WebCom: A Web Based Distributed Computation Platform

CUC, National University of Ireland, University College Cork, Ireland
Presenter - Pushpinder Kaur CHOUHAN

31 March 2005

Outline

Outline

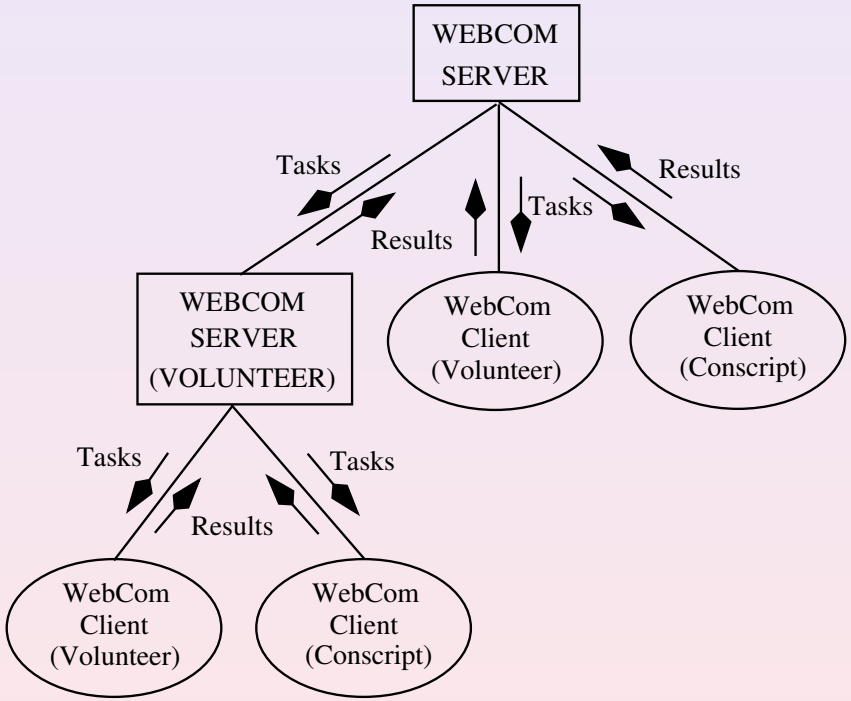
WebCom - Web Communicator

Developed by : Centre for Unified Computing, Cork, Ireland.
Based on: client/slave model with multi-tier bidirectional topology
Developed for: Execution of programs expressed as Condensed Graphs.
Environment: Distributed environment of the World Wide Web and the Internet.
Achieved : Utilizing Java applets, with different applet types for different tasks.
Communication: Remote Method Invocation and Object Request Broker

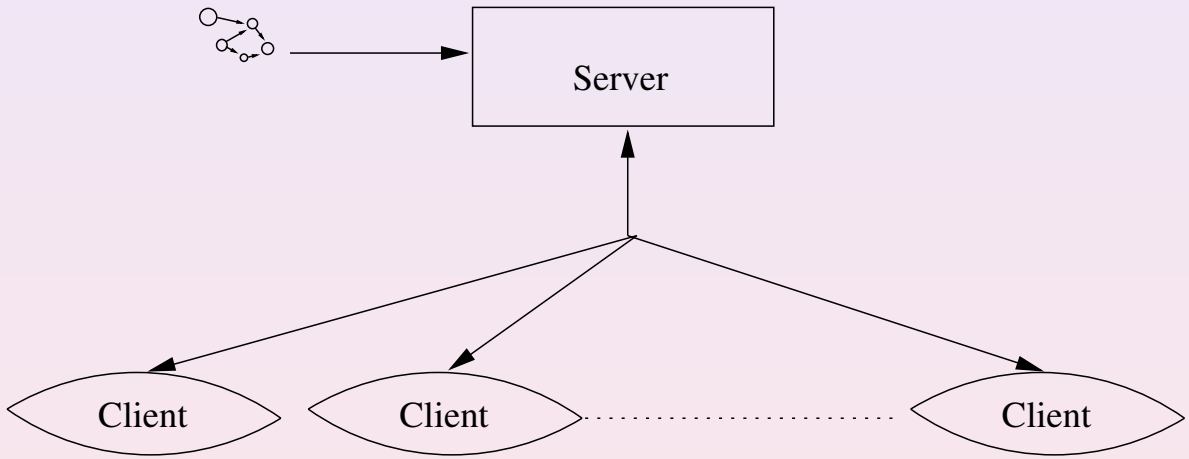
WebCom Components

- Servers: submit tasks
  - implemented as a multi-threaded Java application
  - only trusted machines
  - if no client, execute tasks
- Clients: executes the tasks
  - java applet is downloaded
  - Volunteer or Conscript
  - Promotable or Non-promotable

