

Эволюция систем обработки данных

Типовые функциональные компоненты приложений

- PS (Presentation Services) - средства представления
- PL (Presentation Logic) - логика представления
- BL (Business Logic) - прикладная логика
- DL (Data Logic) - логика управления данными
- DS (Data Services) - операции с базой данных
- FS (File Services) - файловые операции

1. Централизованные системы

Терминал



PS

Мэйнфрейм



PL, BL, DL, DS, FS

1. Централизованные системы

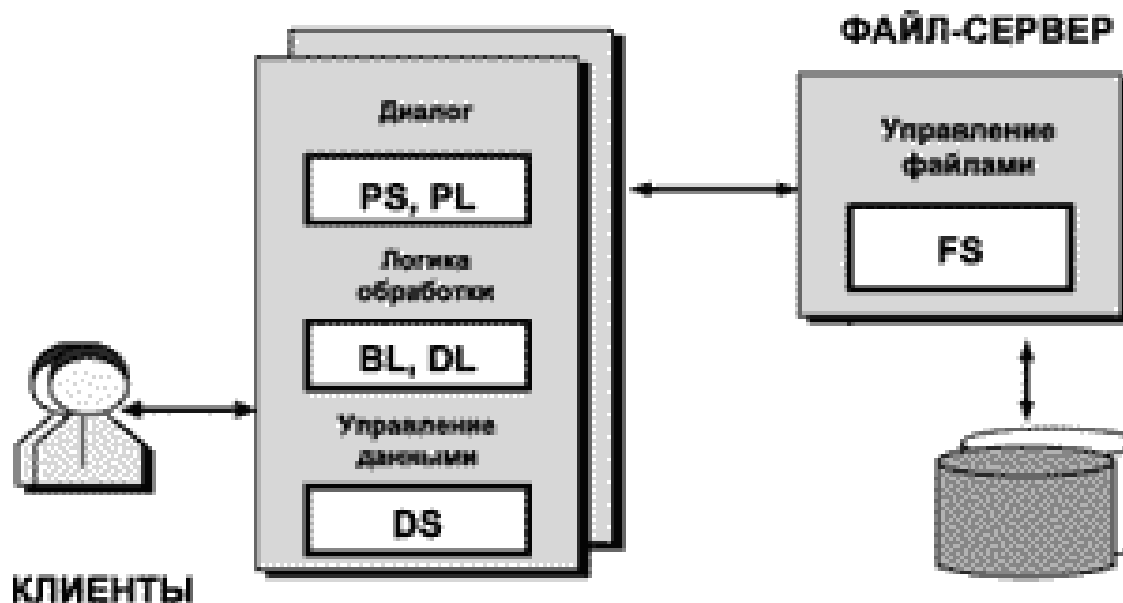
- Достоинства:

- > простота обслуживания и сопровождения

- Недостатки

- > высокая стоимость
- > низкая масштабируемость
- > трудно обеспечить GUI
- > трудно организовать распределенную ОИ

