
СУПЕРКОМПЬЮТЕРНЫЙ КОНСОРЦИУМ УНИВЕРСИТЕТОВ РОССИИ

Проект

*Создание системы подготовки
высококвалифицированных кадров
в области суперкомпьютерных
технологий и специализированного
программного обеспечения*



Московский государственный университет
им. М.В. Ломоносова

Методы и средства построения распределенных программных систем с использованием технологии Java

Лекция 1.
Введение

Структура курса. Лекции

- 1_CourseIntroduction.ppt Введение: определения, мотивация, проблемы
- 2_Architecture&Layers.ppt Типичные архитектуры распределенных приложений
- 3_IPC&Network.ppt Механизмы IPC. Некоторые сведения о сетевых технологиях
- 4_UDP&TCP Sockets.ppt API для протоколов TCP и UDP (пакет java.net)
- 5_Socket Examples.ppt Примеры использования пакета java.net
- 6_Parallel.ppt Проблемы, возникающие при параллельном доступе к данным и подходы к их решению
- 7_Middleware.ppt Роль промежуточного программного обеспечения при разработке распределенных приложений
- 8_RMI.ppt Введение в Java RMI
- 9_RMI Examples.ppt Примеры использования Java RMI
- 10_CORBA.ppt Введение в CORBA
- 11_CORBA Basic Examples.ppt Примеры использования CORBA (базовые)
- 12_CORBA DII&DSI Examples.ppt Примеры использования CORBA (DII&DSI)
- 13_Web Services.ppt Введение в технологию Веб-сервисов
- 14_Web Services Examples.ppt Примеры использования Веб-сервисов
- 15_JMS.ppt Введение в JMS
- 16_JMS Examples.ppt Примеры использования JMS

Структура курса. Практикум

- 6 Самостоятельных работ
- 1 «Большая» задача

Задачи курса

По окончании слушатели должны:

- **знать:**
 - Основные термины и технологии распределенных систем
 - Основные проблемы, возникающие при построении распределенных систем и пути их решения
- **уметь:**
 - Использовать на практике следующие технологии:
 - Сокеты (API UDP и TCP)
 - Java RMI
 - CORBA
 - Web-сервисы
 - JMS

Лекция 1

- Что такое распределенная система?
- Примеры распределенных систем.
- Зачем нужны распределенные системы?
- Какими характеристиками должны обладать распределенные системы?
- Какие проблемы существуют при проектировании и реализации распределенных систем?

Определения

- «Распределенная система это набор независимых узлов (компьютеров), которые представляются пользователю как единая система.»
- «Распределенная система это собрание независимых компьютеров соединенных сетью с программным обеспечением, обеспечивающим их совместное функционирование.»
- «Система, состоящая из набора двух или более независимых узлов, которые координируют свою работу посредством синхронного или асинхронного обмена сообщениями.»
- «Система, чьи компоненты размещены на различных узлах, взаимодействующие и управляемые только посредством передачи сообщений.»
- «...система нескольких автономных вычислительных узлов, взаимодействующих для выполнения общей цели.»
- «...я не могу объяснить, что такое распределенная система, но узнаю ее как только мне ее покажут»

Последствия «распределенности»

- Параллельность
 - Необходимость в разделяемых ресурсах
 - коллизии при доступе
 - Параллельное выполнение
 - «ГОНКИ ПОТОКОВ»
 - Необходимость в синхронизации

Последствия «распределенности»

- Нет «глобального» времени
 - Асинхронная передача сообщений
 - Ограниченная точность синхронизации часов
- Нет «глобального» состояния системы
 - Нет ни одного процесса в распределенной системе, который бы знал текущее глобальное состояние системы

Последствия «распределенности»

- Процессы выполняются автономно, изолированно
- Сбои отдельных процессов могут остаться необнаруженными
- Отдельные процессы могут не подозревать об общесистемном сбое
- Сбои происходят чаще, чем в централизованной системе
- Новые причины сбоев (которых не было в монолитных системах)
- Сетевые сбои изолируют процессы и фрагментируют систему на изолированные части

Принципы разделения

- **Функциональное разделение: узлы выполняют различные задачи**
 - Клиент / сервер
 - Хост / Терминал
 - Сборка данных/ обработка данных
- **Естественное разделение (определяемое задачей)**
 - Система обслуживания сети супермаркетов
 - Сеть для поддержки коллективной работы

Мотивы разделения

- Обмен информацией между системами
- Распределение нагрузки/балансировка: назначение задачи на процессора так, чтобы оптимизировать общую загрузку системы
- Усиление мощности: различные узлы работают над одной задачей
 - Распределенные системы содержащие набор микропроцессоров, по мощности могут приближаться к суперкомпьютеру