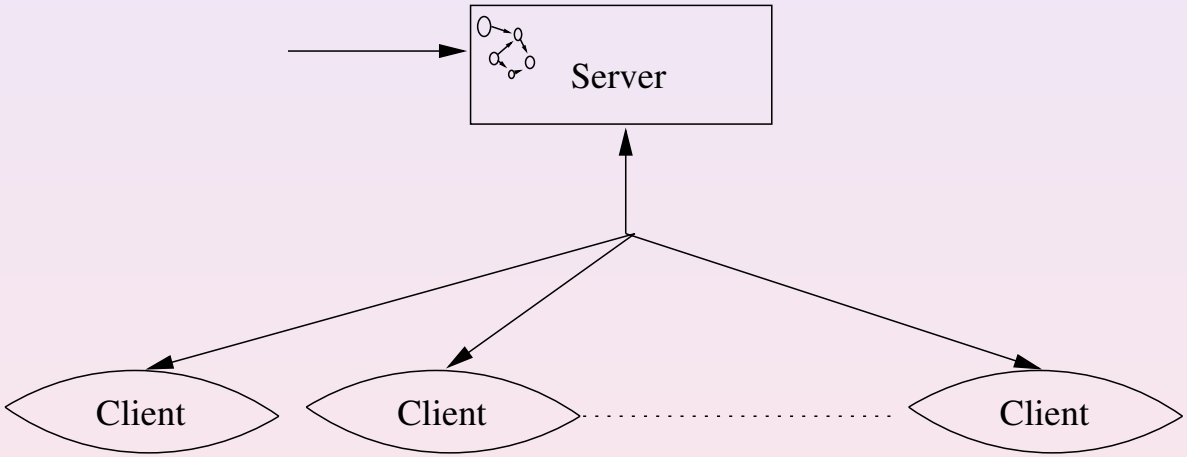
WebCom Client/Server connectivity



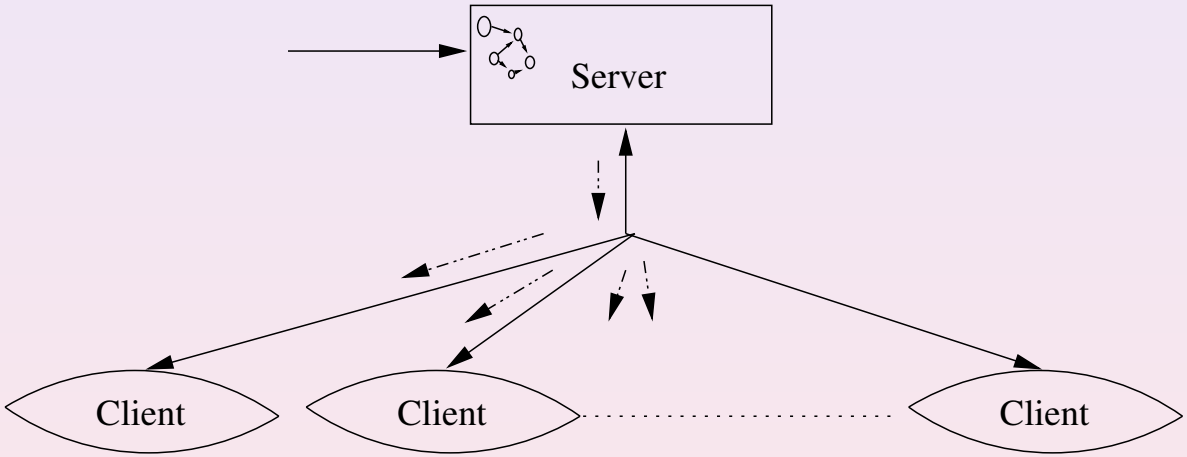
Evolution of P2P network



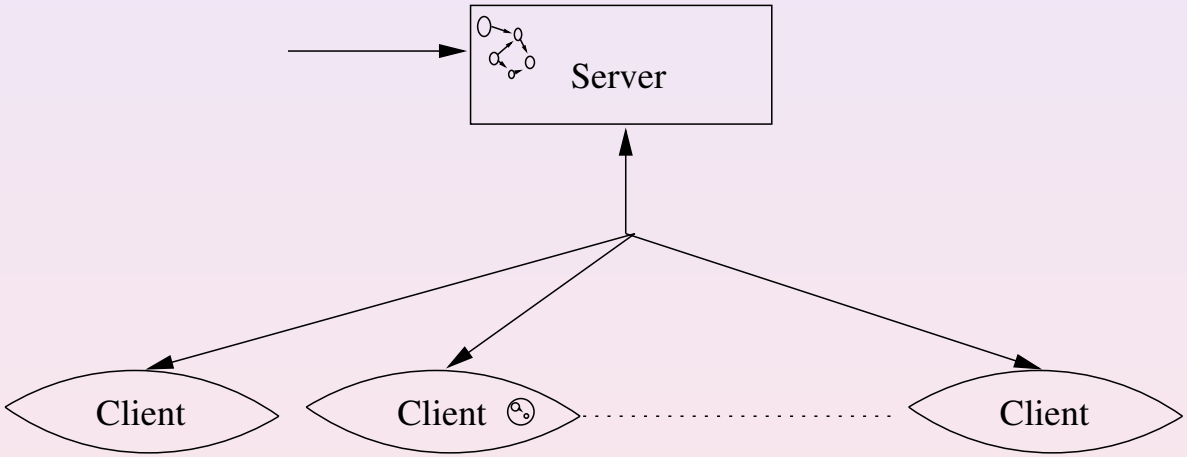
Evolution of P2P network



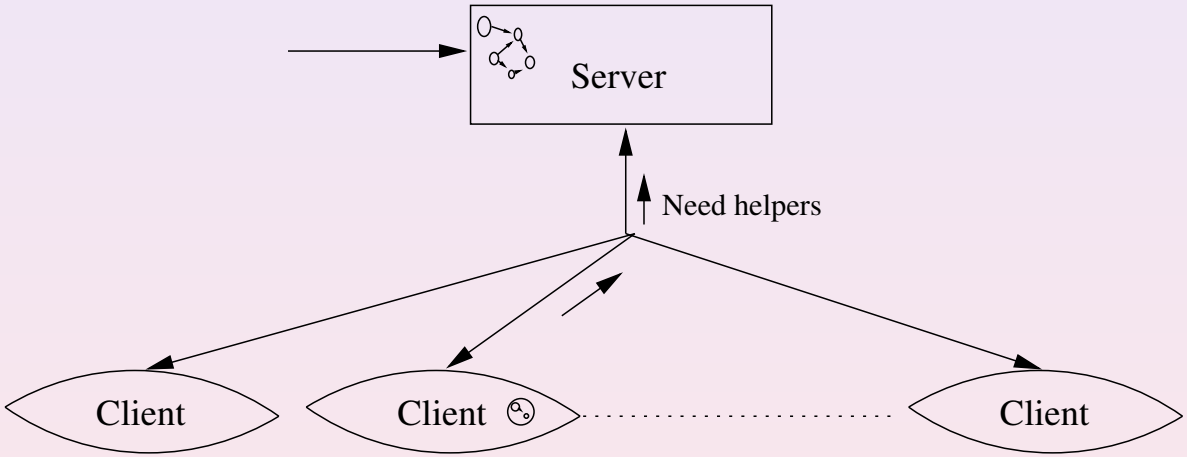
Evolution of P2P network



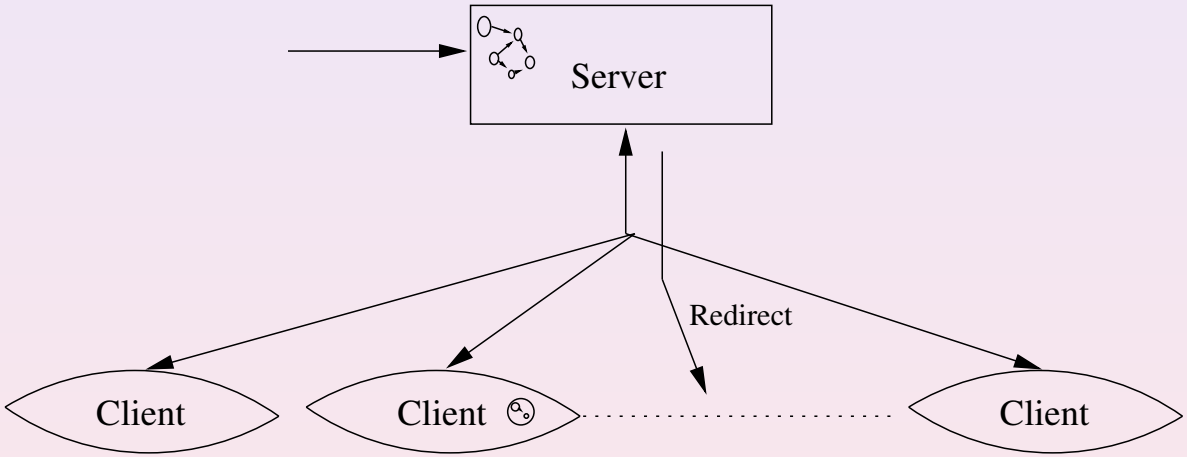
Evolution of P2P network



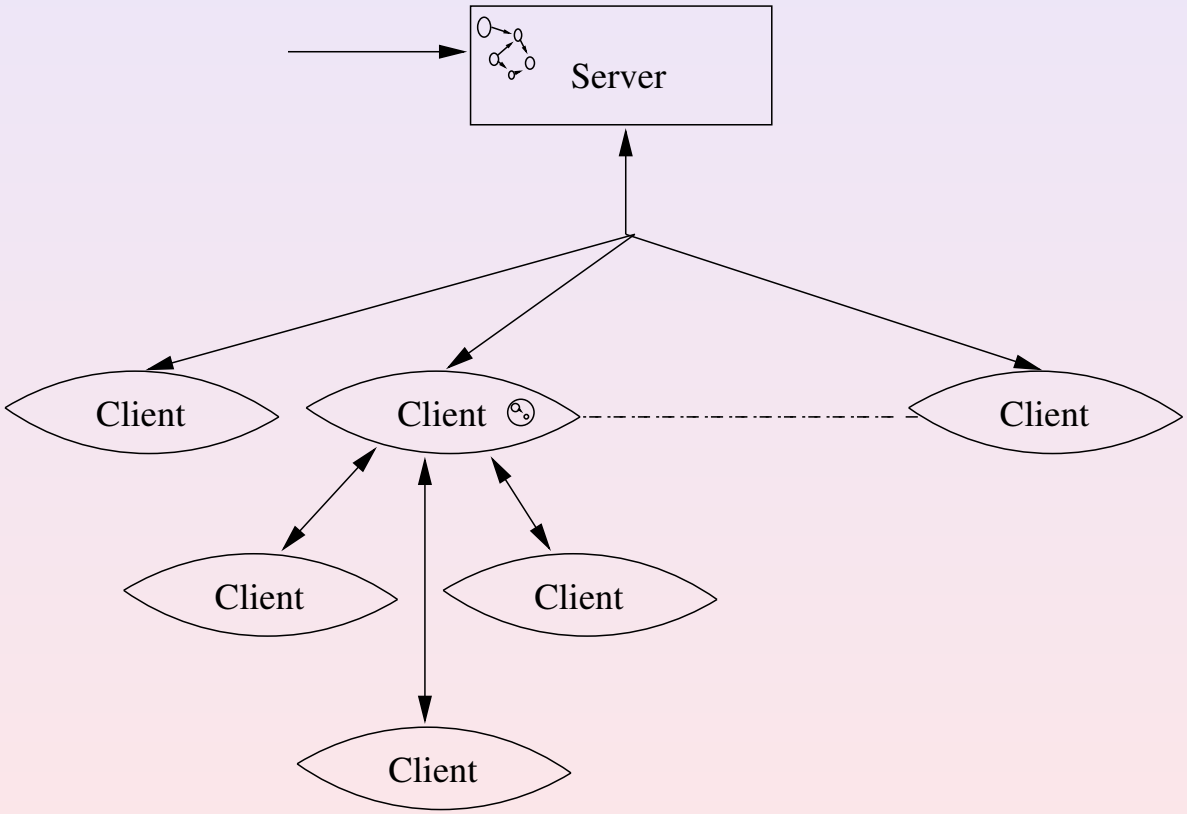
Evolution of P2P network



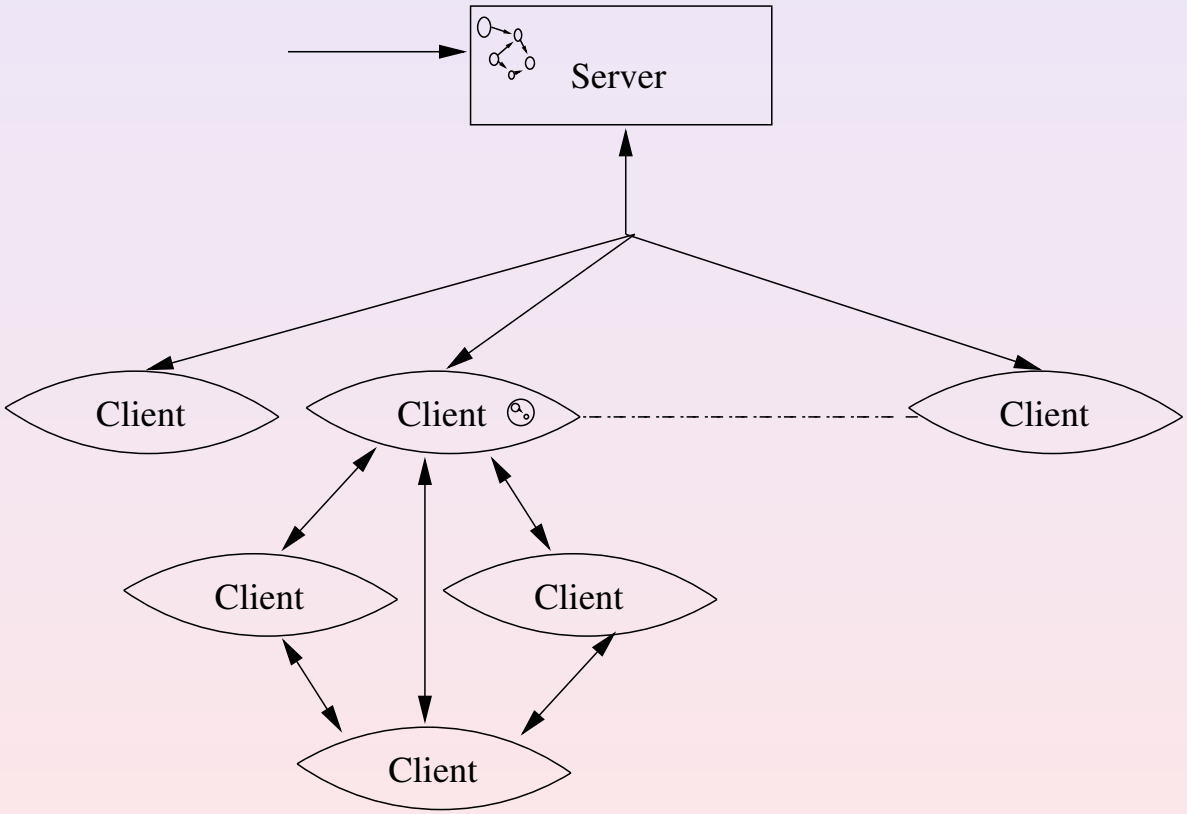
Evolution of P2P network



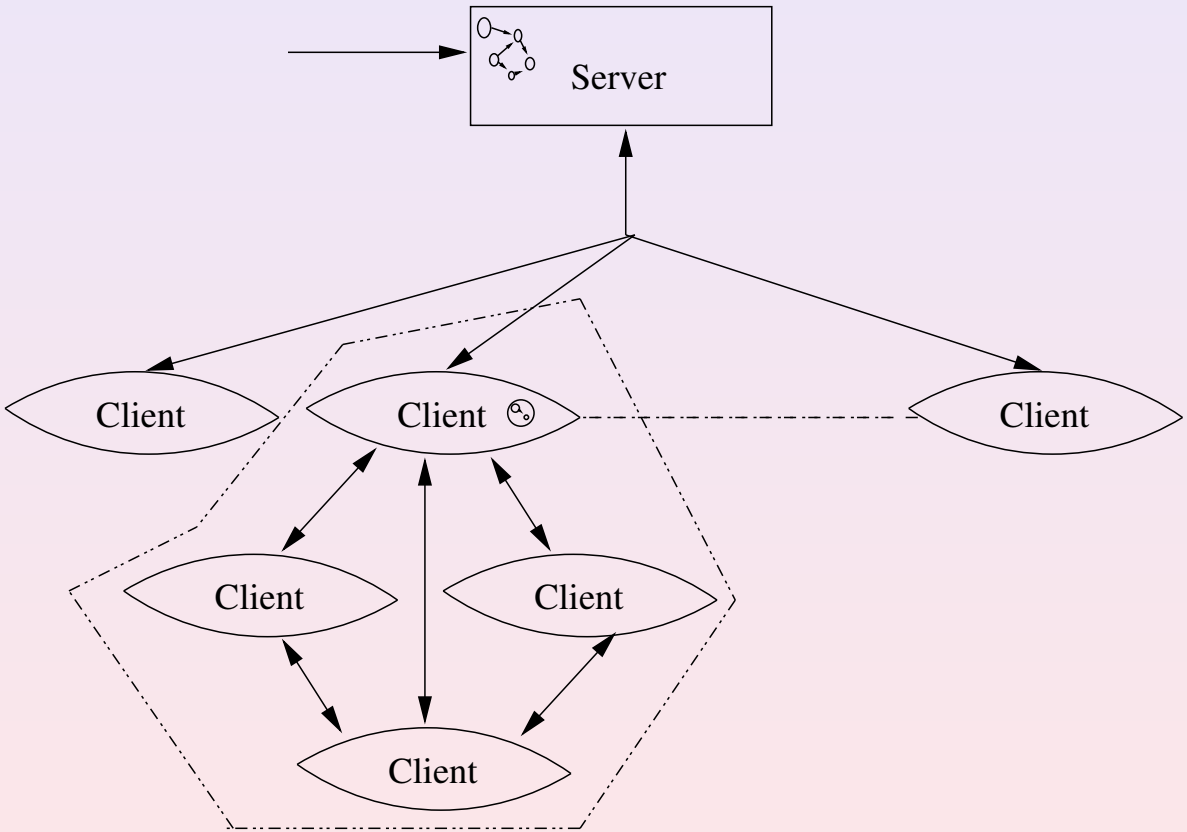
Evolution of P2P network



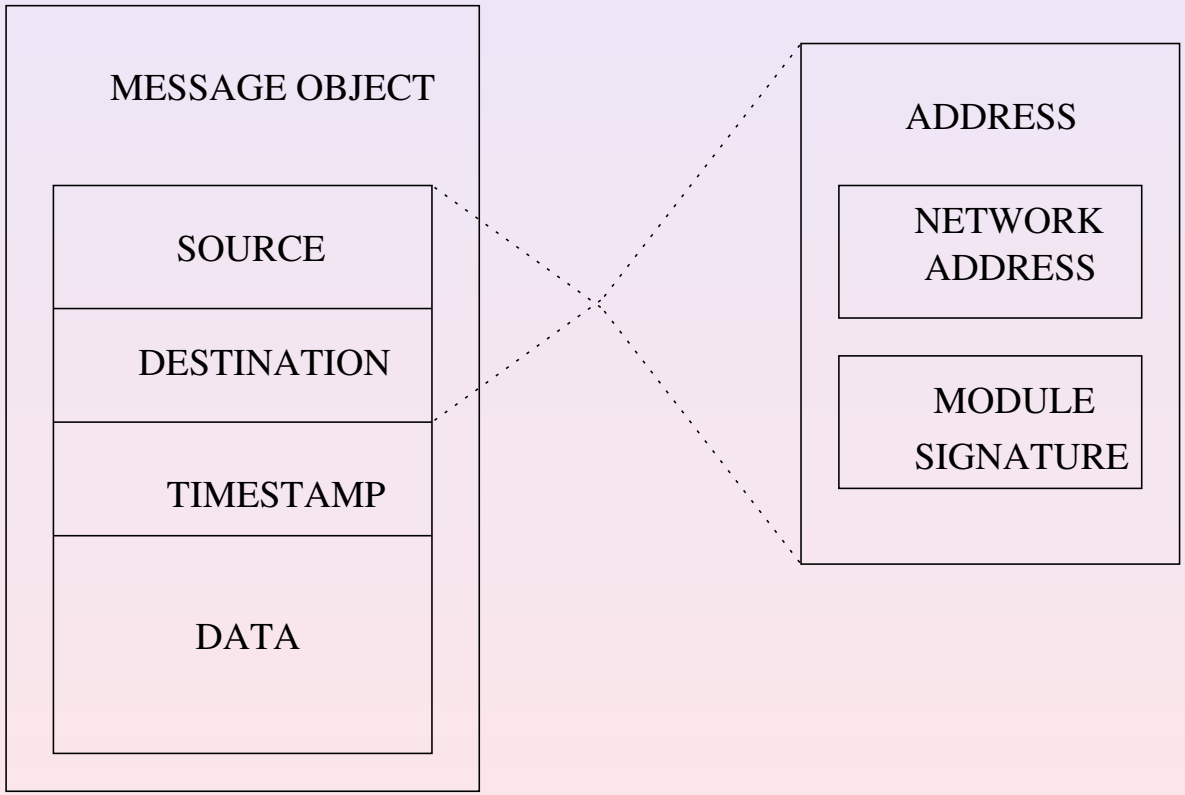
Evolution of P2P network



Evolution of P2P network



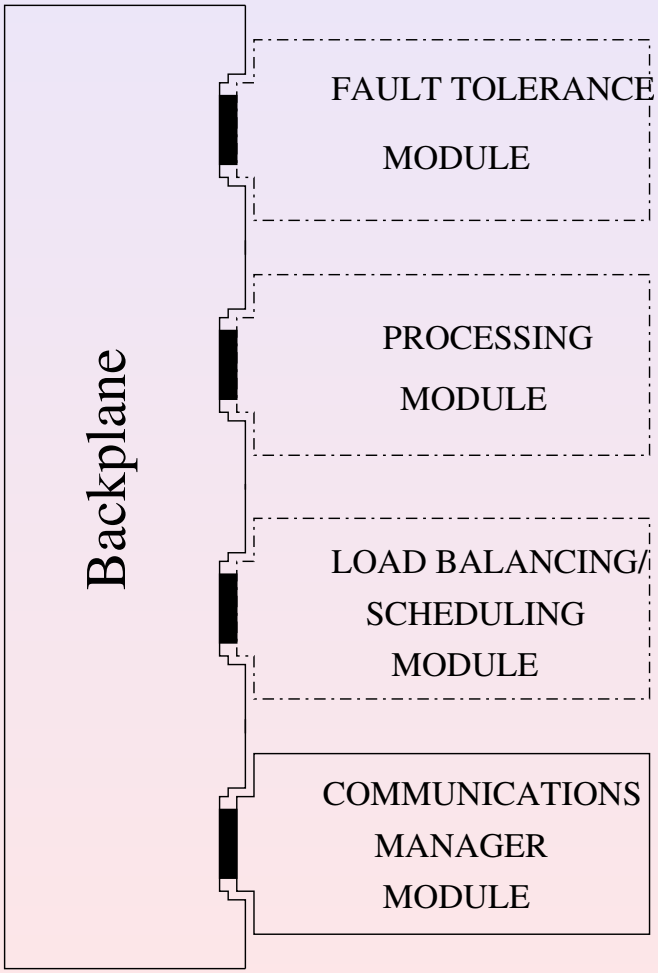
The message object

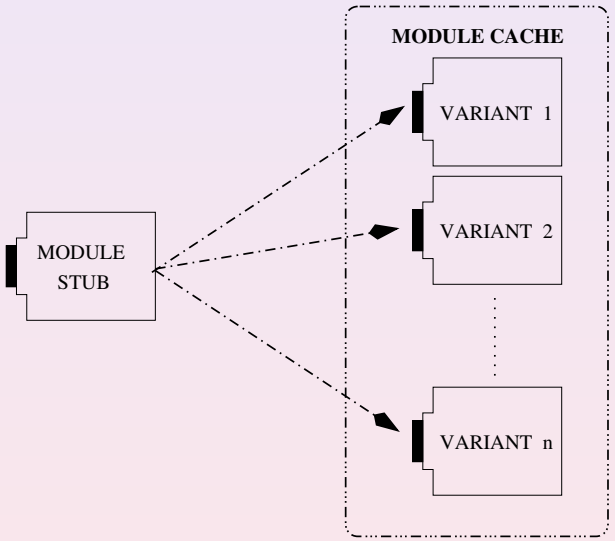


WebCom Architecture

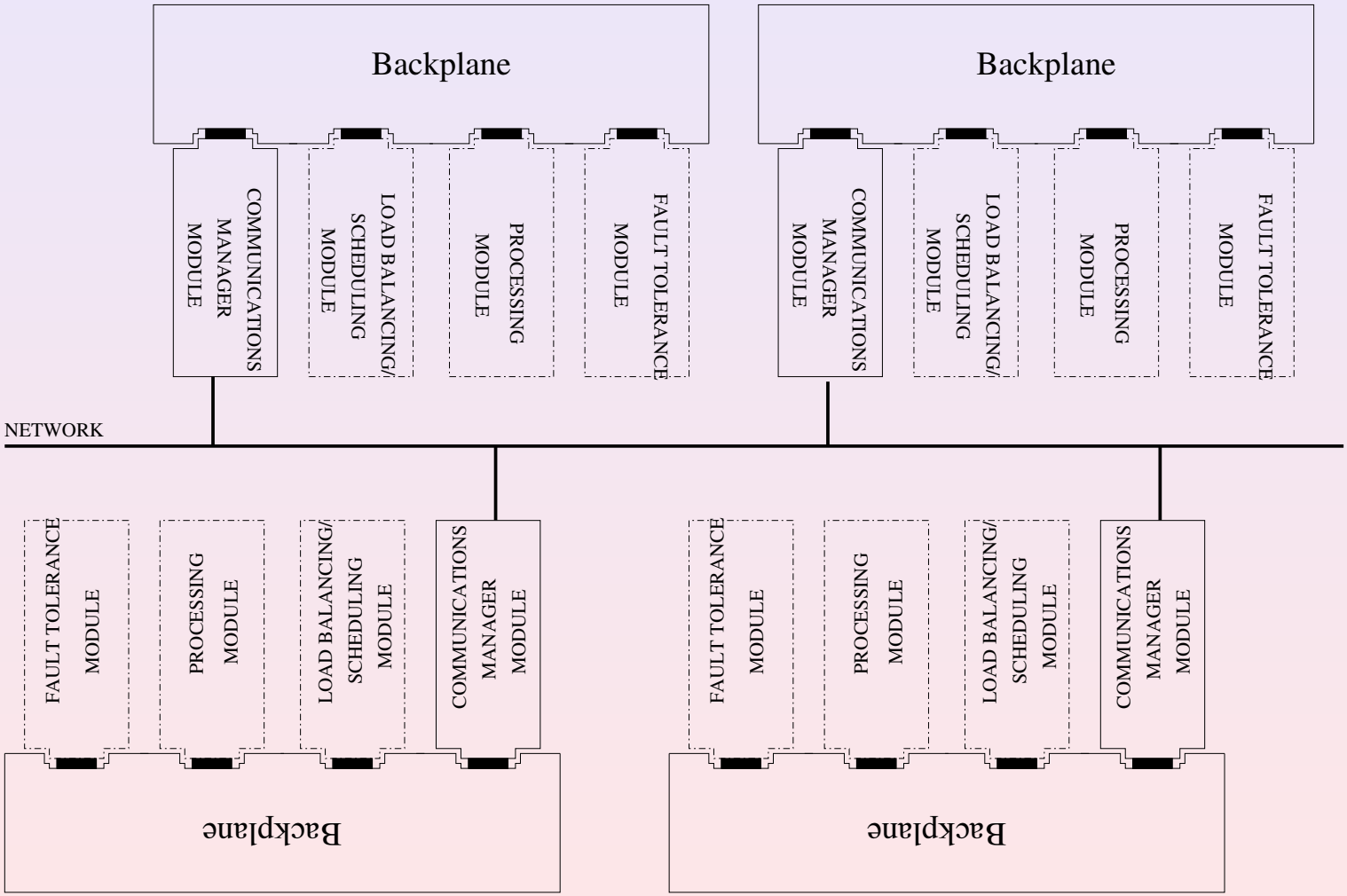
- Computing Platform (Abstract Machine)
  - Manage the network connections
  - uncover and schedule tasks
  - load balancing across the system
  - faults handling
  - execution of tasks
- Development Environment
  - develops the applications
  - specific for Condensed Graphs

Abstract Machine





Abstract Machine connected over a network



Outline

Functions of Backplane

- Initialisation - initial static configuration
- Interconnection - between modules
- Manage queue - processing tasks queue
- Pass tasks between modules

Processing Module

- inspects the message data
- determine the type of message
- add tasks to queue for execution
