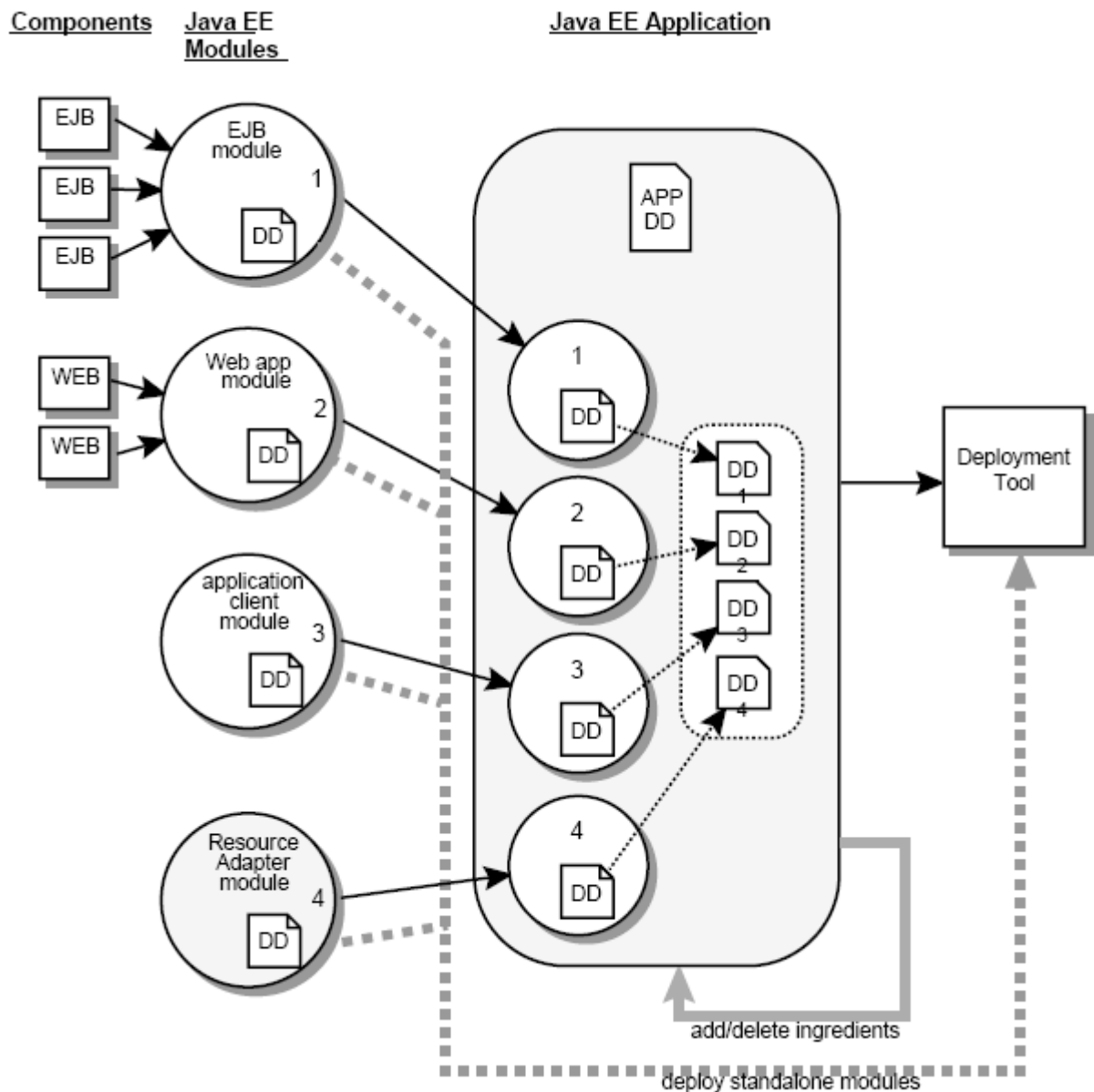


Структура Java EE-приложения

В спецификации Java EE предложен масштабируемый и модульный формат сборки приложения, который обеспечивает переносимость Java EE-приложений на любой Java EE-совместимый сервер приложений.

Java EE-приложение состоит из одного и более Java EE-модулей и необязательного установочного дескриптора для приложения. Если установочный дескриптор присутствует, то в нем описываются модули, составляющие приложение. В противном случае обнаружение модулей выполняется в соответствии с соглашениями о наименовании.



На рис. показана модель Java EE-приложения, а также необязательное использование альтернативных установочных дескрипторов, позволяющих при установке оставить без изменения Java EE-модули, подписанные ЭЦП.

