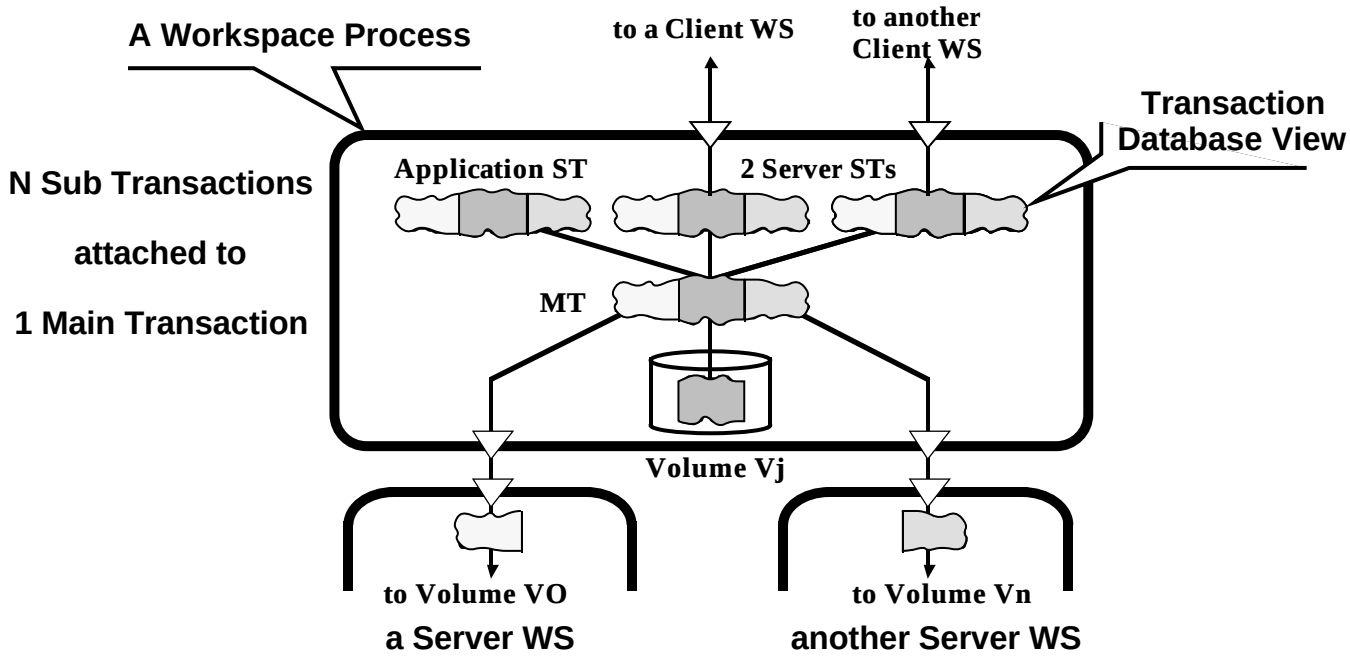
Passing Export — **Behave as a Data Cache**



WS Model : The Process Model

Restrict the graph of extended Transactions

to efficiently implement the Workspace Process



Instanciations of the Workspace Model

A Workspace may be

- **a standalone machine,**
- **a client,**
- **a server or**
- **both client and server**

Instanciations of some architectures :

Standalone Database

Client / Server Database

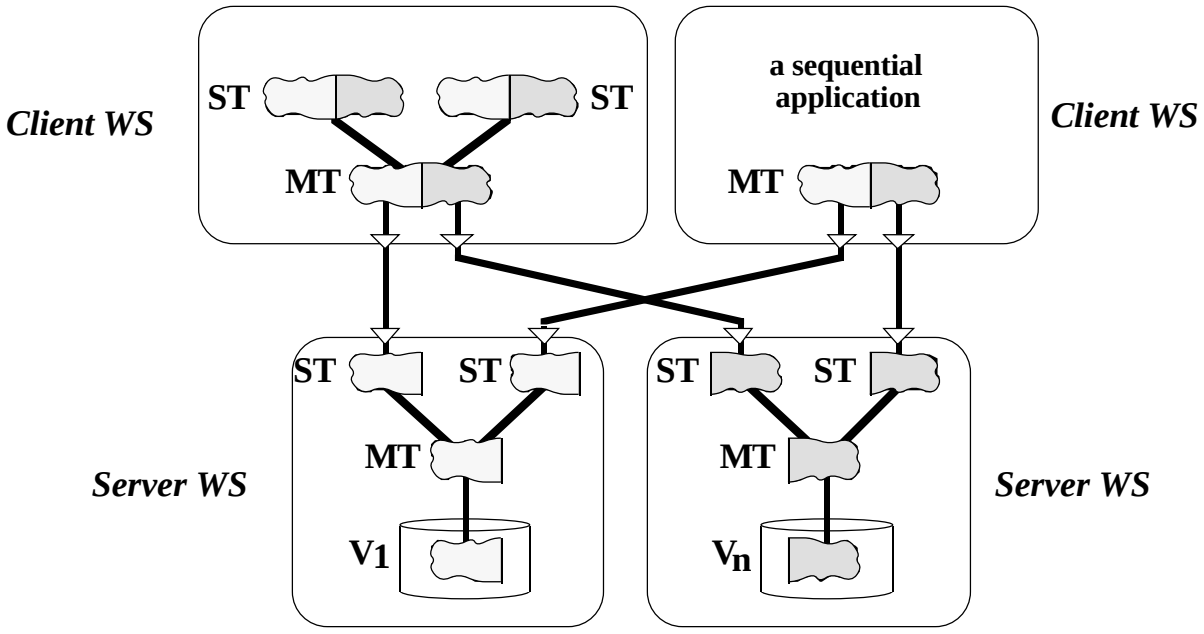
Client / Server Database "à la SHORE"