- execute the task
- send results to non local address
- list of partial results is maintained
- pass the uncovered tasks to the Backplane

Strandard Communication Manager Module

- Maintains two (client + server) distinct lists of connection descriptors
- descriptor facilitates the bidirectional communication between client and server.
- each client have one descriptor
- utilization falls with connections > 30

Advanced Communication Manager Module

- Maintains two descriptors
  - one maintaining an arbitrary number of incoming connections
  - other maintaining an arbitrary number of outgoing connections
- each connection have its own state
  - Available / Unavailable
  - Busy
- A client can indicate different status on each connection

Scheduling Tasks

Push Mechanism: Server transmit tasks

- check the status of client
- client should have sufficient buffer space

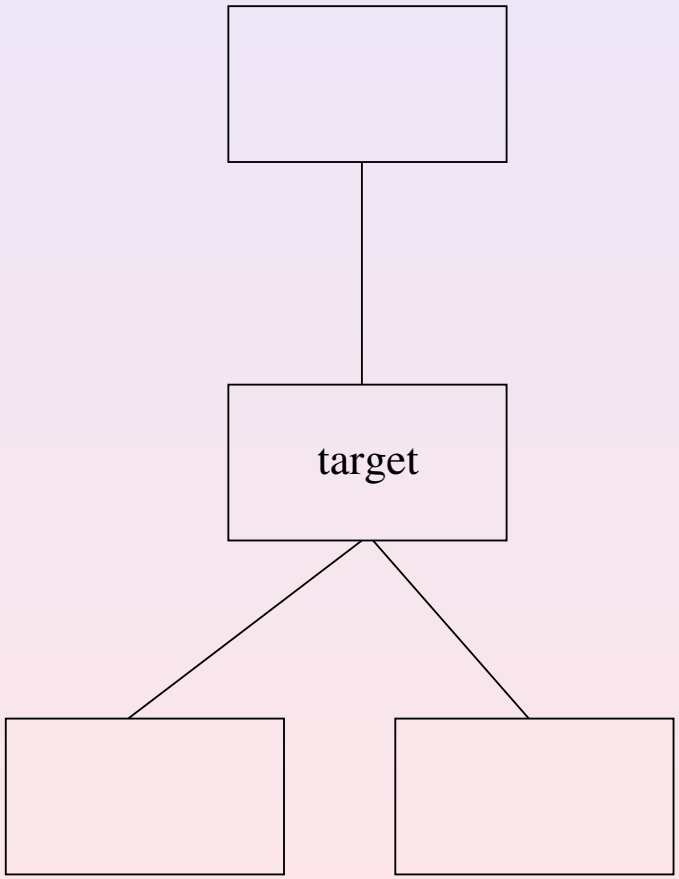
Pull Mechanism: Client take task for execution

- server maintained a queue of scheduled tasks for each client
- queues can be rebalanced for better scheduling

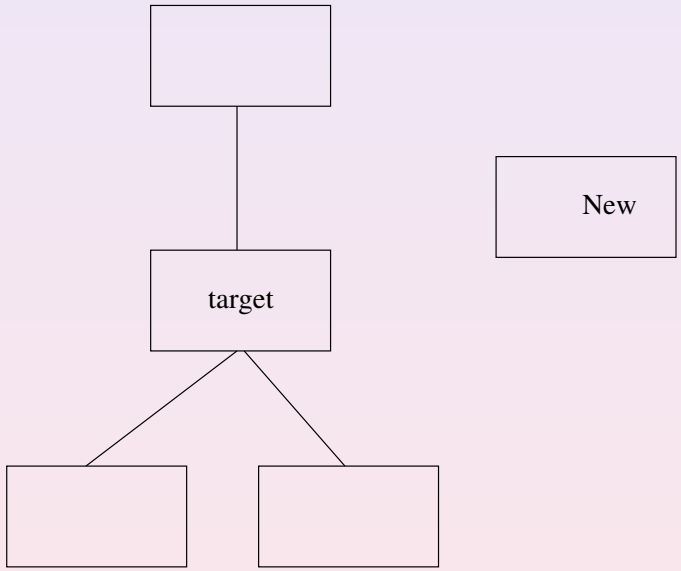
Machine depart from WebCom

- Soft disconnection
  - machine process scheduled tasks offlined
  - temporarily unavailable
  - tasks can be scheduled to be transmitted later
- Hard disconnection
  - connection is closed
  - removed from the descriptors
  - pending tasks will be handled by Fault Tolerance Module

