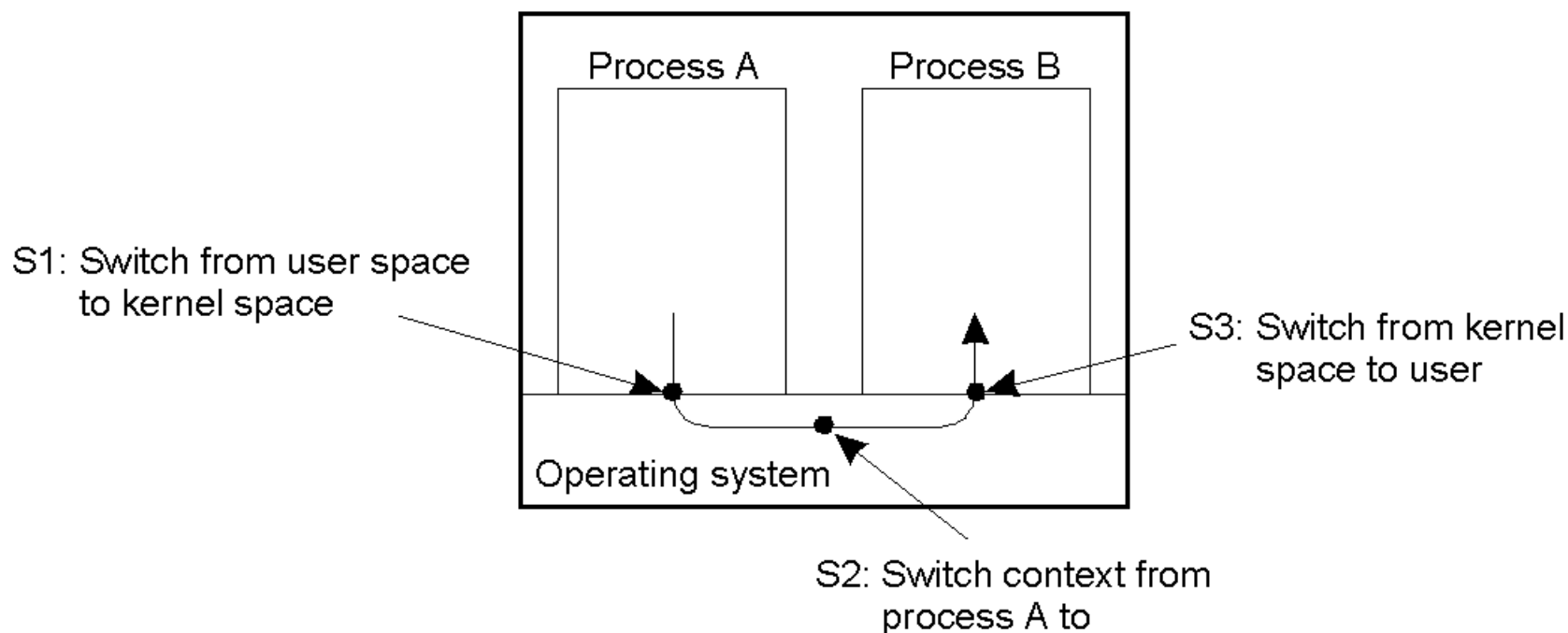


Распределенные системы

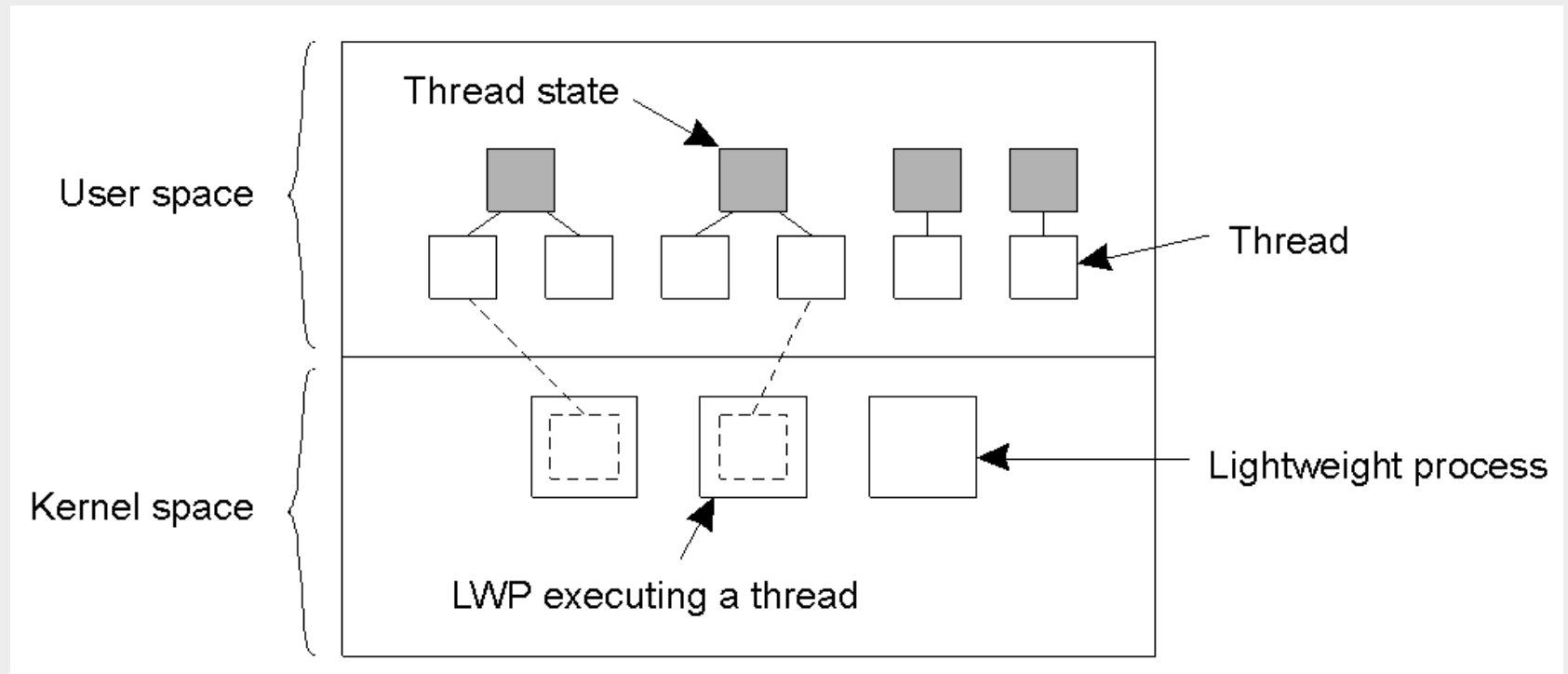
Процессы

Использование потоков (threads)