2. Файл-серверные системы

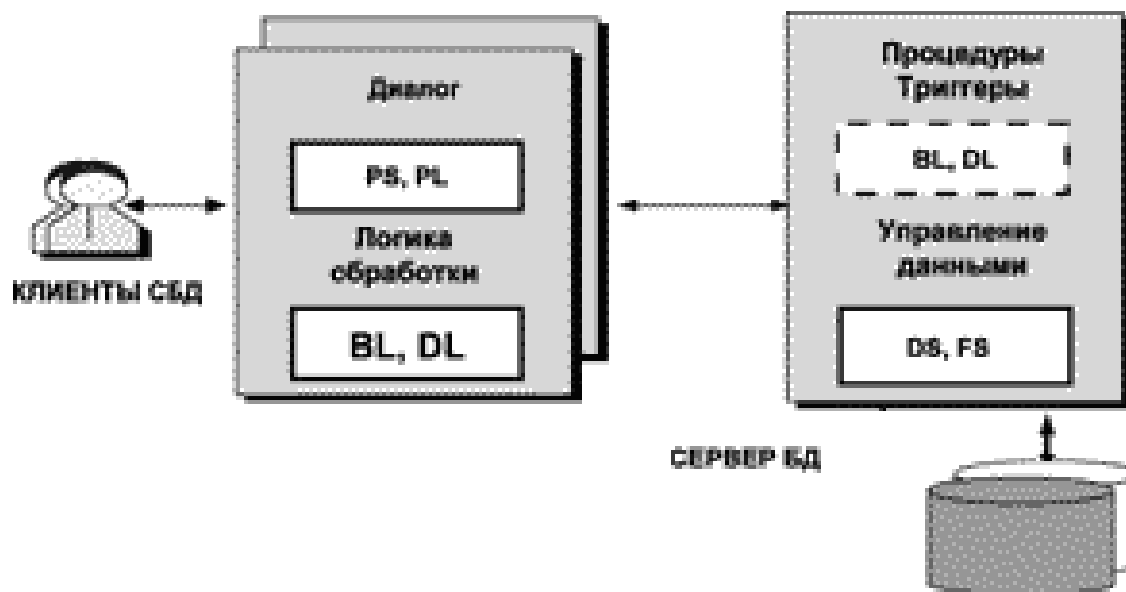


- «Настольные» СУБД
 - > DBase, FoxPro, Clipper, Paradox

2. Файл-серверные системы

- **Достоинства:**
 - > распределенная ОИ
 - > низкая стоимость
- **Недостатки**
 - > большая нагрузка на сеть
 - > Ethernet 10 Мб/с
 - > проблема целостности данных при множественном доступе