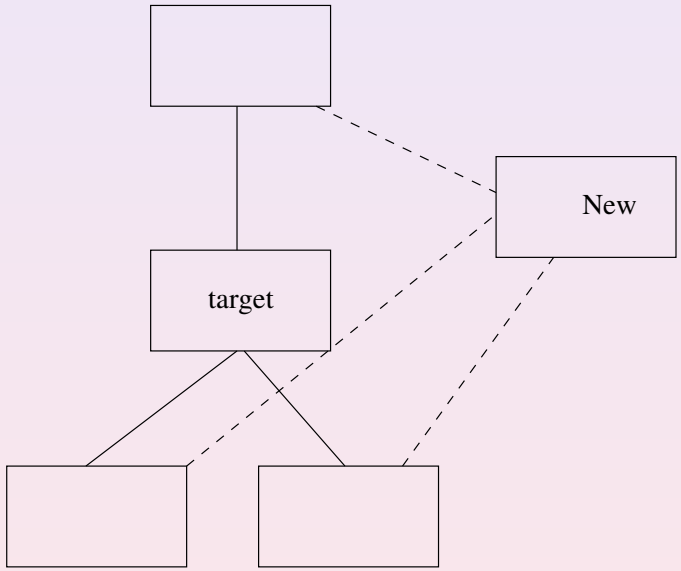
Processor Replacement Procedure



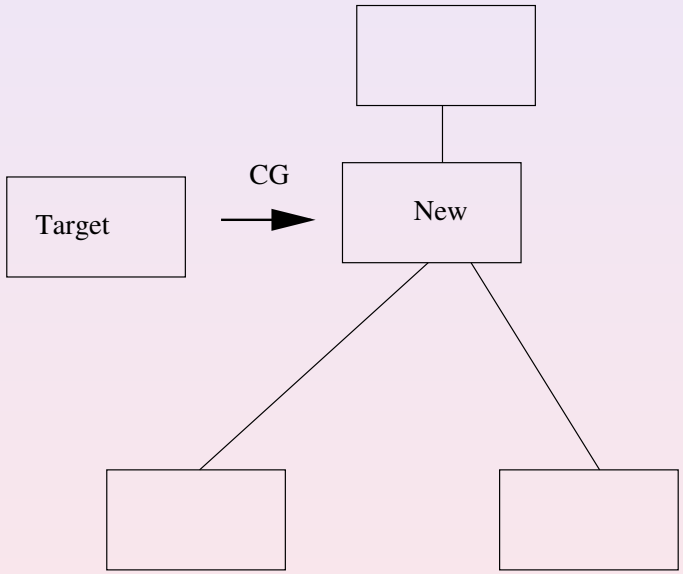
Processor Replacement Procedure



Processor Replacement Procedure



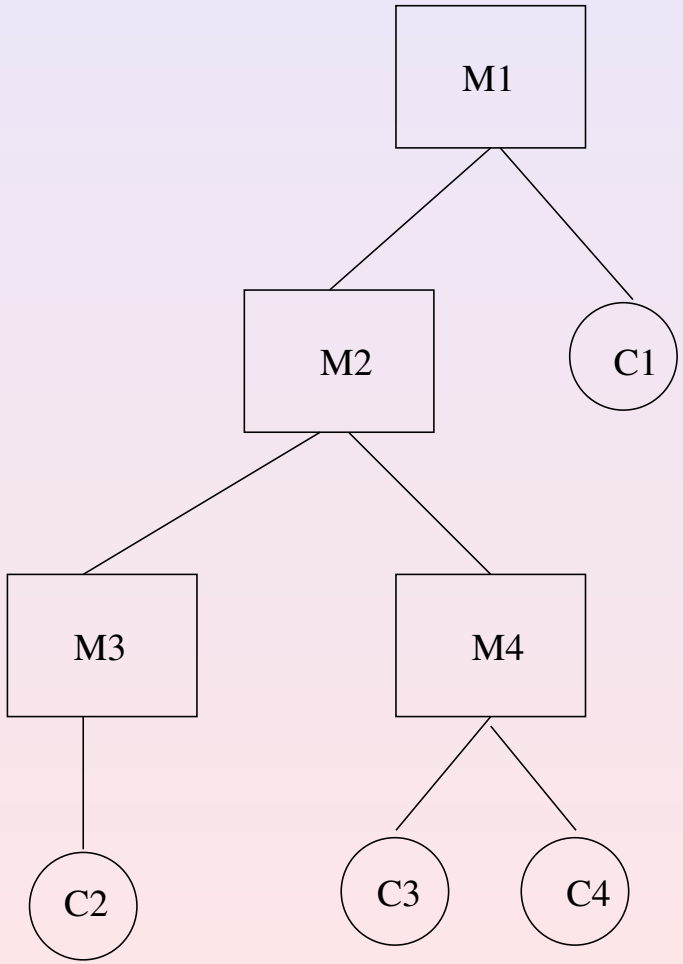
Processor Replacement Procedure



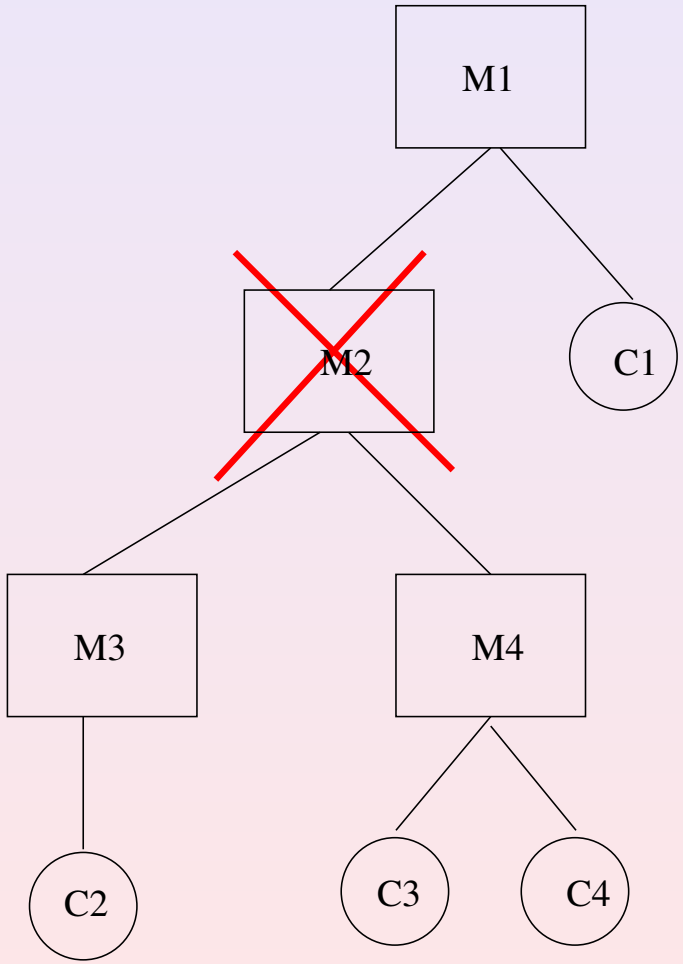
Information on each processor

- knows nodes - parent, children, grandparent and peers
- history list - purge periodically
- size of history list - max no. of instruction in each CG