- Переключение контекста выполнения во время IPC

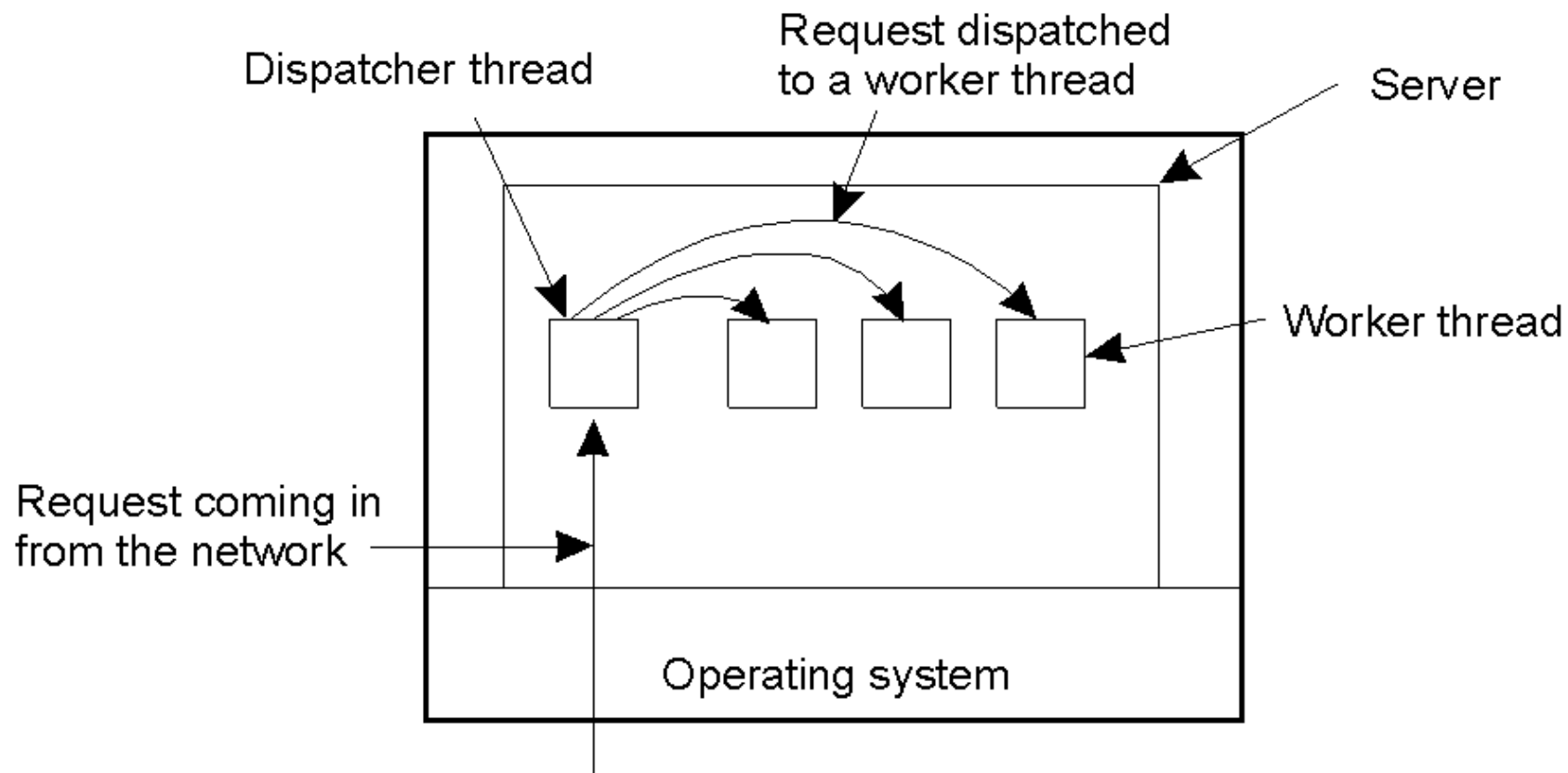


Реализация потоков с использованием LWP



Многопоточные сервера

- Модель организации dispatcher/worker



Многопоточные сервера

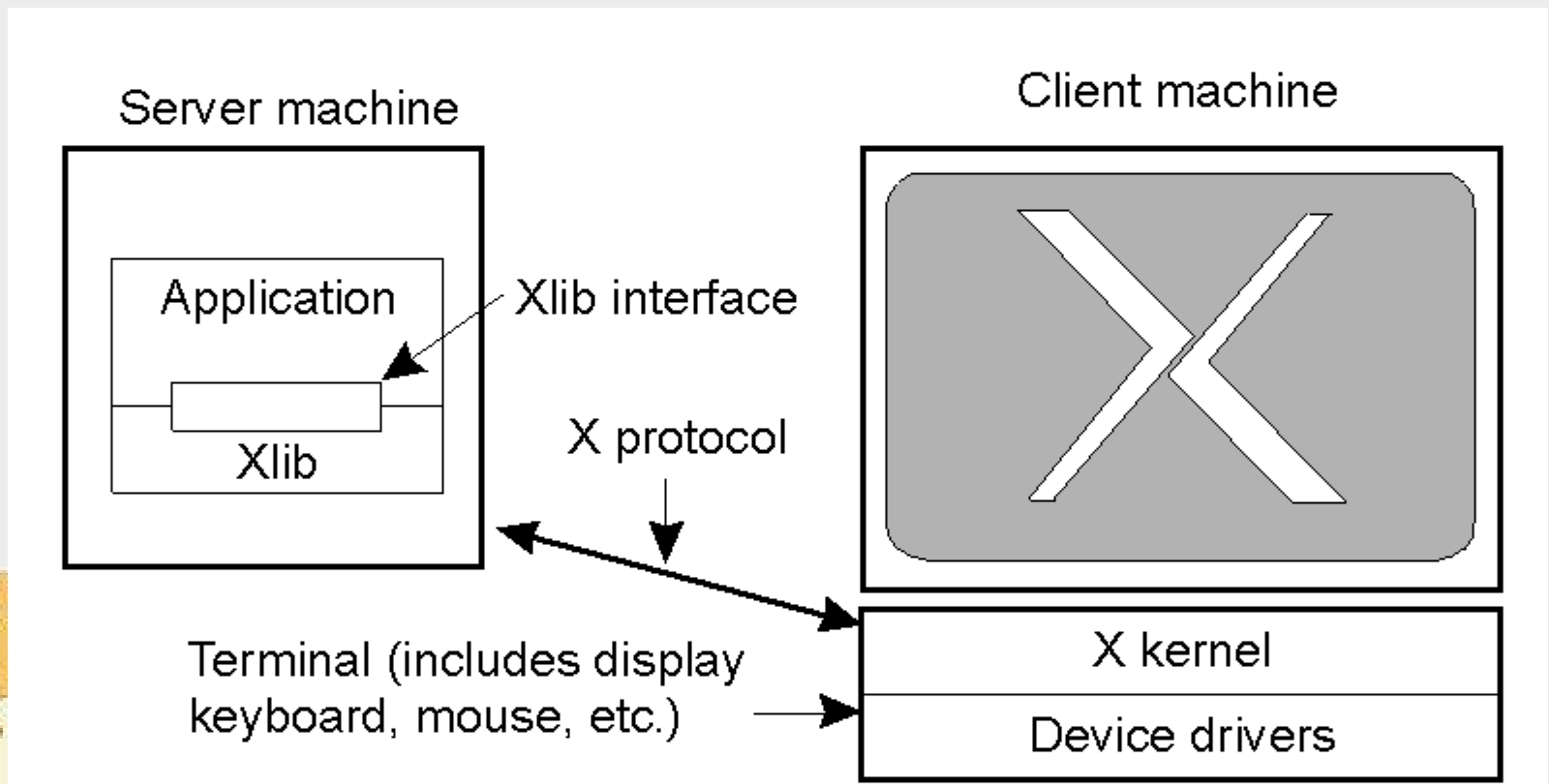
- 3 пути реализации сервера

Модель	Характеристики
Threads (Потоки)	Параллельное выполнение, блокирующие системные вызовы
Single-threaded process (Однопоточный процесс)	Отсутствует параллельное выполнение, блокирующие системные вызовы
Finite-state machine (Конечный автомат)	Параллельное выполнение, неблокирующие системные вызовы



