

Лекция 11

Програмна среда за Java. Платформа J2EE (Java 2 Platform Enterprise Edition)

С помощта на език Java могат да се създават два типа програмни продукти: програмни приложения и аплети. Приложенията са програми, които могат да се изпълняват от командния ред на операционната система. Те биват самостоятелни програми или сървлети. Самостоятелните приложения могат да работят като прозоречни или конзолни приложения.

Прозоречните приложения използват графичен потребителски интерфейс (GUI). Такива приложения използват възможностите, предоставени от платформи като Microsoft Windows, Mac OS, OS/2 и други. Конзолните приложения са приложения, работещи в текстов режим, които не използват графичен интерфейс.

Самостоятелните приложения използват ресурсите на един компютър, докато мрежовите приложения използват наличните ресурси в мрежата. Разпределените приложения могат да осъществяват достъп до обекти, които се изпълняват на различни компютри в мрежата.

Аплетите са програми на Java, които се изпълняват в рамките на Web страница. Поради тази причина, за разлика от приложенията, за да бъдат изпълнени аpletите изискват браузър, поддържащ Java, например Microsoft Internet Explorer 4.0 и по-висока версия. Аpletите се зареждат и изпълняват, като потребителят зарежда Web страница, която съдържа аплети в Web сървъра си. Те могат да се използват за добавяне на динамични елементи в Web страниците, включително звук и анимация.

Прости програмни приложения и аплети могат да се създават с помощта на обикновени или специализирани текстови редактори (например Notepad). Когато се разработва по-сложно програмно приложение, което използва графичен интерфейс, се използват множество готови програмни елементи и класове, които образуват специализираните програмни среди. Най-широко разпространение в момента получава специализираната програмна среда разработена от Sun Microsystems J2EE.

J2EE се използва за разработване, инсталиране и изпълнение на програмни приложения в разпределени среди. Чрез нея се разработват трислойни програмни приложения, които се разполагат на три различни места: машината на клиента, J2EE сървърната машина и машината, съдържаща базата данни с които работи приложението ('задна' част) – фиг. 10.1. J2EE представлява средния слой, който извършва основната обработка на данните (бизнес-логиката). Тя се състои от отделни компоненти. J2EE спецификацията използва следните компоненти:

- Клиентски приложения и аплети;
- сървлети и Java Server Pages (JSP), които са Web компоненти и се изпълняват от сървъра;
- Enterprise JavaBeans (EJB), които са бизнес-компоненти и се изпълняват от сървъра.

J2EE компонентите са програми написани на език Java и се компилирани по същия начин, както стандартните Java програми (класове). Разликата между стандартните Java класове и J2EE компонентите е, че J2EE компонентите се асемблират (комплектоват) в J2EE приложение, са проверени за съответствие с J2EE спецификацията и са подготвени за управление от J2EE сървър.

Контейнерите осигуряват интерфейс между програмните компоненти (Java компонентите) и платформената спецификация. Преди да се изпълнят клиентските компоненти те трябва да бъдат асемблирани в J2EE приложението и да бъдат разположени в съответните контейнери. Контейнерите поддържат функционалността на компонентите, кат осигуряват сигурността им, управлението на транзакциите, връзката с отдалечени ресурси и други. Web контейнерът управлява изпълнението на JSP страниците и сървлетите, а EJB контейнера управлява изпълнението на клиентските приложения.

Клиент на J2EE приложение може да бъде Web клиент (когато се разработва Web приложение) или стандартен клиент в мрежова конфигурация (за програмно приложение). **Web клиентът** съдържа две части: динамична Web страница, съдържаща код на HTML, XML или друг тип форматиращ език и Web браузър, който показва Web страницата. Web клиентът понякога се нарича *'тънък'* клиент. Такова приложение не извършва обработка на данни или изпълнява никакви бизнес правила. Web страниците могат да получават от сървъра специфичен Java код (аплет), който се изпълнява от браузъра. В този случай системата на клиента най-често се нуждае от Java Plug-in (Java виртуална машина) и програмен софтуер за защита на приложенията. Аплетите въвеждат динамично съдържание в Web страниците. Динамични Web страници могат да се реализират и с използване на Web компоненти, които са Java програми, изпълнявани на сървърната машина. Те подготвят съдържанието на страницата, която се изпраща на клиентската машина.

За предпочитане е клиентските приложения с динамично съдържание да се разработват с Web компоненти защото в този случай не е необходим Java Plug-in и защитен софтуер на клиентската машина. Освен това Web компонентите осигуряват по ясен и добре структуриран дизайн на приложението, защото се отделя програмирането на обработката на данни (бизнес логиката) от проектирането на външния вид на Web страниците. Самото проектиране на страниците не изисква познаване на Java и може да се извършва от HTML разработчици.

Стандартният клиент е приложение, което се стартира на клиентска машина и осигурява възможност на потребителя да управлява задачи, които се изпълняват върху сървъра. Той осигурява по-богат интерфейс до бизнес слоя (сървърната машина), отколкото може да се осъществи посредством форматиращ език в Web клиентските приложения. Потребителят-клиент има директен достъп до бизнес-компонентите в средния слой. Затова той се нарича *'пълен'* клиент.

J2EE приложенията са сложни, използват данни от различни източници и обслужват голямо разнообразие от клиенти. За да се реализират такива приложения, бизнес-функциите са изнесени в средния слой. J2EE платформата служи като среден слой и предоставя необходимото за такива приложения обкръжение.