Мотивы разделения

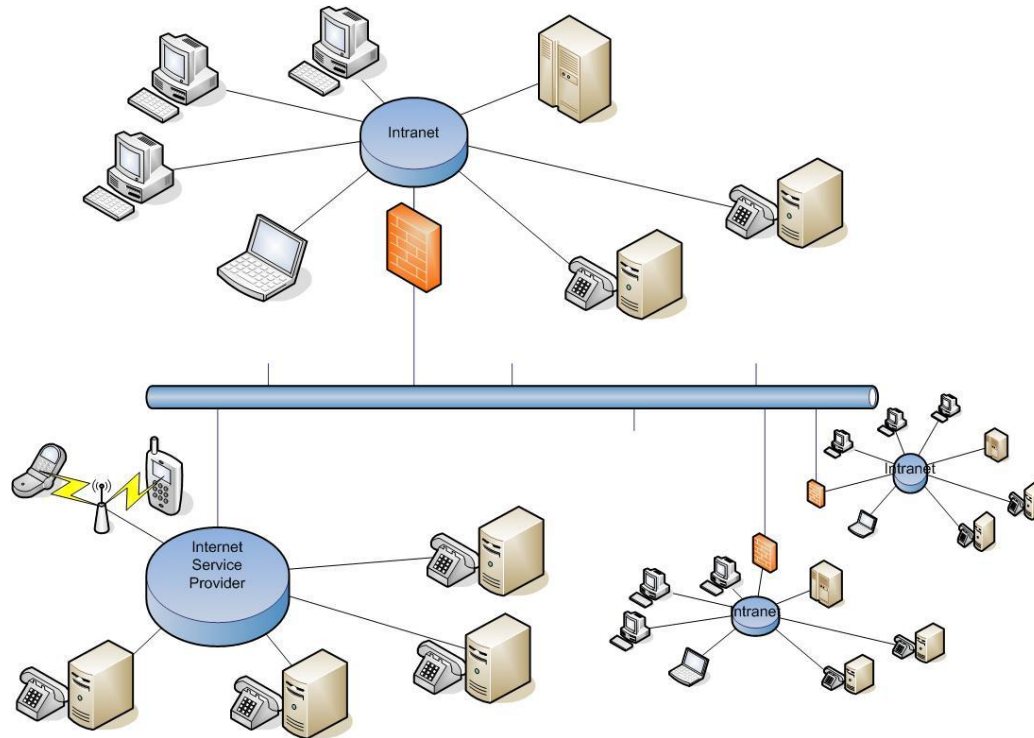
- Физическое разделение: система строится в предположении, что узлы физически разделены (требования к надежности, устойчивости к сбоям)
- Экономические: набор дешевых чипов может обеспечить лучшие показатели отношения цена/производительность, чем мейнфрэйм
- Специализация компонентов: упрощение и удешевление

Примеры распределенных систем

- Internet (?)
- Intranet
- Вычислительные кластера
- ...