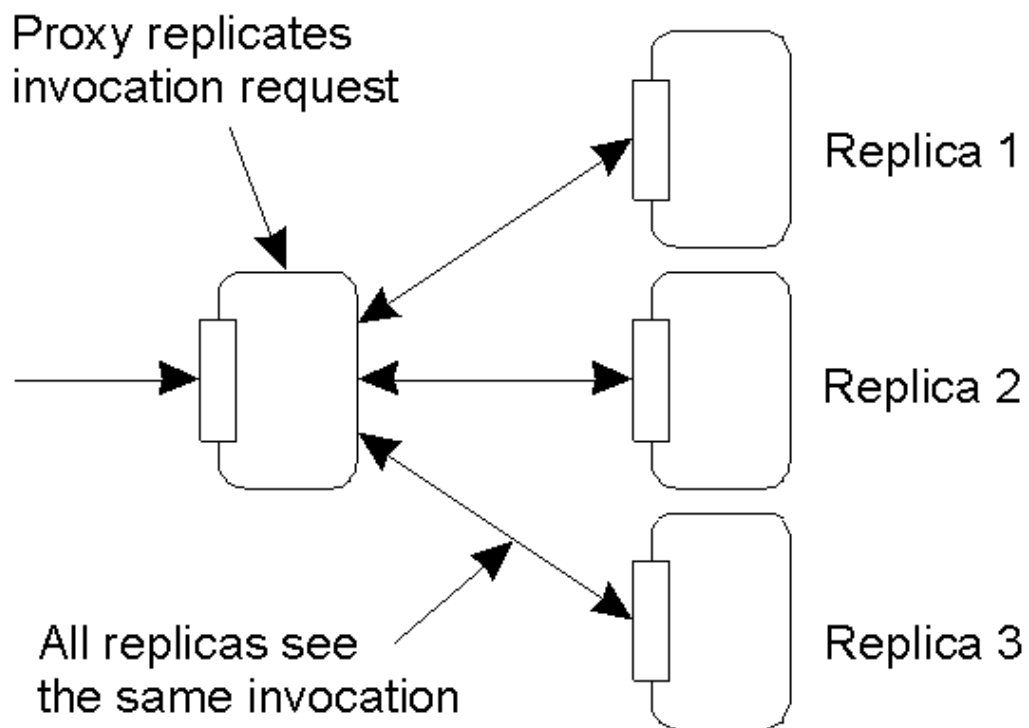
X-Window System

- Основы организации X Window System



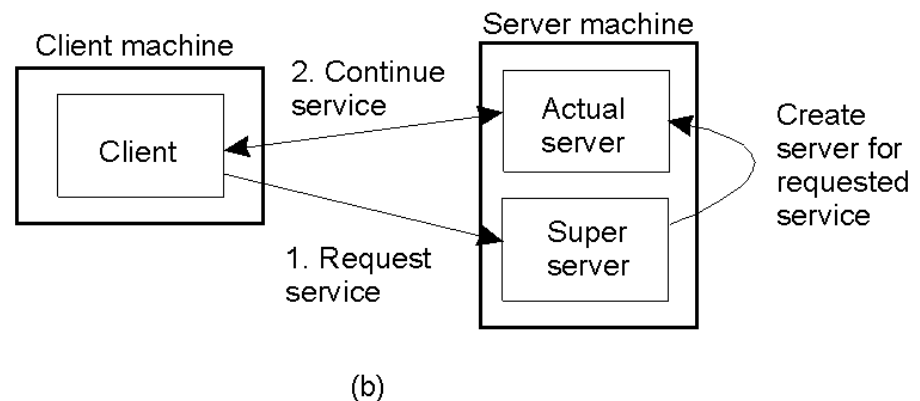
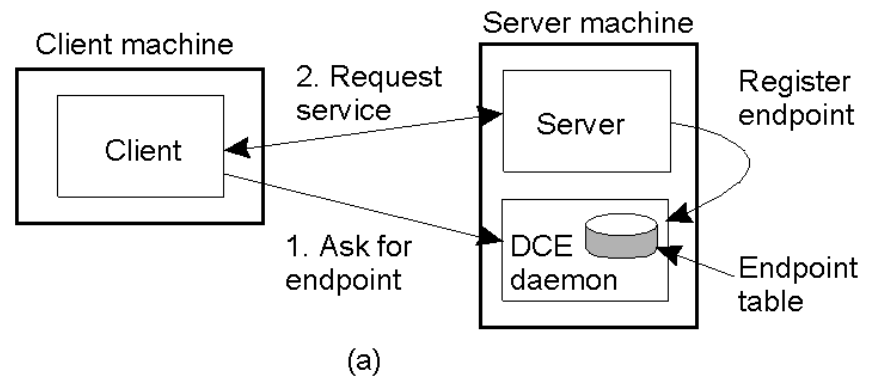
Прозрачность распределения на основе клиентского ПО

- Возможный подход к обеспечению прозрачности за счет клиентского ПО



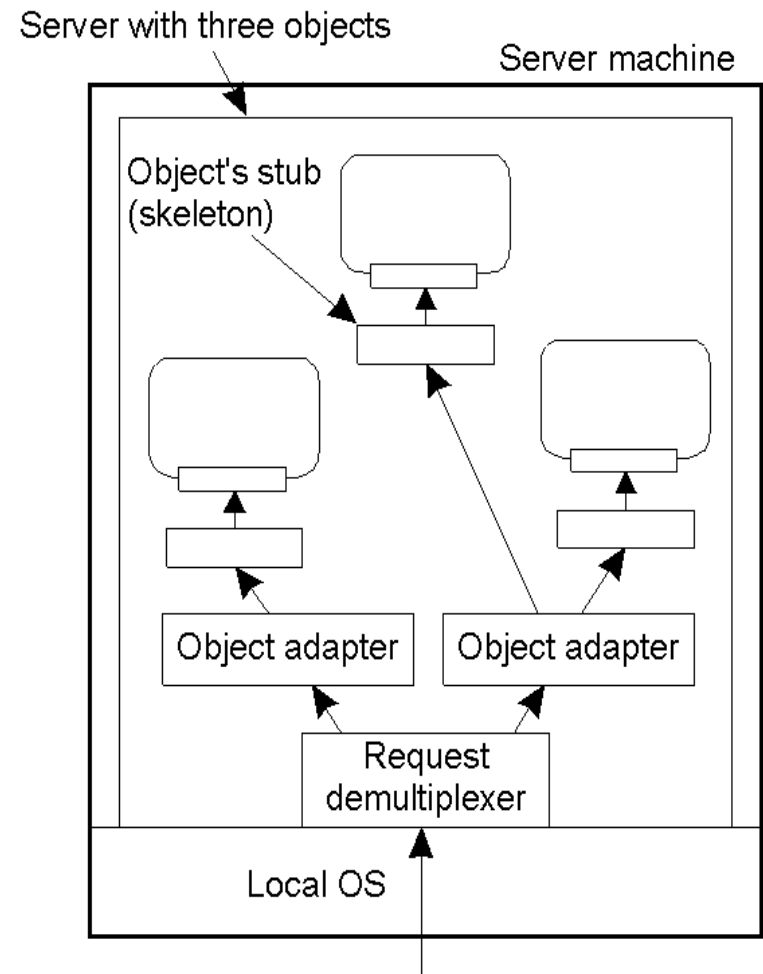
Сервер: Общие проблемы проектирования

- a) Привязка клиента к серверу с использованием DCE
- b) Привязка клиента к серверу с использованием суперсервера