Java EE-модуль представляет собой пакет, содержащий один или несколько компонентов одного типа (веб-компонент, EJB, клиентское приложение или коннектор) и необязательный установочный дескриптор соответствующего типа. Структура пакета и установочного дескриптора определяется в спецификации на компоненты определенного типа. Например, спецификация EJB Core определяет структуру EJB-модуля и дескриптора

ejb-jar.xml, спецификация сервлетов – структуру веб-приложения и дескриптора web.xml, и т.д.

Отдельный Java EE-модуль также представляет собой допустимое Java EE-приложение и может быть установлен сам по себе, без установочного дескриптора для приложения.

Java EE-приложение упаковывается с использованием формата JAR в файл с расширением .ear (сокращение от Enterprise ARchive). Установочный дескриптор для приложения (если он присутствует) должен называться META-INF/application.xml.

Помимо Java EE-модулей приложение может содержать необходимые им библиотеки классов. Для подключения библиотек используется три альтернативных способа:

1. Файл в формате JAR (с расширением .jar, .war или .rar) может ссылаться на другой JAR-файл или каталог, содержащий файлы классов, с помощью заголовка Class-Path в своем манифесте (файл META-INF/MANIFEST.MF внутри JAR-файла). Путь должен быть относительным, а при необходимости указать несколько JAR-файлов или каталогов пути к ним разделяются пробелами.

2. .ear-файл может содержать каталог с библиотеками классов, который по умолчанию называется lib. Файлы из этого каталога (но не из его подкаталогов), имеющие расширение .jar, доступны всем компонентам Java EE-приложения.

3. Веб-приложение может содержать библиотеки классов в каталоге WEB-INF/lib.

Если в Java EE-приложении отсутствует установочный дескриптор для приложения, то для определения модулей, входящих в приложение, используются следующие правила:

1. Все файлы, имеющие расширение .war, считаются веб-приложениями (веб-модулями).

2. Все файлы, имеющие расширение .rar, считаются адаптерами ресурсов.

3. Каталог lib считается каталогом с библиотеками классов.

4. Для всех файлов с расширением .jar, находящихся не в каталоге lib, выполняются следующие проверки:

- 4.1. Если JAR-файл содержит файл META-INF/MANIFEST.MF с заголовком Main-Class или содержит файл META-INF/application-client.xml, то он считается модулем клиентского приложения.

- 4.2. Если JAR-файл содержит файл META-INF/ejb-jar.xml или содержит хотя бы один класс, аннотированный как EJB-компонент, то он считается EJB-модулем.

Пример упаковки Java EE-приложений.

Java EE-приложение app1.ear содержит два EJB-модуля: ejb1.jar и ejb2.jar. При этом компоненты второго EJB-модуля используют компоненты из ejb1.jar, поэтому им нужно клиентское представление этих компонентов, которое хранится в библиотеке классов ejb1_client.jar. Для упрощения сборки приложения модуль ejb1.jar также использует эту библиотеку классов, которая в его установочном дескрипторе явно обозначена как библиотека, содержащая клиентские представления EJB-компонентов. Таким образом, app1.ear имеет следующую структуру:

```
app1.ear:
    META-INF/application.xml
    ejb1.jar      Class-Path: ejb1_client.jar
                установочный дескриптор содержит:
                <ejb-client-jar>ejb1_client.jar</ejb-client-jar>
    ejb1_client.jar
    ejb2.jar      Class-Path: ejb1_client.jar
```

Java EE-приложение app2.ear содержит EJB-модуль ejb3.jar и веб-приложение webapp.war. Эти модули также вызывают EJB-компоненты из модуля ejb1.jar

приложения app1.ear, для чего им необходима библиотека классов ejb1_client.jar. Поместим ее в каталог lib, чтобы не использовать манифесты модулей. Приложение app2.ear имеет следующую структуру:

```
app2.ear:
    META-INF/application.xml
    lib/ejb1_client.jar
    ejb3.jar
    webapp.war
        WEB-INF/web.xml
        WEB-INF/lib/servlet1.jar
```

Java EE-приложение и использование локальных интерфейсов EJB-компонентов

Локальное представление EJB-компонентов можно использовать только в компонентах, размещающихся либо в том же EJB-модуле, либо в другом модуле (EJB-модуле или веб-приложении) того же Java EE-приложения.

Если необходимо использовать EJB-компоненты из другого Java EE-приложения или из независимого EJB-модуля, то у них обязательно должно быть удаленное представление.