3. Клиент-серверные системы

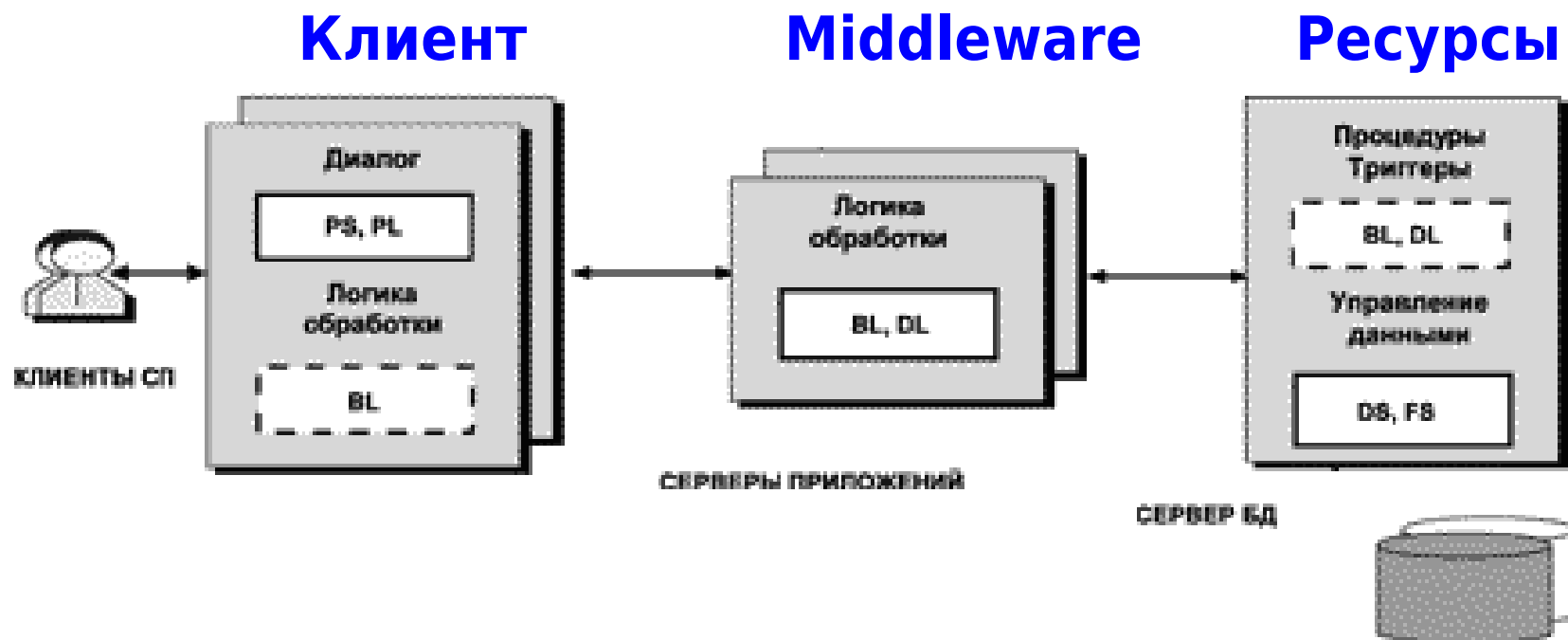


- Доступ к данным как к некоторому набору абстрактных структур (деревья, таблицы)
- Логика обработки данных может выполняться с помощью хранимых процедур и триггеров на сервере БД

3. Клиент-серверные системы

- **Достоинства:**
 - > балансировка нагрузки на вычислительные узлы и сеть
 - > целостность данных при множественном доступе
 - > данные отделены от приложений
- **Недостатки**
 - > сложность сопровождения
 - > обновление клиентов
 - > непереносимость ХП и триггеров
 - > ХП ухудшают масштабируемость
 - > увеличение времени реакции из-за пакетных заданий

4. Многоуровневые системы



- Приложения промежуточного уровня предоставляют клиентам прикладной программный интерфейс доступа к ресурсам и средствам обработки данных

4. Многоуровневые системы

- **Достоинства:**

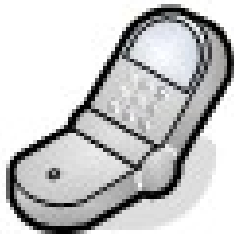
- > независимость клиентов от ресурсов
- > гибкое сопровождение слоев приложения
- > поддержка распределенных транзакций
- > различные представления с единой BL

- **Недостатки**

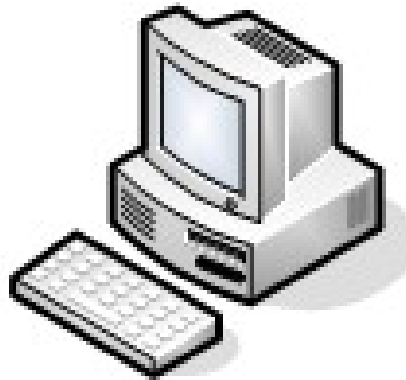
- > относительная дороговизна
- > BL может появиться на всех уровнях