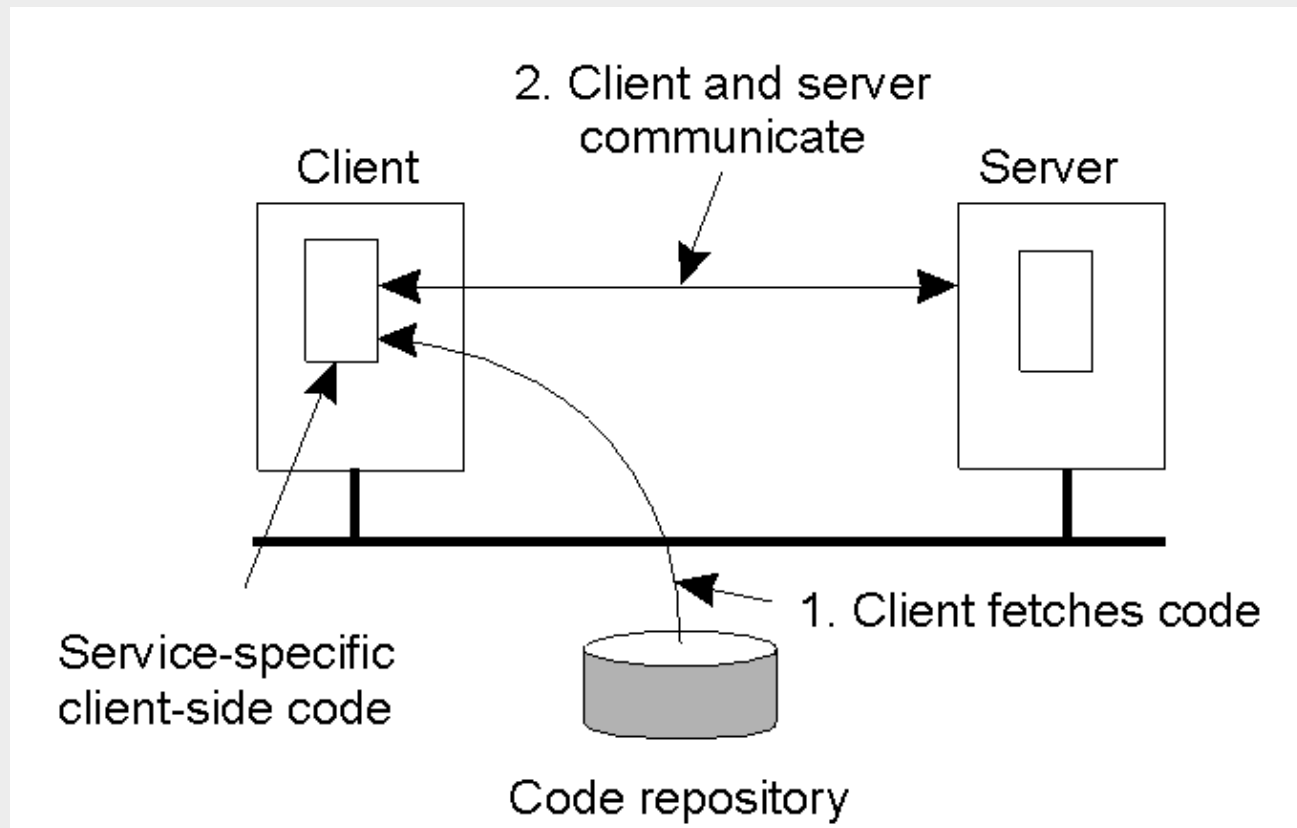
Объектный адаптер

- Организация объектного сервера с поддержкой различных политик активации



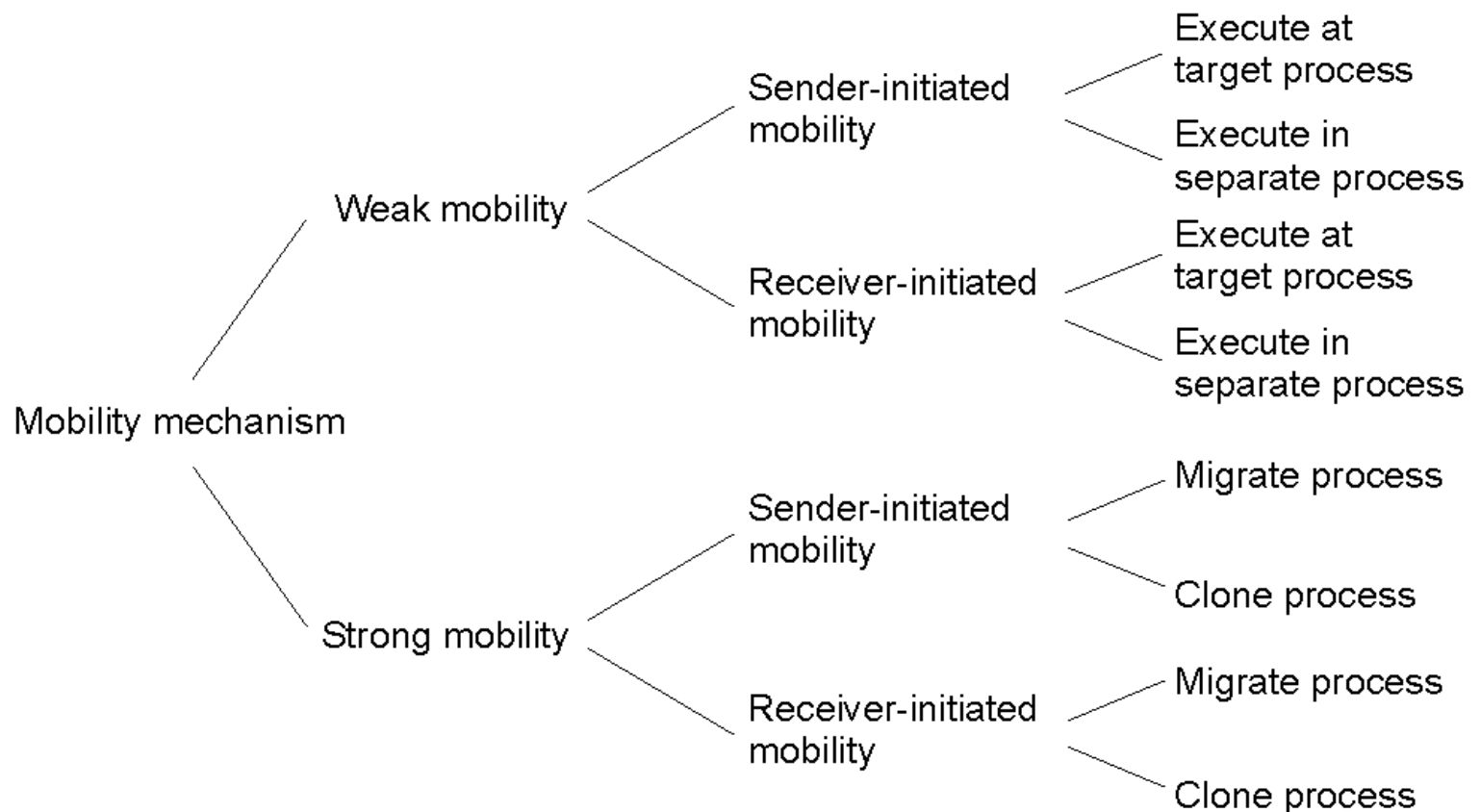
Миграция кода

- Динамическое конфигурирование клиента для взаимодействия с сервером



Модели миграции кода

■ Альтернативные варианты