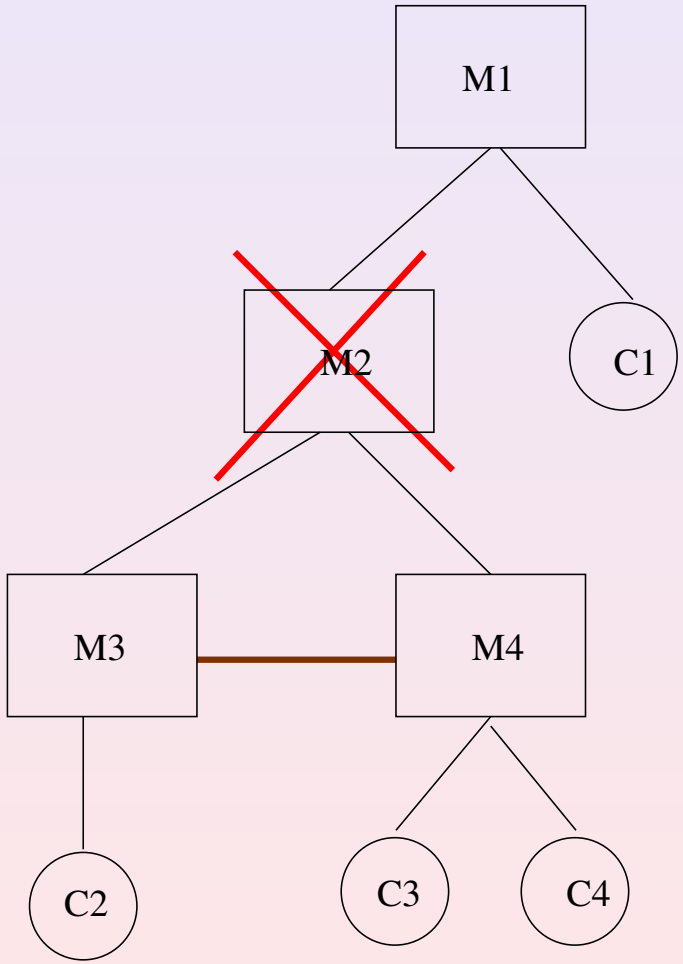
Fault Recovery Procedure



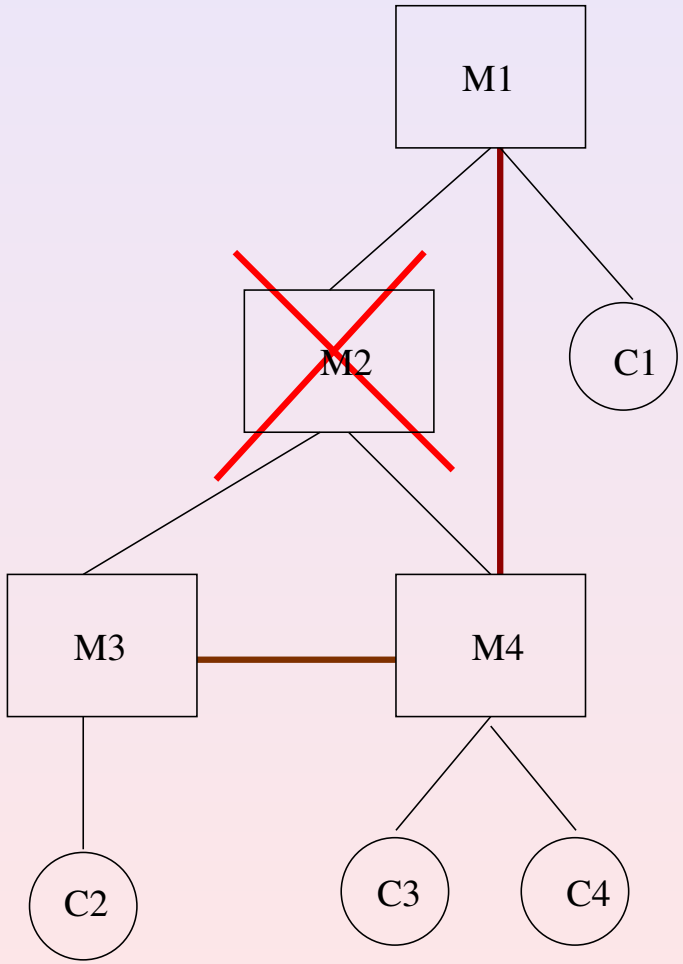
Fault Recovery Procedure



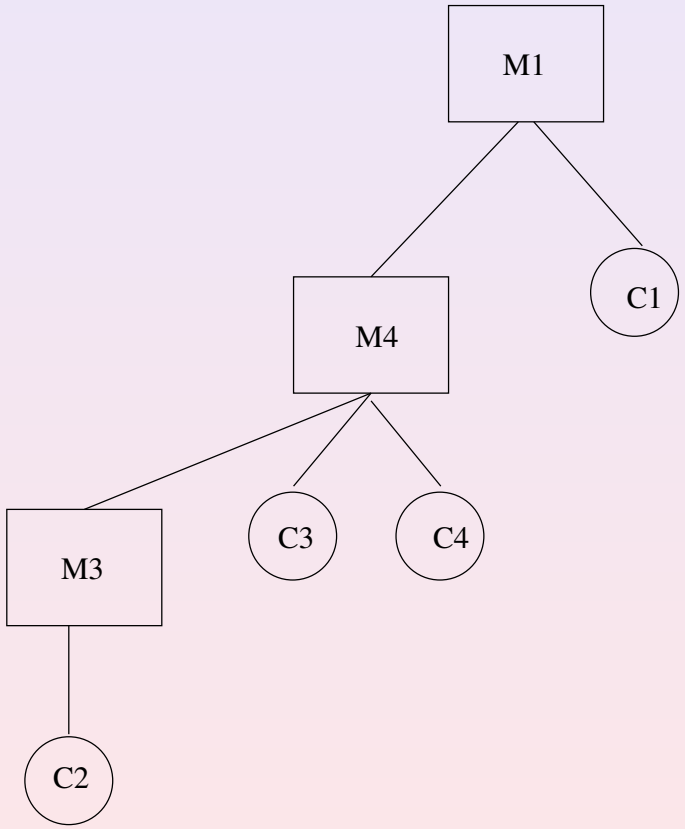
Fault Recovery Procedure



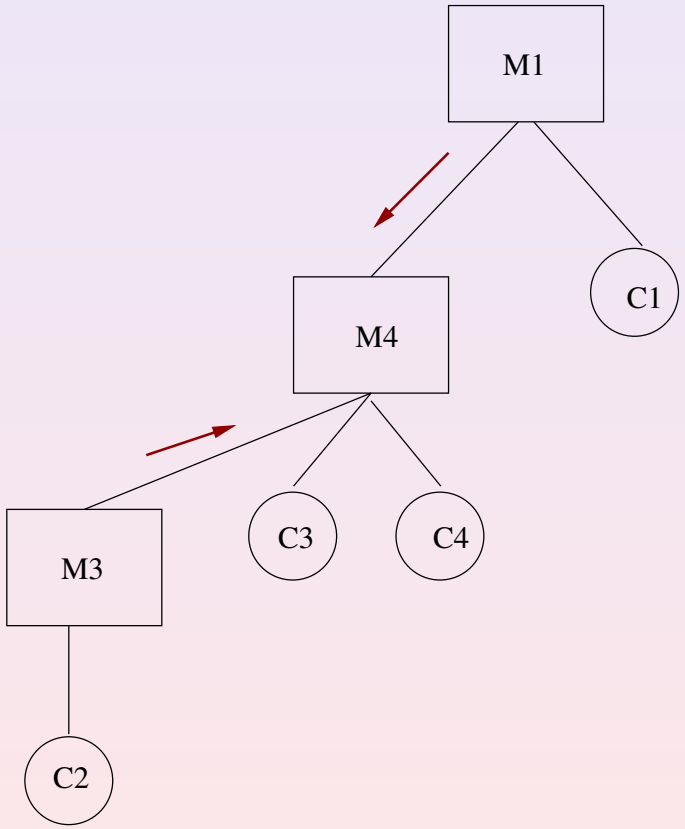
Fault Recovery Procedure



Fault Recovery Procedure



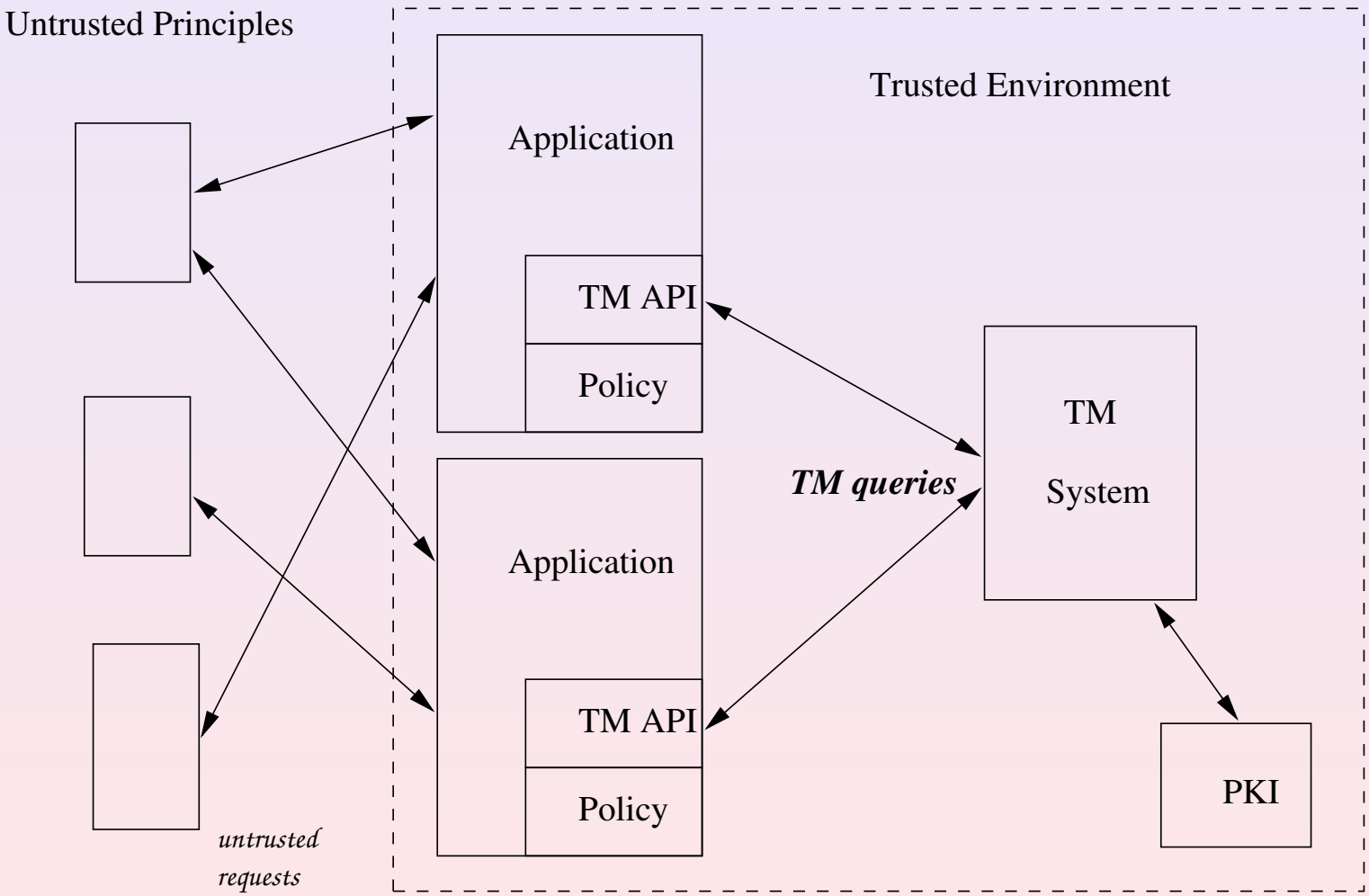
Fault Recovery Procedure



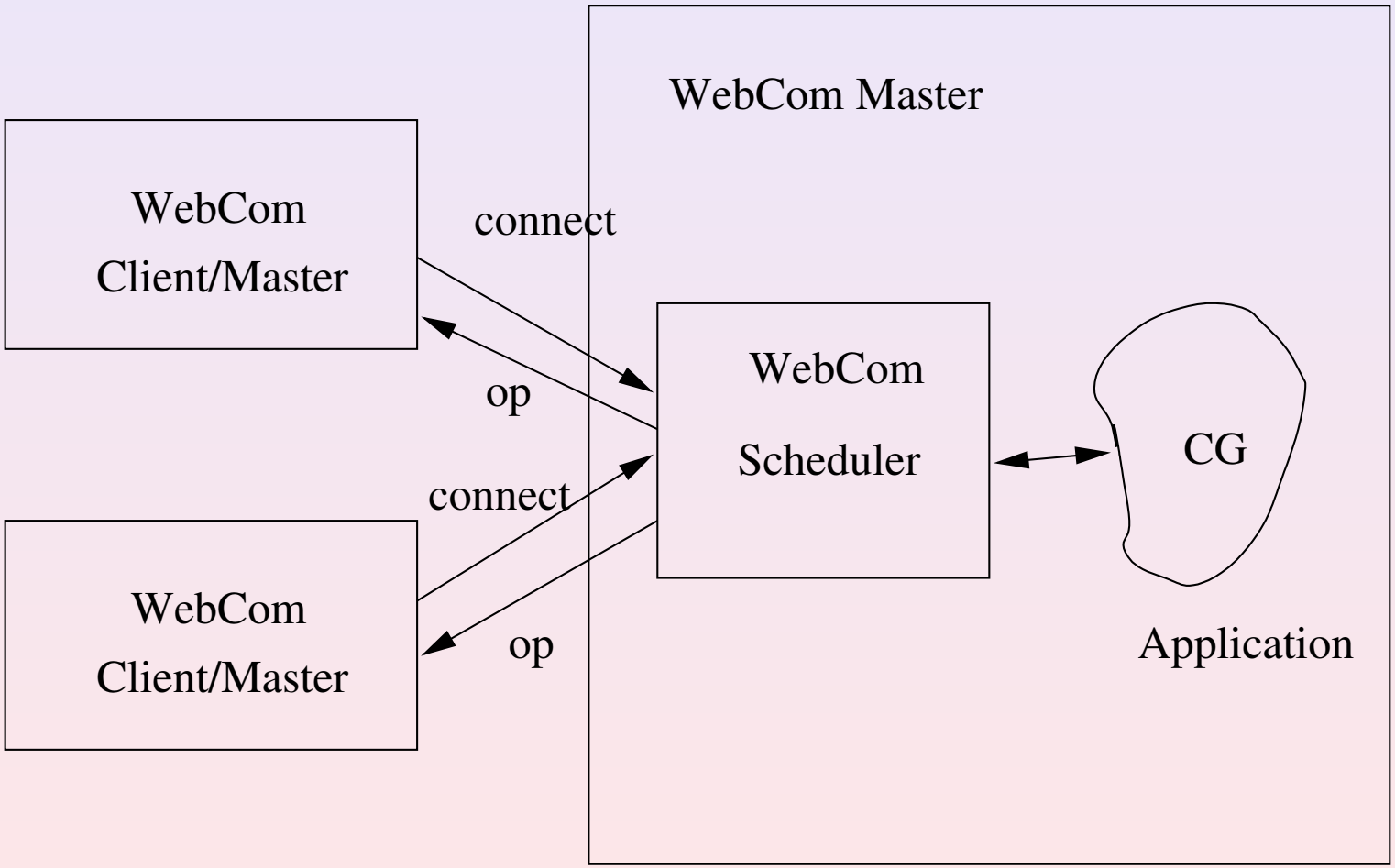
Trust Management System

- TM- construct and interpreat the trust relationships among public keys.
- Policy - credentials, representing the keys that are trusted
- Credentials - specify delegation of authority among public keys
- PKI - manage the credentials
- Master and Client Authorization

KeyNote Trust Management System

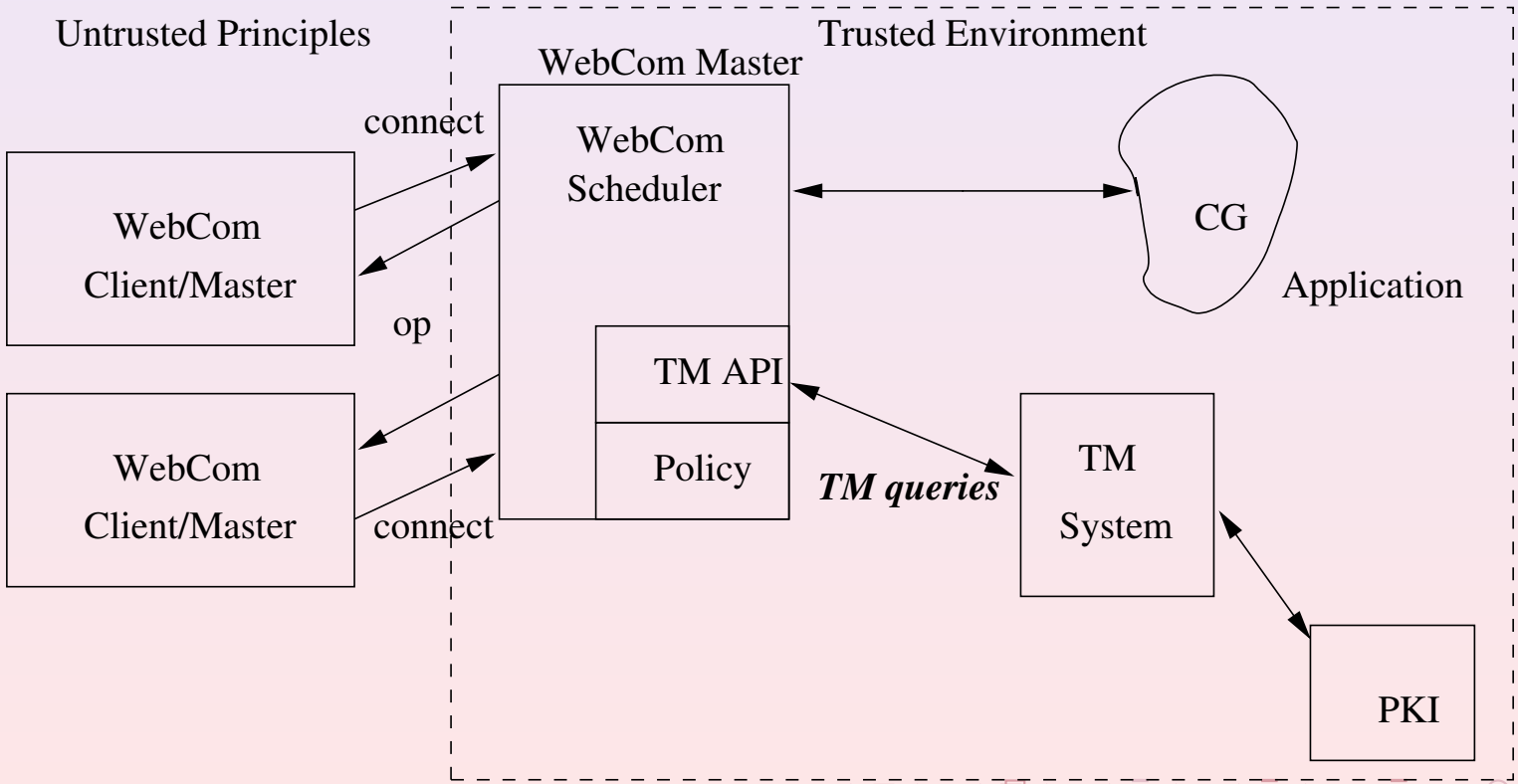


The WebCom Distributed System



KeyNote + WebCom

- X509 certificates are used for authentication
- KeyNote credentials are used for authorisation