Введение в платформу Java EE

Java Fits All



Java ME



Java SE



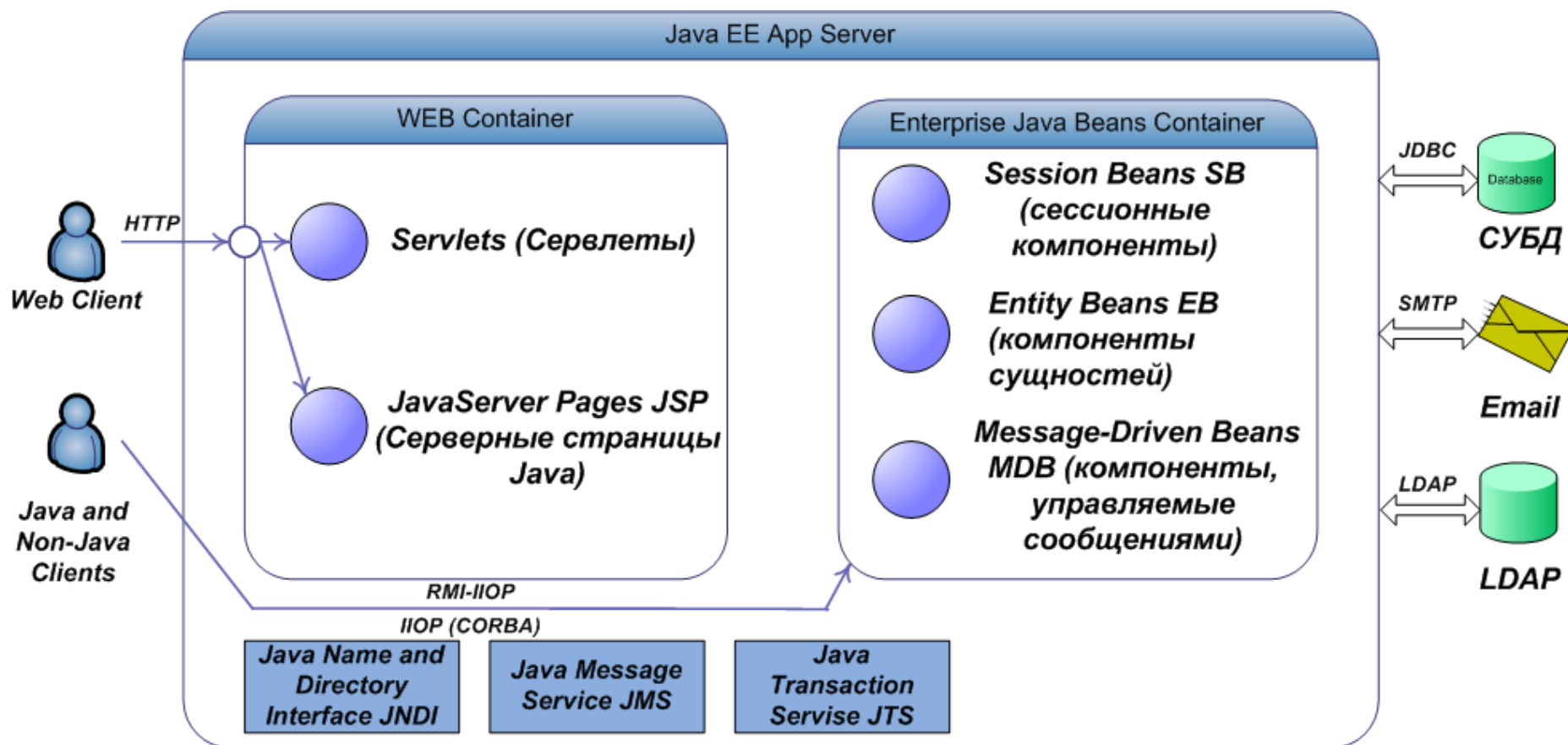
Java EE



Общая характеристика платформы Java EE

- Платформа для разработки и развертывания *middleware*-приложений
 - > Переносимая
 - > Открытая (JCP)
- Определяет **компонентную модель** приложений
- Предоставляет компонентам **набор сервисов**
- Конкретная реализация платформы — **сервер приложений** (application server)
 - > Является **контейнером** для Java EE