Миграция и локальные ресурсы

Привязка ресурсов к машине

Привязка ресурсов к процессу

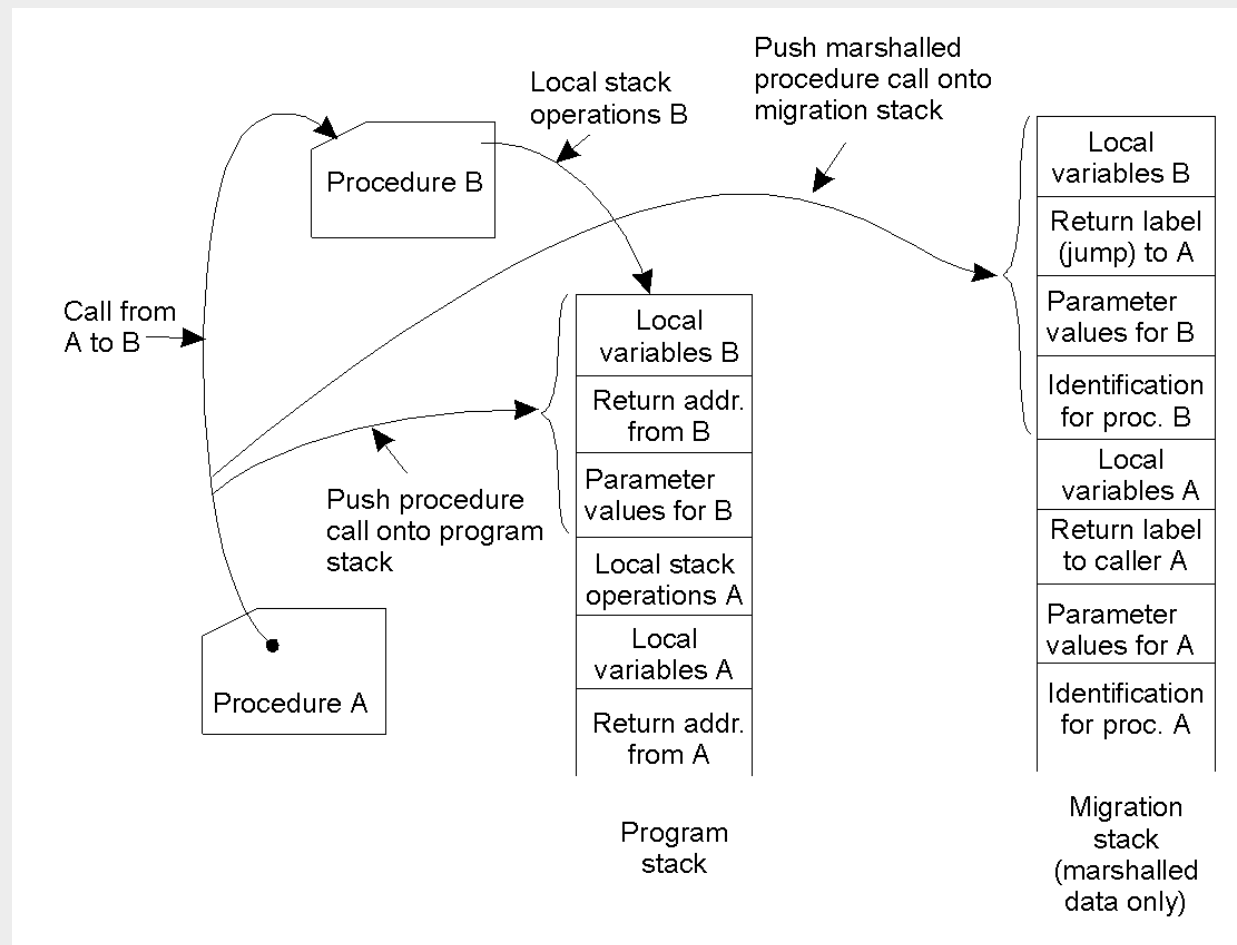
	Неприсоединенный рес	Связанный рес	Фиксированный рес
По идентификатору	MV (or GR)	GR (or MV)	GR
По значению	CP (or MV, GR)	GR (or CP)	GR
По типу	RB (or GR, CP)	RB (or GR, CP)	RB (or GR)