WAN Front-end

Client / Server Database Architecture

the database is distributed on N servers :

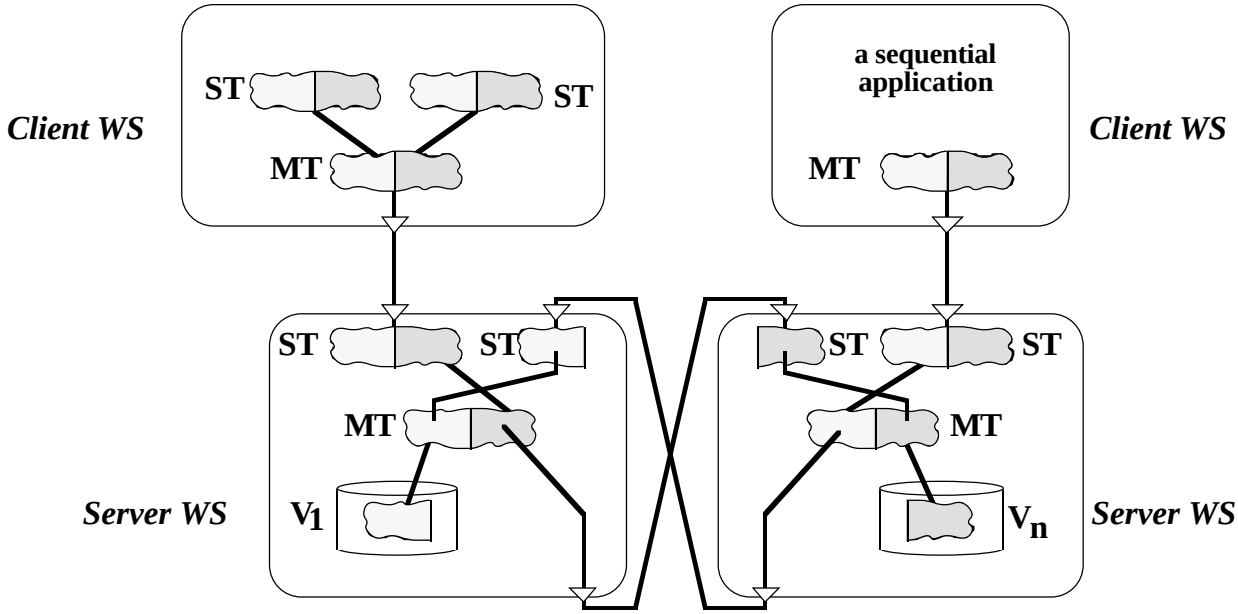
each client is directly connected to N servers