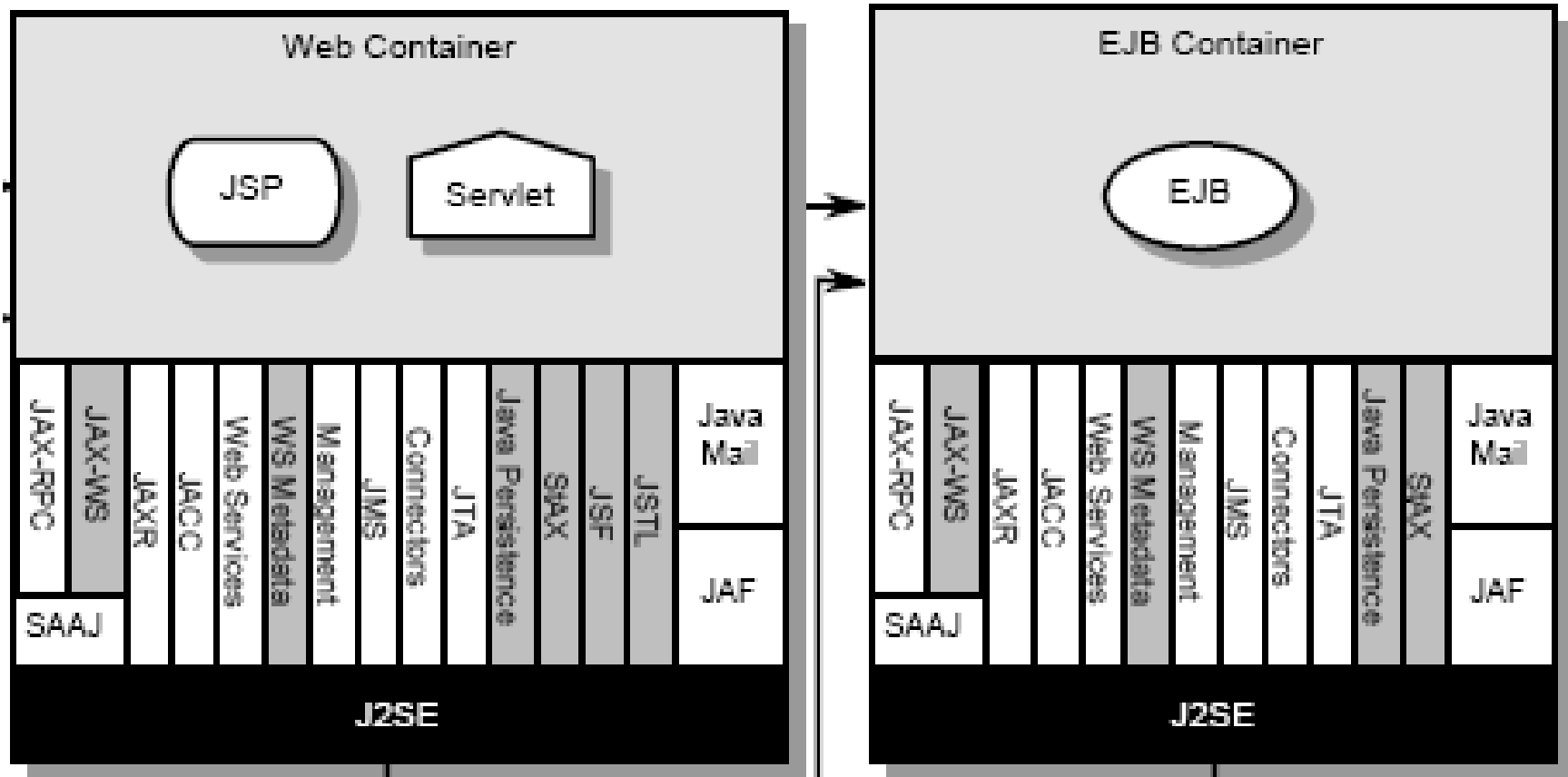
Структура сервера приложений Java EE



Набор сервисов контейнера

- **Управление жизненным циклом компонентов**
- Служба имен и каталогов (JNDI)
- Служба обмена сообщениями (JMS)
- Управление доступом к компоненту
- Координация распределенных транзакций (JTS)
- и др.