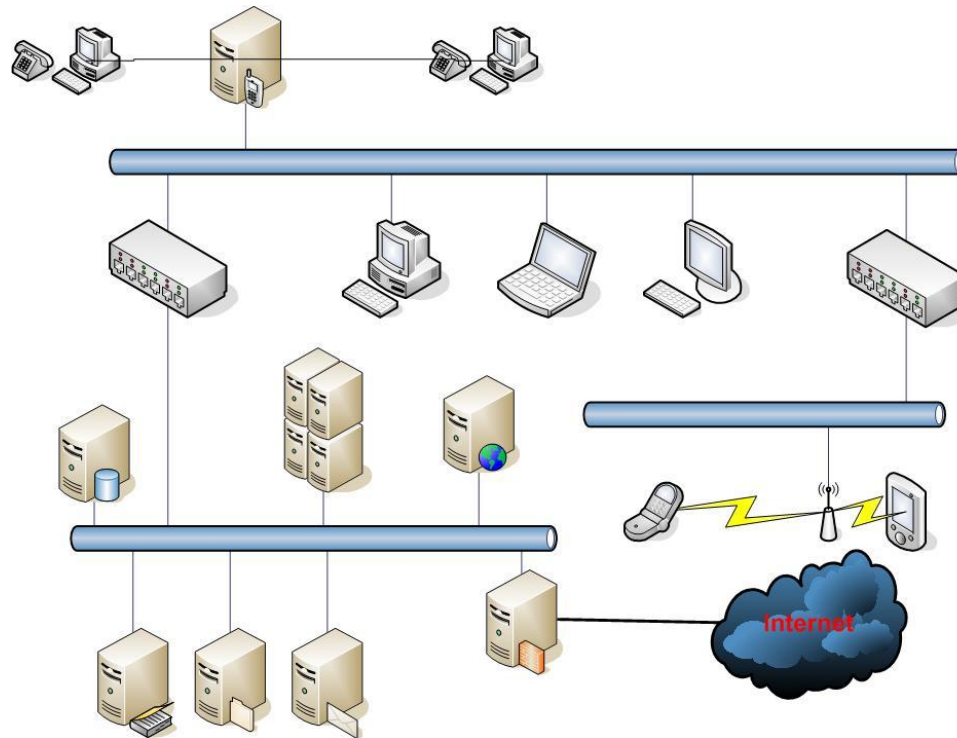
Пример: Internet

- Гетерогенная сеть компьютеров и приложений
- Реализация взаимодействия - IP стек



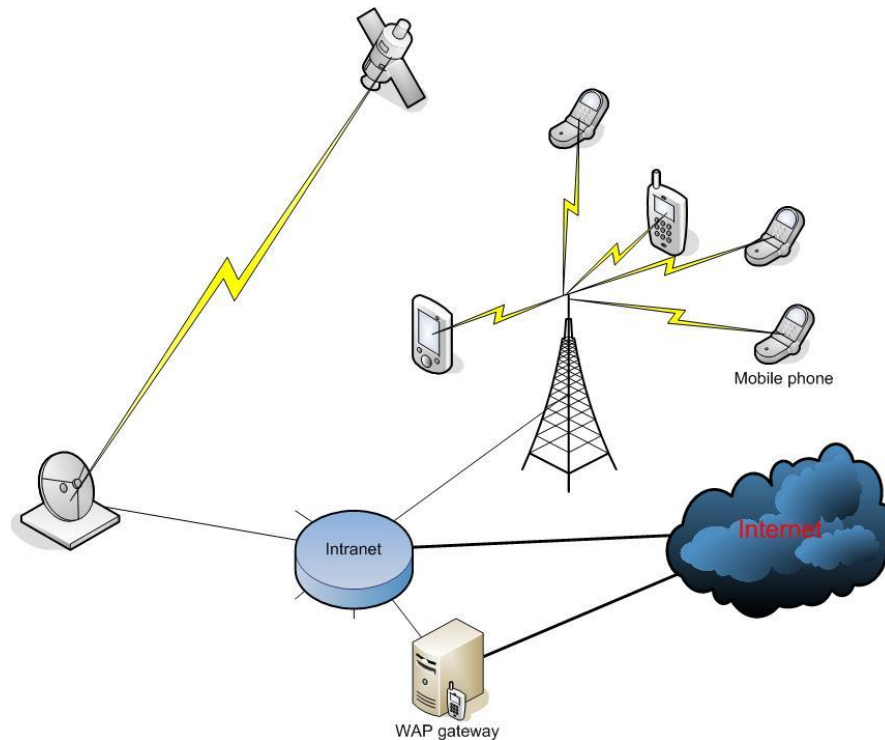
Пример: Intranet

- Администрируется локально
- Взаимодействие с Internet
- Обеспечивает сервисами (внутренних и внешних пользователей)



Пример: Wireless Information Devices

- Система сотовой связи (GSM)
 - Ресурсы разделяемы (радио частота, время передачи на частоте,...)



Другие примеры

- Системы управления аэропортом
- Бортовые управляющие системы самолета



Требования

- Открытость
- Безопасность
- Масштабируемость
- Механизмы обработки ошибок и восстановления после сбоев
- Методы решения проблем параллелизма
- Прозрачность
- Управляемость

Открытость

- Гарантирует расширяемость
- Возможность повторного использования
- Важные факторы:
 - Наличие четких спецификаций
 - Наличие полной документации
 - Опубликованные интерфейсы
 - Тестирование и проверка на многих платформах
- **Использование открытых протоколов и стандартов**

Безопасность

- Физическая распределенность = возможность доступа злоумышленников к компонентам
- Как правило – большое количество пользователей
- Часто – большая «публичность»
- Три компонента:
 - Защищенность
 - Целостность
 - Доступность
- Задача: посылка значимой информации по сети безопасно и эффективно
 - Механизмы авторизации
 - Криптография
 - ...

Безопасность. Нерешенные проблемы

- Атаки типа DoS (отказы в обслуживании)
- Безопасность мобильного кода
 - Непредсказуемые эффекты
 - Может вести себя подобно троянскому коню...