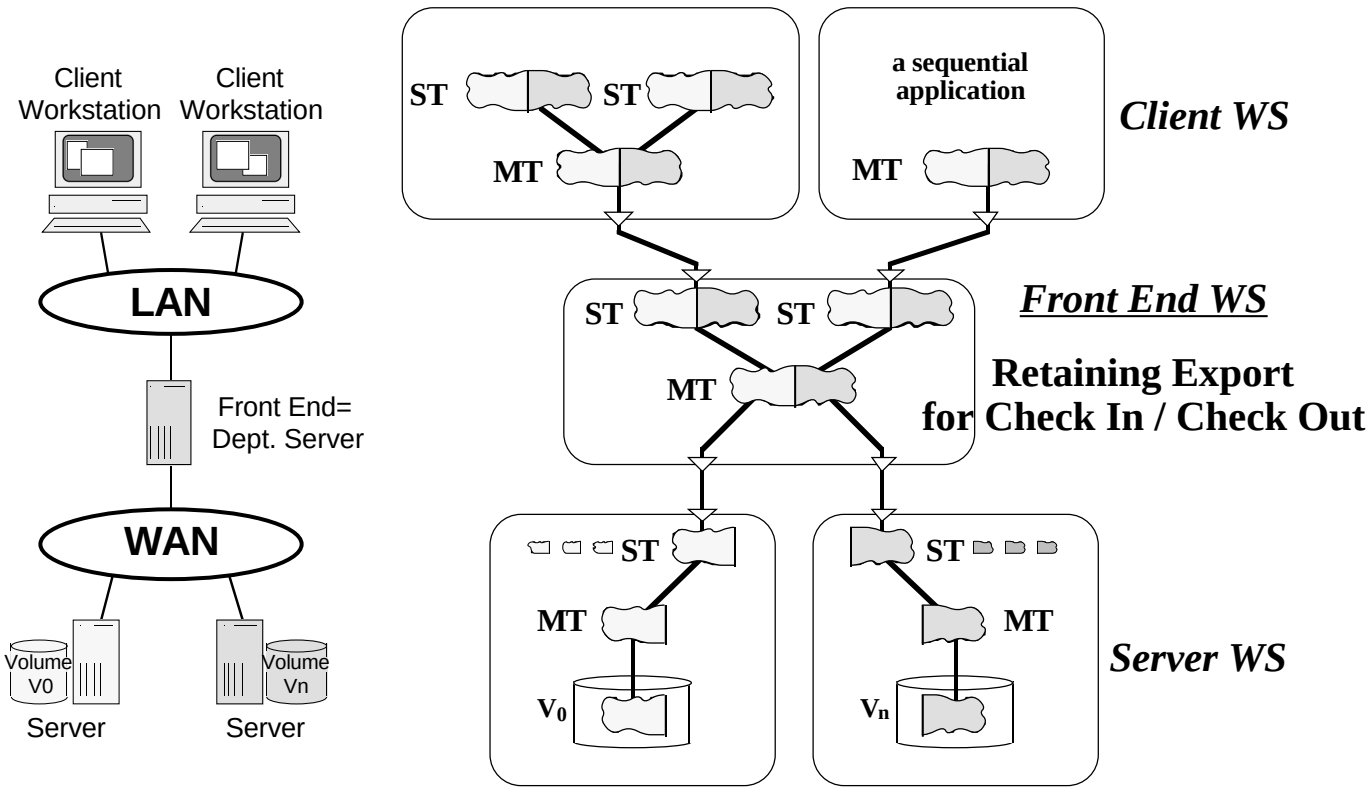
Client / Server “à la SHORE” (U. of Wisconsin)

each server exports its part to the N-1 servers

and merges all parts and export them to its connected clients



Client / Server with WAN Front-End



Mixing Data and Query Shipping

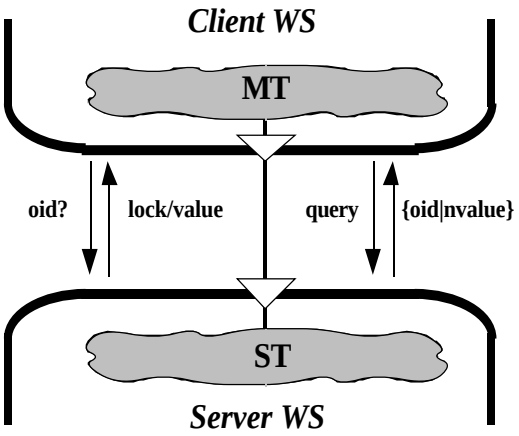
Data Shipping

- moves confidential data to unsecured clients

Query Shipping

- involves bigger servers

back to the Mainframe Model



Merging Data-Query Shipping =

- 2 kinds of access permissions
  - *1 for operation on server*
  - *1 for data export*
- consistency data between Client MT and Server ST

Implementation

2 implementations

WEA - YOODA

True MultiThreading

parallelism (CPU,IO) and asynchronous communications

Memory Mapping

page buffering, implicit 2P locking

Callback 2 Phase Locking

lock cache in the client

C++ interface (ODMG like)

Futures Works

- **Merge of WEA-YOODA**

- **Applications**
  - **Document Database on heterogeneous networks**
  - **Cooperative Information Systems**
  - **possible extension to Asynchronous Video Stream**

Mixing Data and Query Shipping

Data Shipping

➠ **moves confidential data to unsecured clients**

Query Shipping

➠ **involves bigger servers**

back to the Mainframe Model

Merging Data-Query Shipping =

☞ **2 kinds of access permissions**
- ***1 for operation on server***
- ***1 for data export***

☞ **consistency data between Client MT and Server ST**