- Варианты переноса кода на другую машину
 - GR – глобальная ссылка
 - MV – перенос ресурса
 - CP – копирование ресурса
 - RB - повторная привязка ресурса



Миграция в гетерогенных системах

- Принцип миграции стека в гетерогенных системах



Миграция кода в D'Agents

```
proc factorial n {  
  if ($n ≤ 1) { return 1; }           # fac(1) = 1  
  expr $n * [ factorial [expr $n - 1] ] # fac(n) = n * fac(n - 1)  
}
```

```
set number ...      # tells which factorial to compute
```

```
set machine ...     # identify the target machine
```

```
agent_submit $machine -procs factorial -vars number -script {factorial $number }
```

```
agent_receive ...   # receive the results (left unspecified for simplicity)
```

■ Пример слабой мобильности с инициированием со стороны отправителя



Миграция кода в D'Agents

■ Пример сильной мобильности

```
all_users $machines

proc all_users machines {
  set list ""
  foreach m $machines {
    agent_jump $m
    set users [exec who]
    append list $users
  }
  return $list
}

set machines ...
set this_machine ...

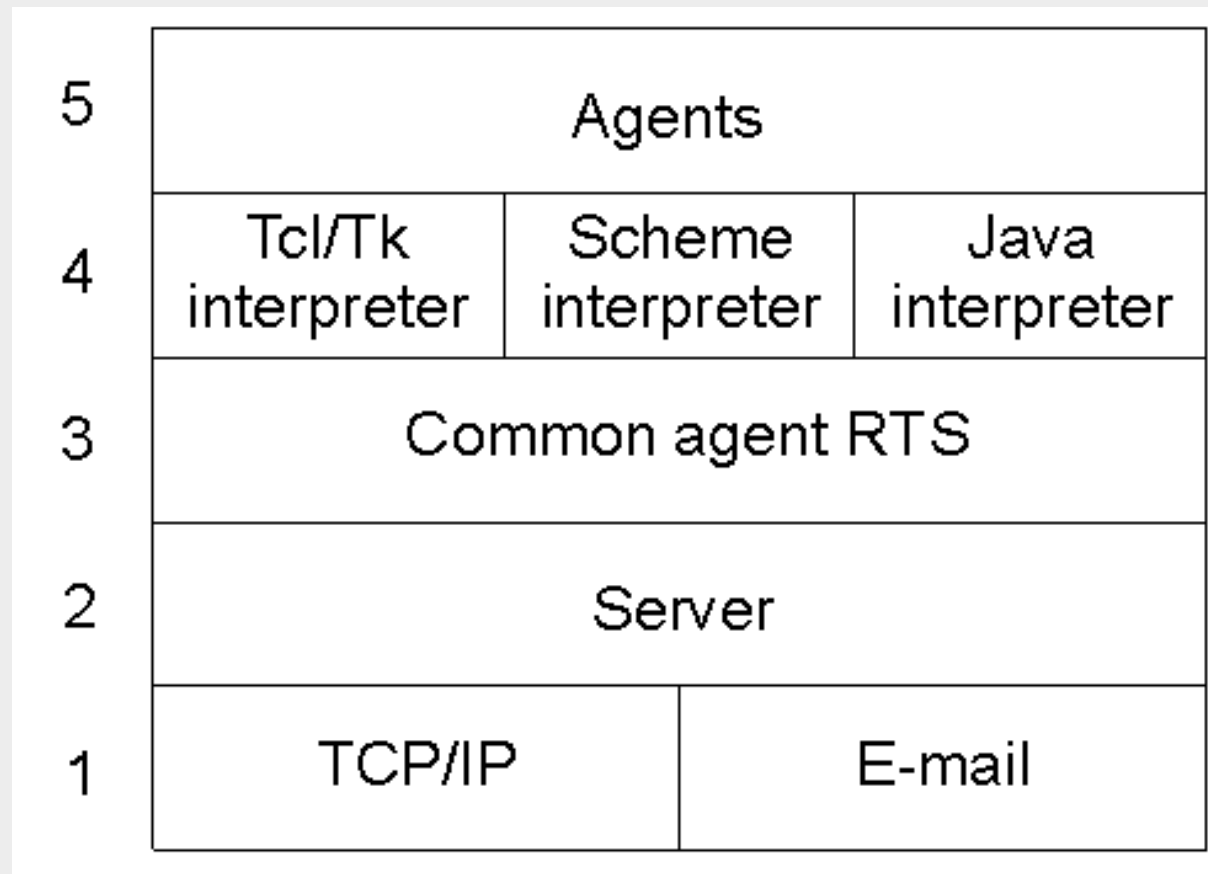
# Create a migrating agent by submitting the script to this machine, from where
# it will jump to all the others in $machines.
agent_submit $this_machine -procs all_users
                             -vars machines
                             -script { all_users $machines }

agent_receive ...
```

Create an initially empty list
Consider all hosts in the set of given machines
Jump to each host
Execute the who command
Append the results to the list
Return the complete list when done
Initialize the set of machines to jump to
Set to the host that starts the agent
#receive the results (left unspecified for simplicity)