Масштабируемость

- Распределенная система масштабируема, если она остается эффективной при увеличении числа обслуживаемых пользователей или ресурсов
- Проблемы:
 - Контроль стоимости ресурсов
 - Контроль потерь производительности
- Некоторые рекомендации
 - Децентрализация алгоритмов
 - Тиражирование (репликация) и кэширование данных

Обработка ошибок

- Сбои более частые, чем в централизованных системах (введен дополнительный источник сбоев)
- Обработка сбоев включает:
 - Определение факта сбоя (может быть невозможно)
 - Маскирование
 - Восстановление
- Диагностика
 - Может быть возможна (ошибки передачи - контрольная сумма)
 - Может быть невозможна (удаленный сервер не работает или просто очень загружен?)

Параллелизм

- Контроль и планирование параллелизма
- Правильное планирование доступа в параллельных потоках (устранение взаимноисключений, транзакции)
- Синхронизация (семафоры, критические секции)
 - Безопасно, но уменьшают производительность
- Разделяемые объекты (ресурсы) должны работать корректно в многопоточной среде

Прозрачность

- Соккрытие гетерогенной и распределенной структуры системы так, чтобы пользователю система представлялась монолитной

Прозрачность

- **Прозрачность доступа:** доступ к локальным и удаленным ресурсам посредством одинаковых вызовов
- **Прозрачность расположения:** доступ к ресурсам вне зависимости от их физического расположения
- **Прозрачность параллелизма:** возможность нескольким процессам параллельно работать с ресурсами, не оказывая влияния друг на друга
- **Прозрачность репликации:** возможность нескольким экземплярам одного ресурса использоваться без знания физических особенностей репликации
- **Прозрачность обработки ошибок:** защита программных компонентов от сбоев, произошедших в других программных компонентах. Восстановление после сбоев
- **Прозрачность мобильности:** Возможность переноса приложения между платформами, без его переделки
- **Прозрачность производительности:** возможность конфигурации системы с целью увеличения производительности при изменении состава платформы выполнения
- **Прозрачность масштабируемости:** возможность увеличения производительности без изменения структуры программной системы и используемых алгоритмов

Управляемость

- Распределенные ресурсы не имеют центральной точки управления
- Локальная оптимизация системы не всегда означает глобальную оптимизацию
 - Нужен глобальный взгляд на проблему
 - Не всегда возможен (есть системы, никому не принадлежащие)

Сложности при реализации

- Выбор архитектуры
- Гетерогенная среда
- Сложность развертывания
- Сложность отладки

Архитектура

- Сильнейшая зависимость важнейших характеристик от выбранной архитектуры
 - Интерфейс модулей системы определяет количество «нелокальных» вызовов
 - Практическая невозможность отказа от принятых в начале проектирования неверных решений
- Решение: использование широко известных шаблонов архитектур

Гетерогенная среда

- Различные компоненты системы выполняются на различных платформах
- Решения:
 - Использование распространенных открытых стандартов и протоколов
 - Использование промежуточного программного обеспечения (middleware)

Сложность развертывания

- Различные части приложения должны быть развернуты на различных узлах (с различными платформами)
- Решения:
 - Разделение на модули развертывания
 - Использование специализированных инструментальных средств (deploy tools)

Сложность отладки

- Нет глобального состояния
- Параллельность → неповторяемость (невоспроизводимость) результатов выполнения
- Компоненты распределены по разным узлам

Итоги

- Распределенная система:
 - Автономные (но соединенные средой передачи данных) узлы
 - Взаимодействие посредством передачи сообщений
- Много доводов в пользу того, что распределенные системы нужны и их нужно уметь строить
- Распределенные системы существуют и их нужно уметь развивать и поддерживать
- При разработке распределенных систем возникают специфические проблемы

О проекте

Целью проекта является создание национальной системы подготовки высококвалифицированных кадров в области суперкомпьютерных технологий и специализированного программного обеспечения.

Задачами по проекту являются:

Задача 1. Создание сети научно-образовательных центров суперкомпьютерных технологий (НОЦ СКТ).

Задача 2. Разработка учебно-методического обеспечения системы подготовки, переподготовки и повышения квалификации кадров в области суперкомпьютерных технологий.

Задача 3. Реализация образовательных программ подготовки, переподготовки и повышения квалификации кадров в области суперкомпьютерных технологий.

Задача 4. Развитие интеграции фундаментальных и прикладных исследований и образования в области суперкомпьютерных технологий. Обеспечение взаимодействия с РАН, промышленностью, бизнесом.

Задача 5. Расширение международного сотрудничества в создании системы суперкомпьютерного образования.

Задача 6. Разработка и реализации системы информационного обеспечения общества о достижениях в области суперкомпьютерных технологий.

См. <http://www/hpc-education.ru>