Набор сервисов контейнера



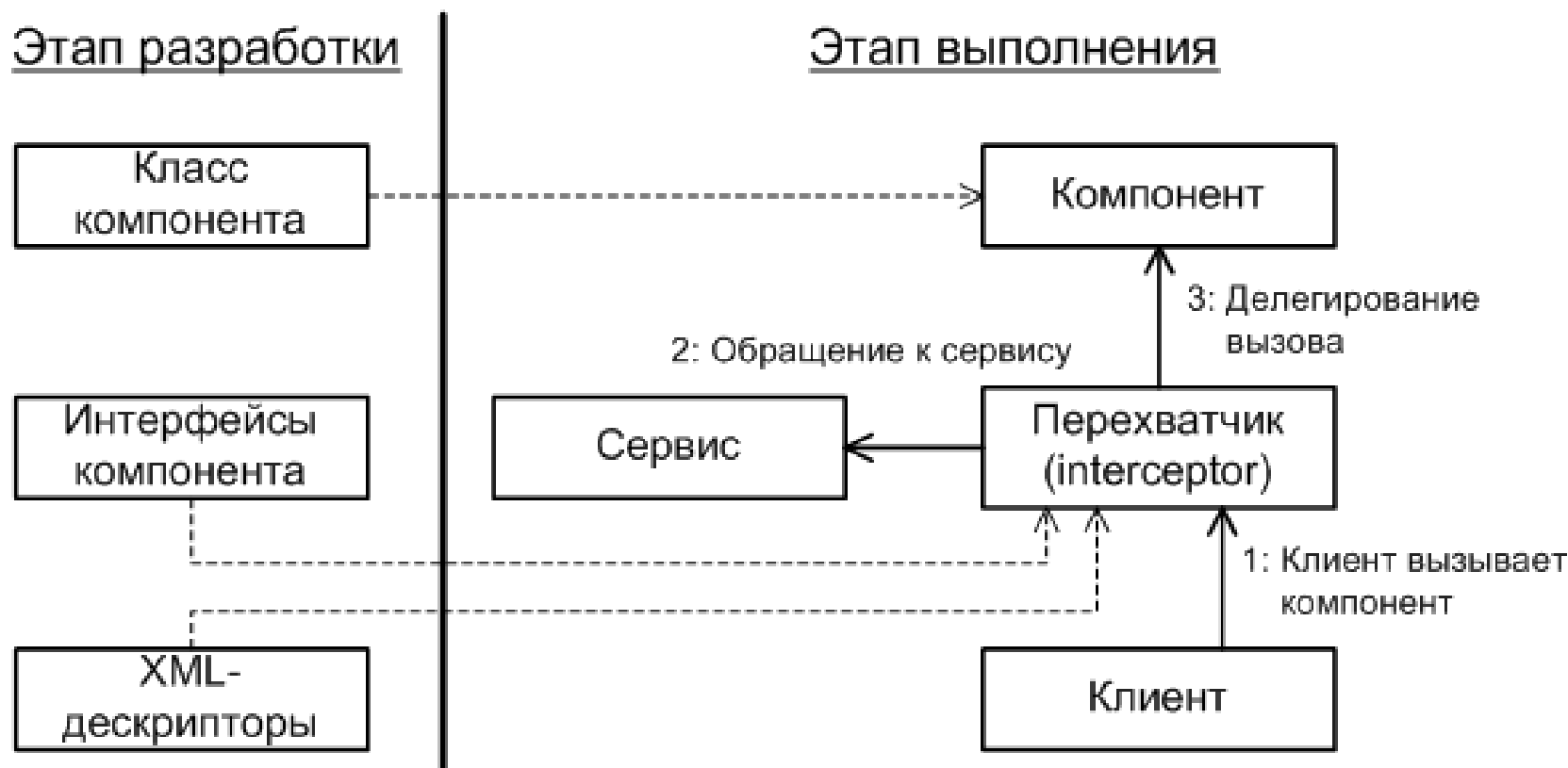


Виды компонентов в Java EE



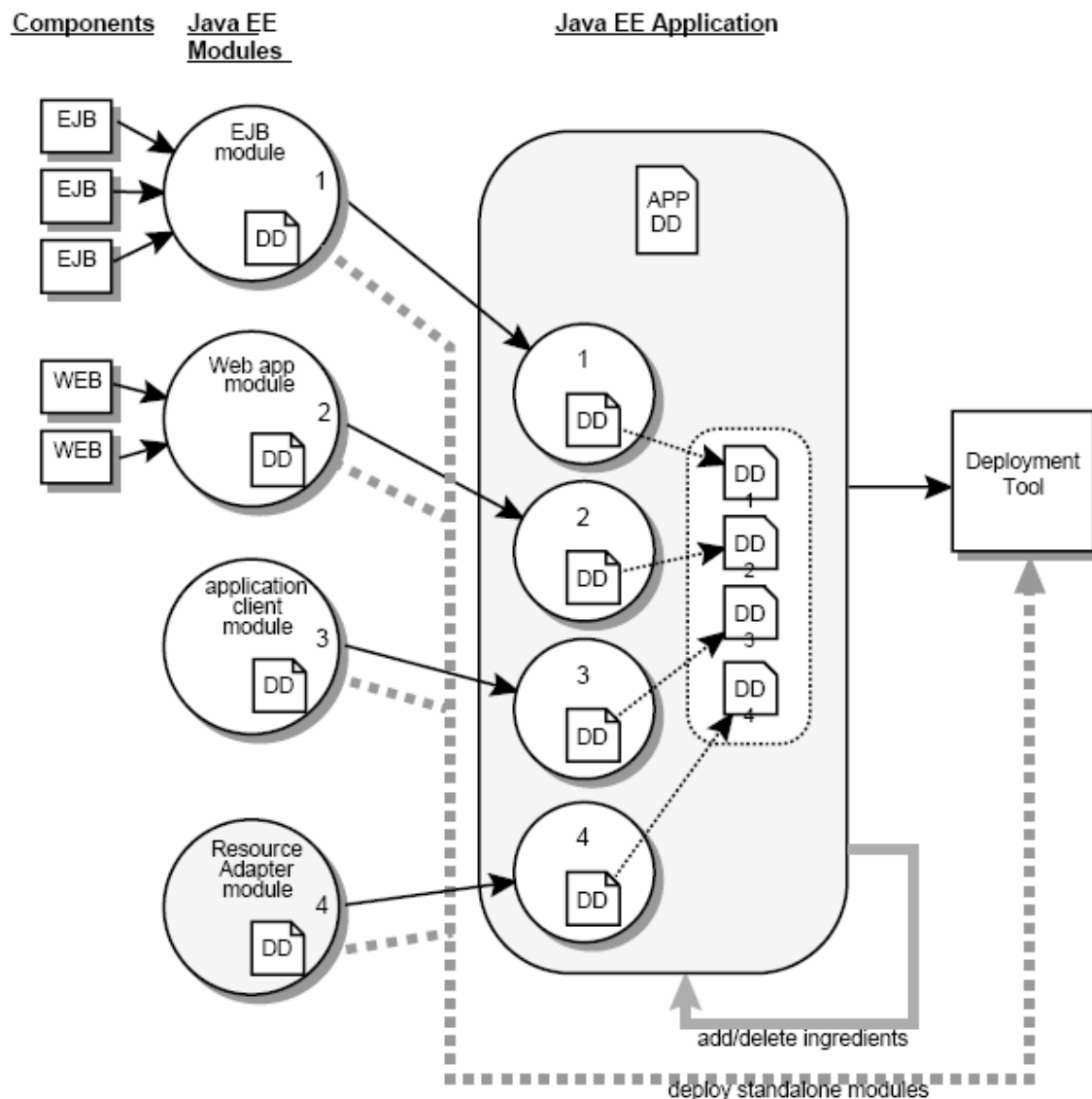
- **Веб-компоненты** — для реализации веб-интерфейса
 - > Сервлеты (servlets)
 - > Серверные страницы Java (JavaServer Pages – JSP)
- **Enterprise Java Beans (EJB)** – невизуальные компоненты для реализации бизнес-логики
 - > Сессионные компоненты (session beans)
 - > Компоненты сущностей (entity beans)
 - > Компоненты, управляемые сообщениями (message-driven beans)

Концепция Implicit Middleware



Также **аннотации**,
начиная с Java EE 5

Структура Java EE-приложения



Жизненный цикл Java EE-приложения





Роли при разработке Java EE-приложения

- Component provider
- Application assembler
- Deployer
- Platform provider
 - > Container provider
- Tools provider
- System administrator