Проблемы реализации

- Архитектура D'Agents



Проблемы реализации

- Описание агента представлено следующими частями

Состояние	Описание
Глобальные переменные интерпретатора	Переменные, необходимые интерпретатору
Глобальные системные переменные	Коды возвратов, коды ошибок, строки сообщений об ошибках
Глобальные переменные программы	Определяемые пользователем глобальные переменные программы
Определения процедур	Определение сценариев, выполняемых агентом
Стек команд	Стек, выполняемых в настоящий момент команд
Стек вызовов	Стек записей об активации

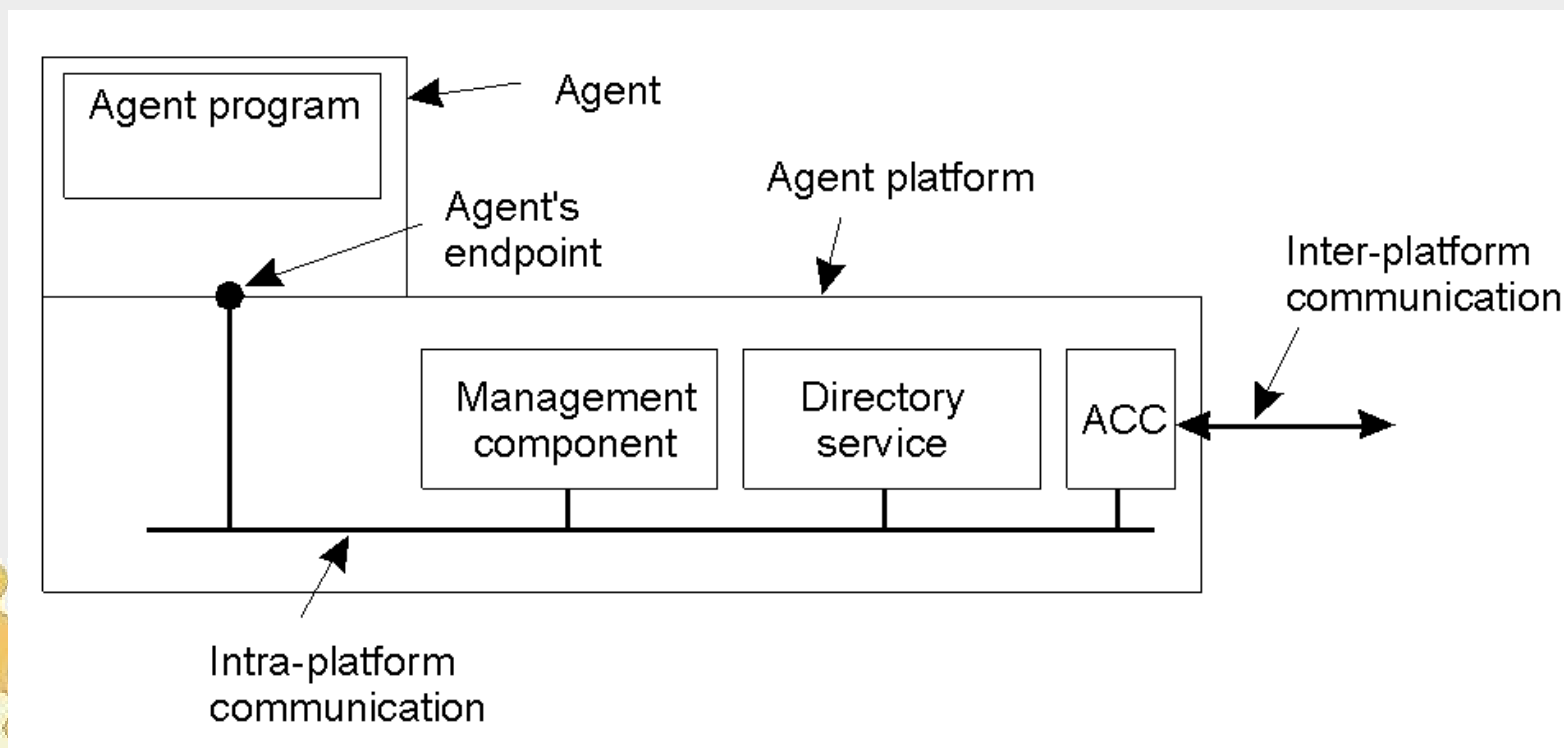
Программные агенты в распределенных системах

■ Некоторые важные свойства агентов

Свойство	Общность для агентов	Описание
Автономность	Да	Способность работать автономно от других
Реактивность	Да	Способность своевременно реагировать на изменения в окружении
Проактивность	Да	Способность инициировать действие, влияющее на окружение
Коммуникативность	Да	Способность обмениваться информацией с пользователями и другими агентами
Продолжительность	Нет	Имеет относительно долгое время жизни
Мобильность	Нет	Способность перемещаться с места на место
Адаптируемость	Нет	Способность к обучению

Технология агентов

- Общая модель платформы агентов



Языки взаимодействия агентов

- Примеры различных типов сообщений (ЦЕЛЬ, СОДЕРЖАНИЕ)

Цель	Описание	Содержимое сообщения
INFORM	Информировать, что данное предположение истинно	Предположение
QUERY-IF	Запросить, истинно ли данное предположение	Предположение
QUERY-REF	Запрос данного объекта	Выражение

Языки взаимодействия агентов

Поле	Значение
Цель	INFORM
Отправитель	max@http://fanclub-beatrix.royalty-spotters.nl:7239
Получатель	elke@iiop://royalty-watcher.uk:5623
Язык	Prolog
Онтология	genealogy
Содержимое	female(beatrix),parent(beatrix,juliana,bernhard)



- Пример сообщения между агентами, использующими Пролог для описания генеалогии