load balancing strategies

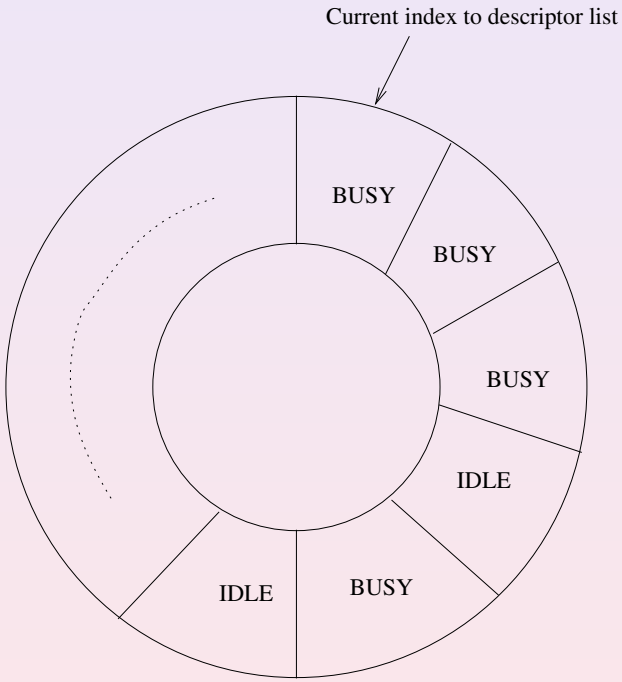
General

- static or dynamic
- centralised or distributed
- active or passive

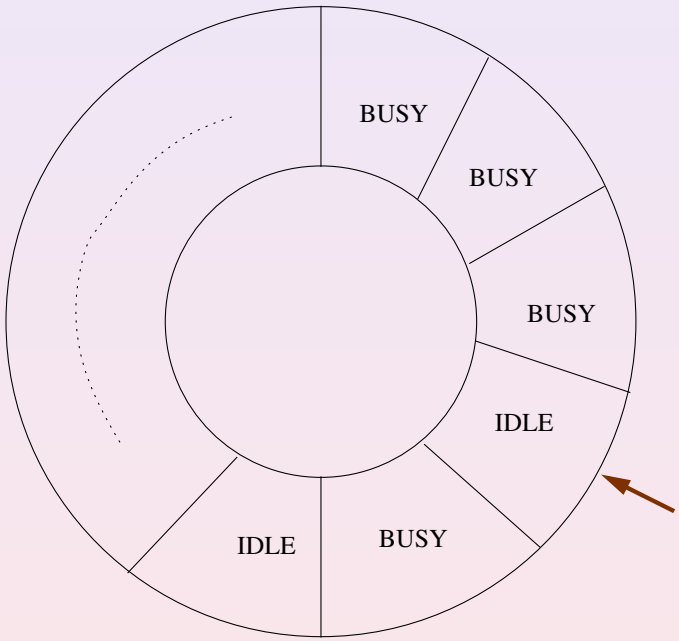
WebCom strategies

- dynamic
- centralised
- passive

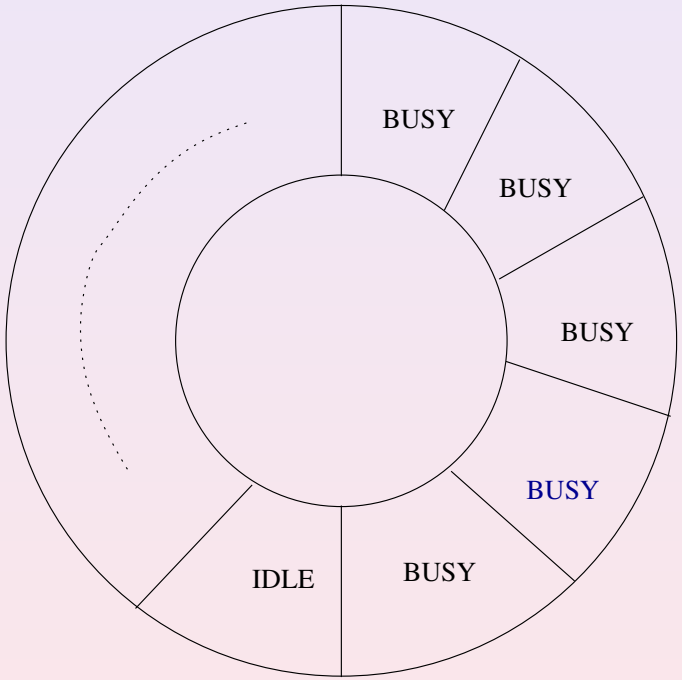
Round Robin fashion



Round Robin fashion



Round Robin fashion

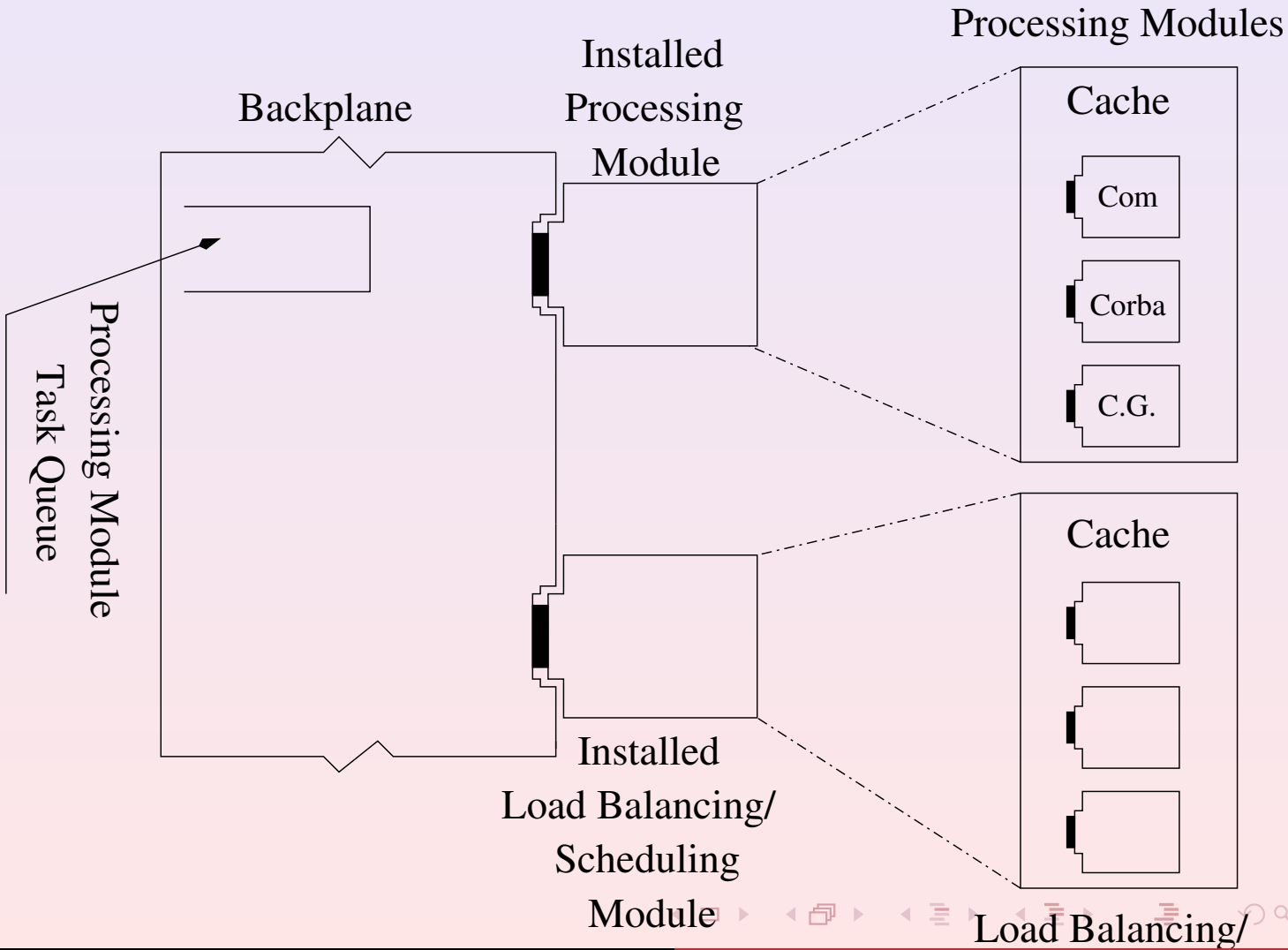


Scheduling - Locally or Remotely

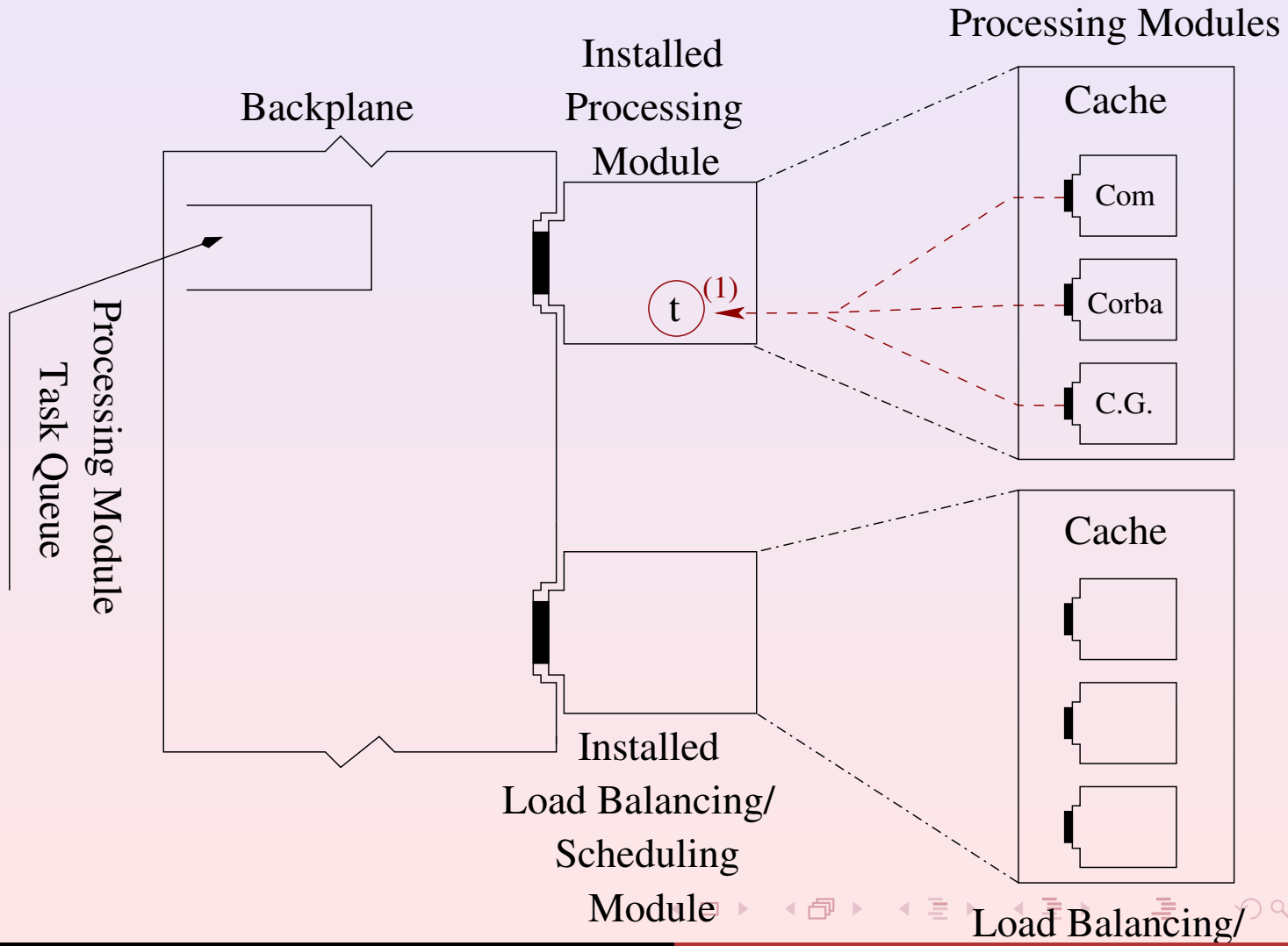
- start with round robin
- expected execution time matching
  - $Lc_{(j)} = Tax_{(i)}c_{(j)} \times G_{(i)}$
  - $Tax_{(i+1)}c_{(j)} = Lc_{(j)} \times G_{(i+1)}$
  - $\Delta Exp_{(i,j)} = Tax_{(i+1)}c_{(j)} - G_{(i+1)}$
- 0 - doesn't matter
- Positive - Locally
- Negative - Remotely

Outline

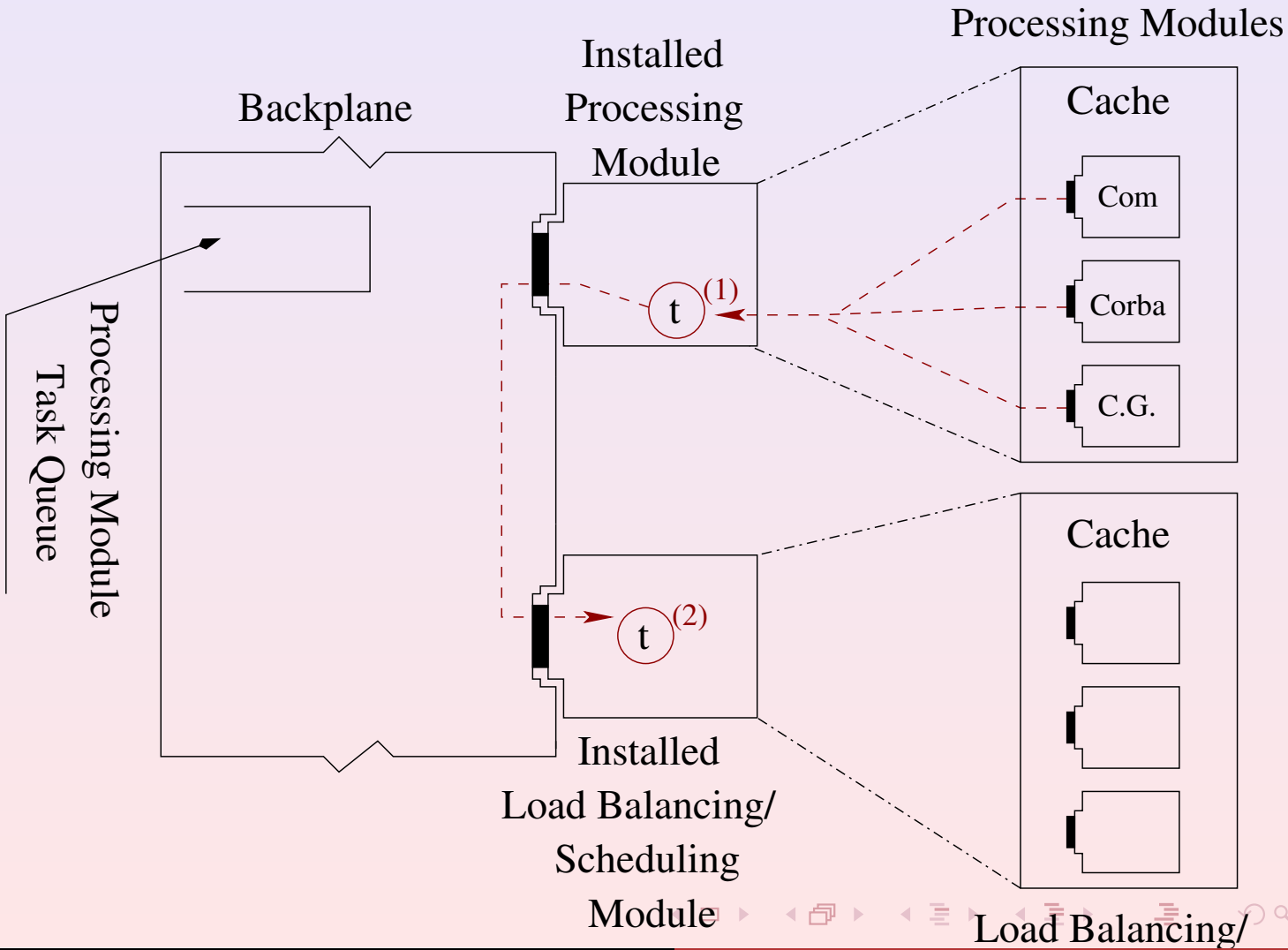
Local execution of task



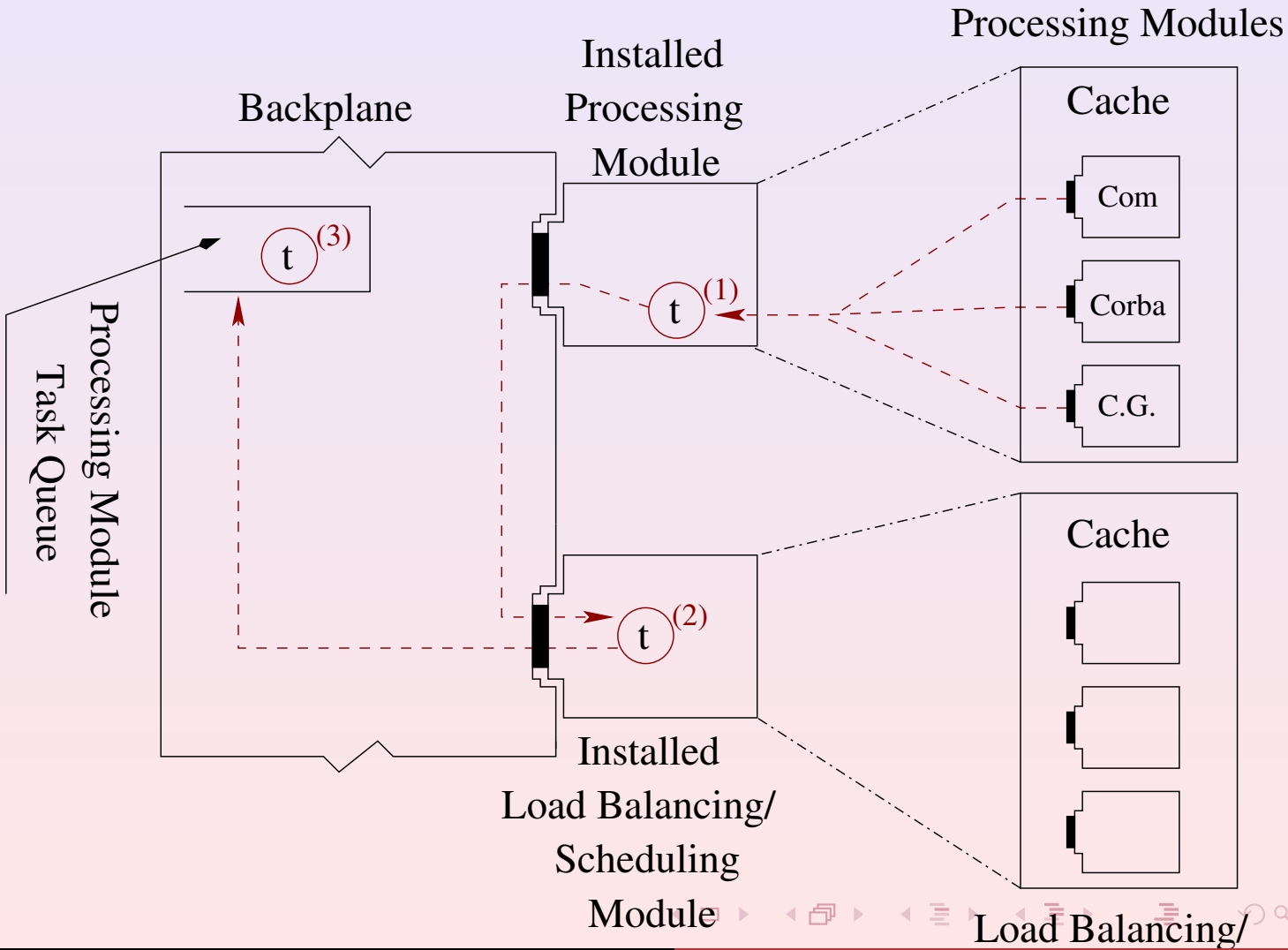
Local execution of task



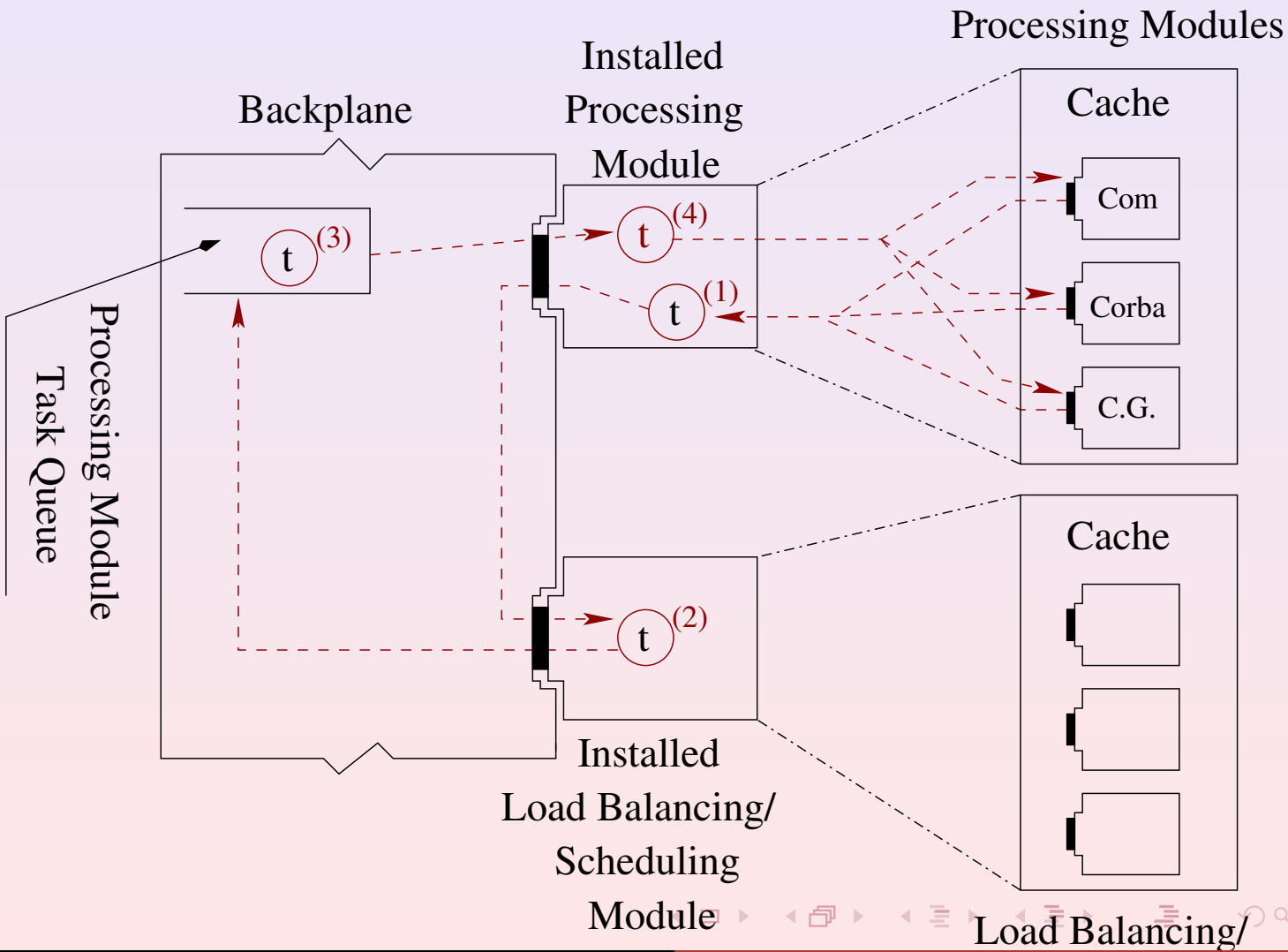
Local execution of task



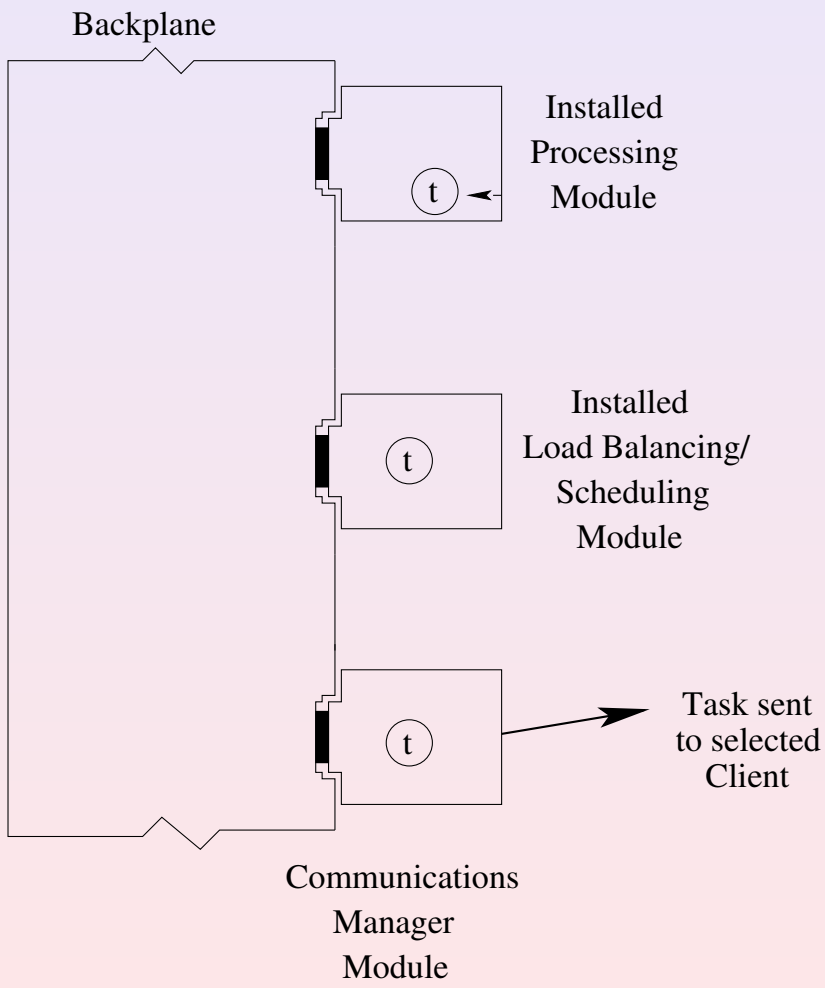
Local execution of task



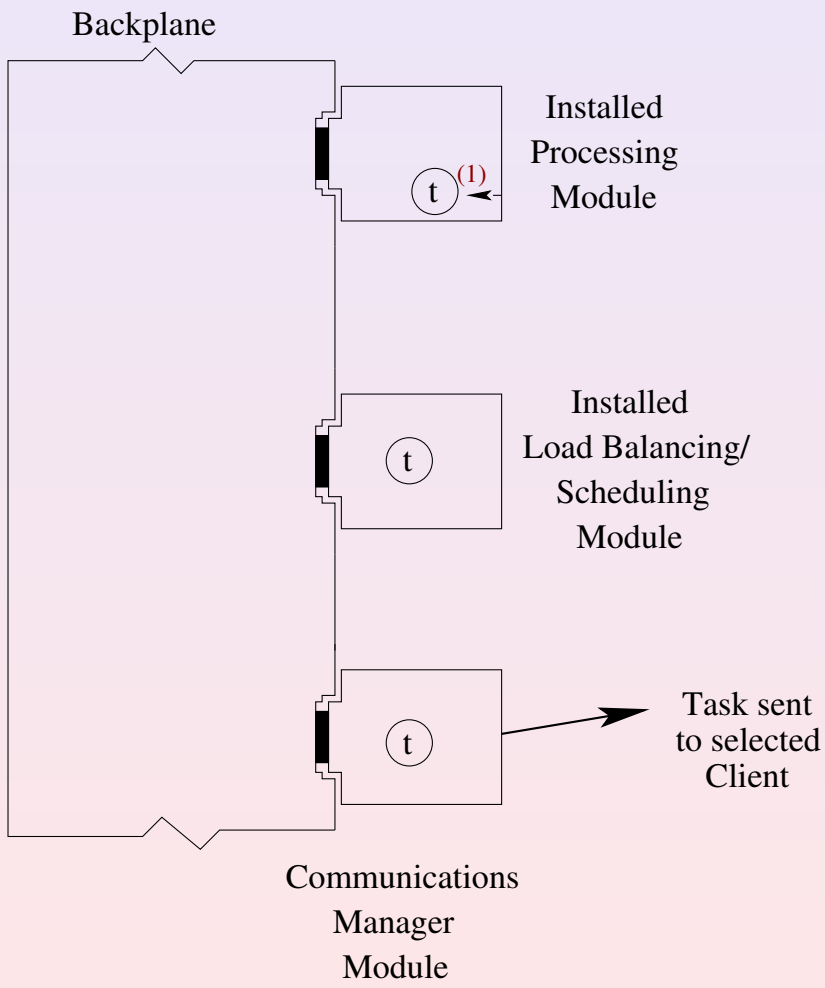
Local execution of task



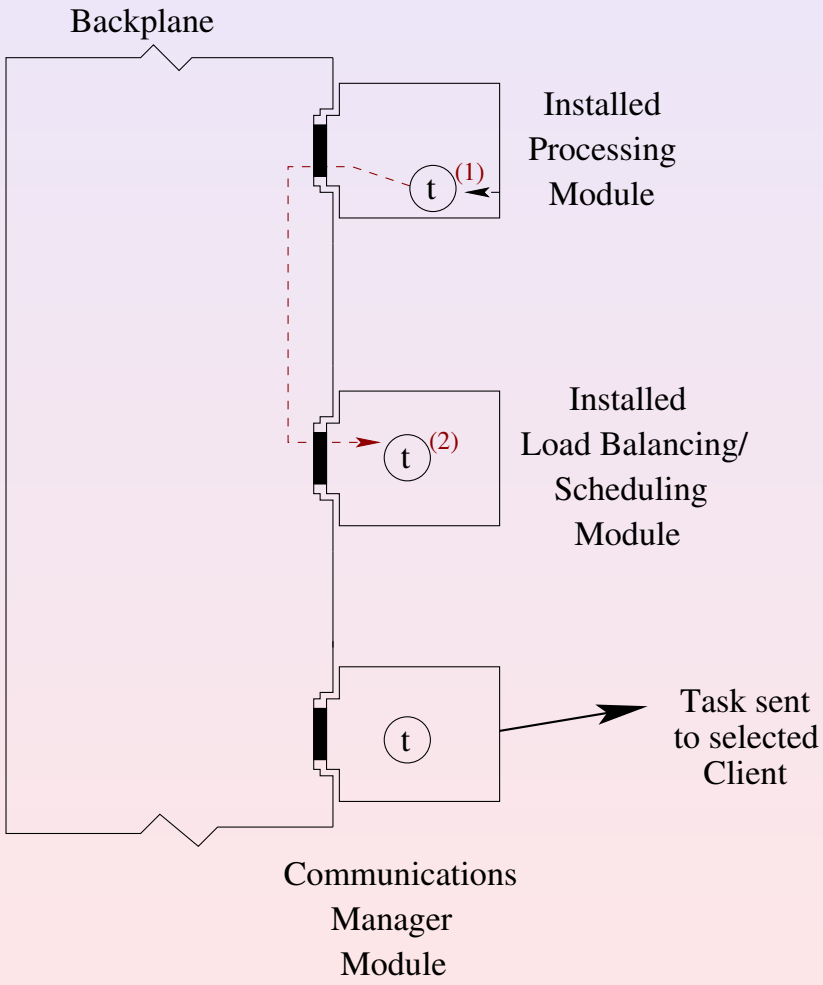
Remote execution of task



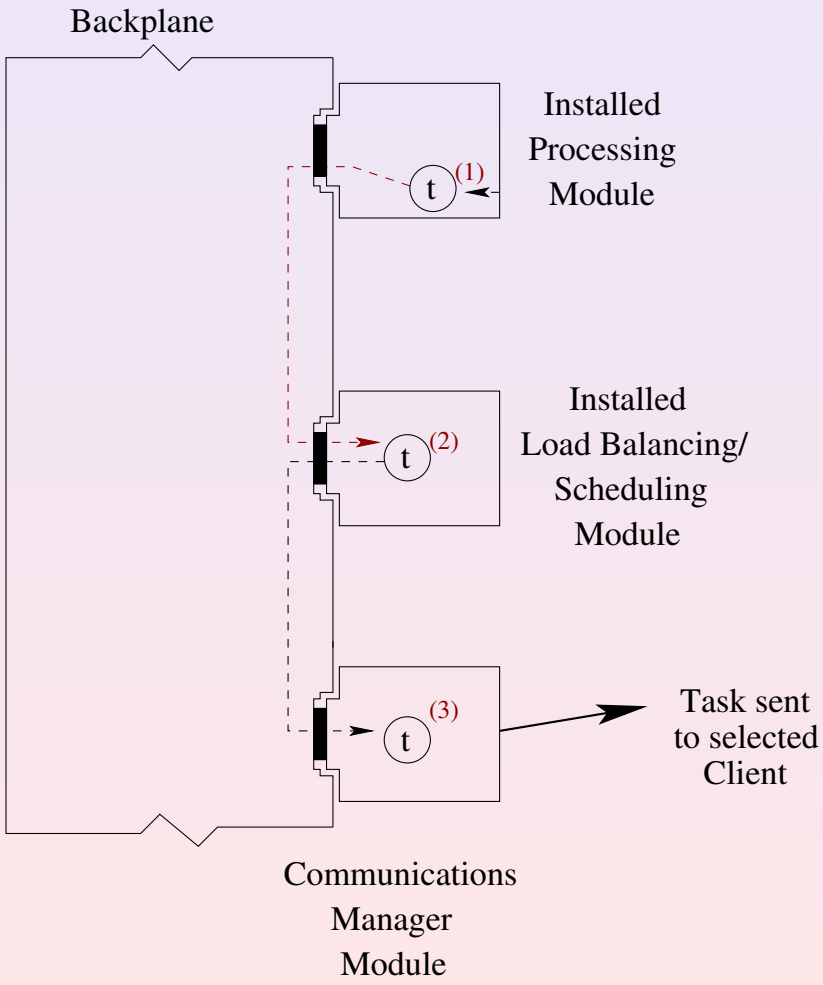
Remote execution of task



Remote execution of task



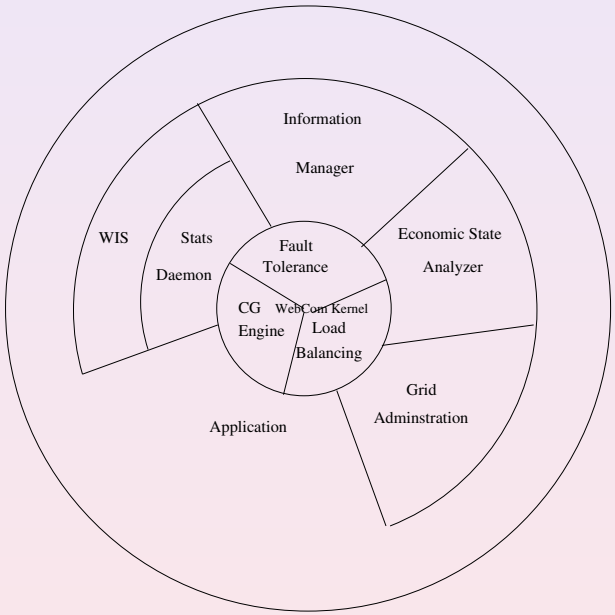
Remote execution of task



Outline

WebCom+Grid Operating System

- seamless interoperatibility with existing core grid middlewares
- operate between the system hardware and installed grid middleware
- select the appropriate middleware based on specific requirements of the application



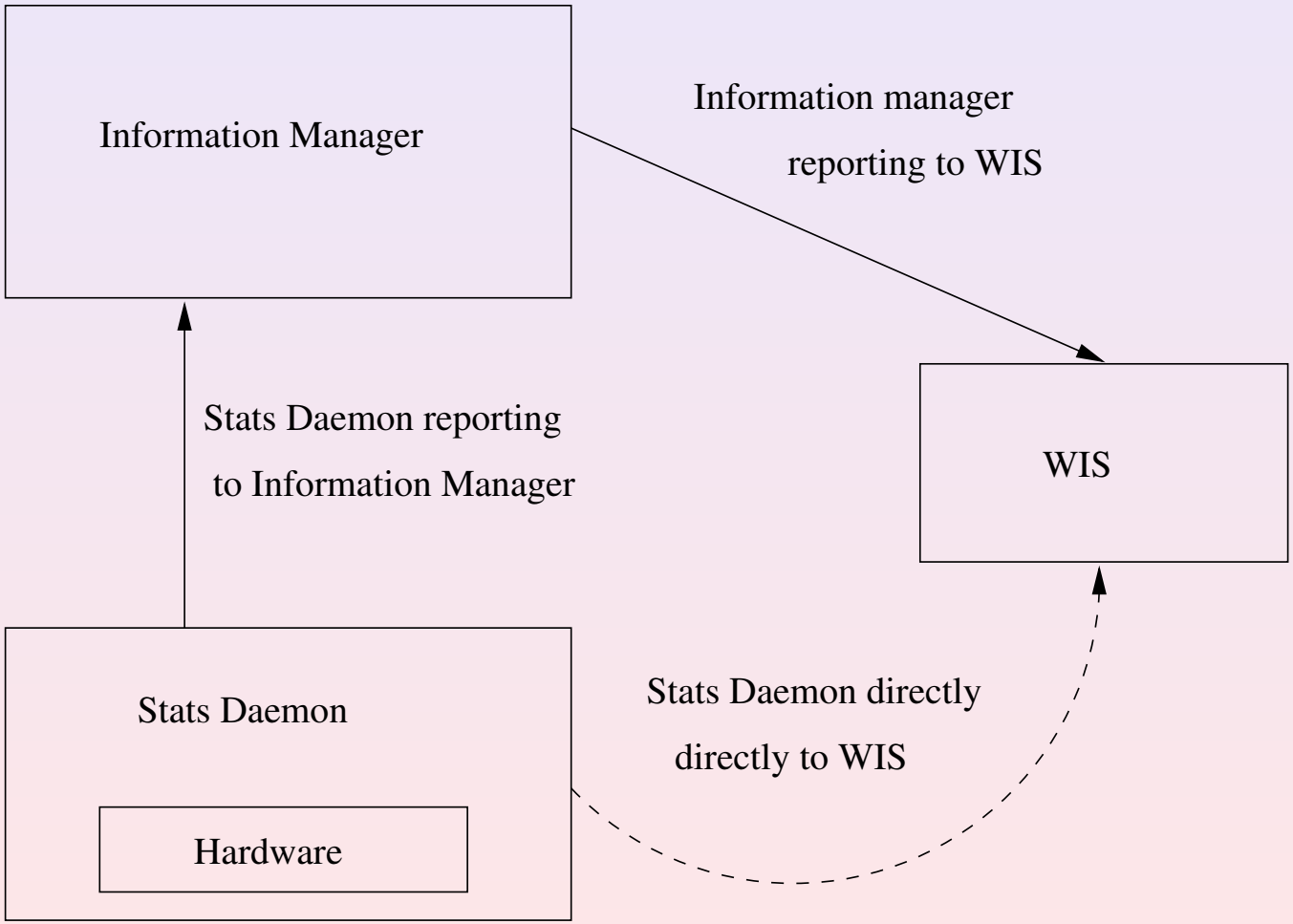
Components of WebCom-G OS

- Economy Status Analyser
  - collects overall resource status of the grid machines
  - evaluates total cost of utilization by using various algorithms
- Stats Daemon
  - uses standard system calls to retrieve system information
  - logs system usage to the hard drive

Components of WebCom-G OS

- WebCom-G Information Module
  - information gathering module within the WebCom-G OS
  - 3 components
    - low-level stats daemon to run directly on hardware
    - higher-level information manager
    - WIS proper communicate with single stats daemon or middlewares
- Grid Administration
  - allow administration by user or middleware
  - dynamically renice the processes

Architecture of WebCom-G Information gathering Module



Outline

Conclusion

- Push and pull connections
- Task targeting and forwarding
- Client promotion and redirection
- Load Balancing and fault tolerance
- Security throught trust management system
- Glue between system hardware and middlewares

Future Work

- DIET + WebCom
- Execute WebCom application on DIET
- Use WebCom fault tolerance for DIET