Архитектурата на J2EE съдържа следните компоненти: J2EE сървър, EJB контейнер и Web контейнер. Те реализират средния слой в многослойните Java приложения (фиг. 10.1). J2EE сървърът предоставя EJB и Web контейнери. При разработване на Web приложения се използва Web контейнера, а при разработване на стандартно клиент приложение се използва EJB контейнер. Клиентът комуникира с бизнес слоя в J2EE или директно (при стандартни приложения) или посредством JSP или сървлети при Web приложения.

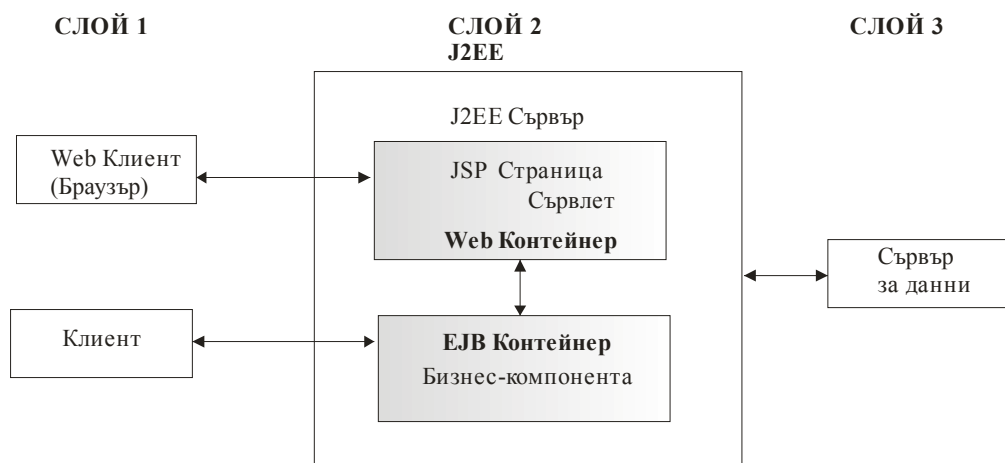
Бизнес-компонентите EJB са сървър компоненти, които се изпълняват на J2EE сървър. Те са компоненти на средния слой, съдържащи методи, които реализират бизнес правила. Чрез тях се реализира бизнес-логиката на програмното приложение. Има три вида компоненти: сесийни компоненти, компоненти предоставящи данни (*entity beans*) и компоненти предаващи съобщения (*message-driven beans*). Сесийните компоненти представят временните комуникации с клиента. Когато клиентът завърши изпълнението на дадена задача, сесийните компоненти се деактивират и връзката с данните се преустановява. За разлика от тях, компонентите предоставящи данни осигуряват сигурно съхранение на данните. Компонентите за предаване на съобщения използват така наречения Java Message Service (JMS) – списък за съобщения на Java и осигуряват на бизнес компонентите получаване на съобщения от работата на системата.

Бизнес-компонентите управляват преобразуването на данните и информационния поток между клиента (или аплети) и J2EE сървъра или между сървъра и базата данни.

EJB контейнерът управлява **бизнес-компонентите**, които се използват в J2EE приложенията. В тези компоненти се съдържа само бизнес-логиката на приложението. Не е необходимо да се пише код за системни услуги, например за управление на транзакции или за сигурността на приложението. EJB компонентите предоставят такива услуги и могат да се настройват по време на инсталиране на сървъра.

J2EE Web компоненти могат да бъдат или сървлети или JSP страници. Затова Web контейнерът управлява изпълнението на Web страници (JSP) и на сървлети.

Сървлетите са програми на Java, които могат да се поставят на Web сървър с поддръжка на Java, за да разширят и допълнят функционалността на сървъра (както аpletите допълват функционалността на браузърите). Те могат да се използват за добавяне на динамично съдържание към страници, взаимодействие с база данни или за обработка на данни от страна на сървъра. Сървлетите са Java програмни класове, които в динамичен режим осъществяват запитвания и конструират отговори към Web клиентите. Сървлетите се зареждат в паметта, когато се използват за първи път. При следващи извиквания те не се зареждат отново. Сървлетите се изпълняват в адресното пространство на Web сървъра. Тъй като те са написани на Java, те са платформено независими.



Фиг. 10.1 Архитектура на J2EE

JSP (Java Server Pages) са текстово базирани Web страници, които съдържат стандартни HTML етикети, описващи статичното съдържание на страниците и Java код, ограден със специални етикети. Java кода реализира динамичното съдържание на страници. След компилация, JSP страницата генерира сървлет. Затова JSP страниците включват пълната функционалност на сървлетите. За разлика от сървлетите, JSP страниците са по-близо до статичните Web страници, защото съдържат форматиращ текст.

Приложения с графичен интерфейс

Програмният език Java се използва най-вече за създаване на графични приложения. За целта Java съдържа класове за разработване на графичния интерфейс на приложенията. Въпреки това, примерите с основните концепции на езика са представени като конзолни приложения, защото така те са по-разбираеми. Ако се използват графични приложения, голяма част от кода описва графичния интерфейс и така се губи основата на програмните концепции на езика.

Приложенията с графичен интерфейс предоставят възможност за работа на потребителя с компоненти на графичния интерфейс като обикновени прозорци, диалогови прозорци, бутони, текстови полета и други. Тези компоненти осъществяват взаимодействието с

потребителя. Освен това, програмистът трябва да предостави функционалност за всяка от графичните компоненти в съответствие с конкретното приложение.

Разработчиците на Java предоставят мощна и гъвкава поддръжка за създаване на графичния интерфейс на приложенията. Тази поддръжка се осигурява от ресурсите вградени в програмните среди. От версия 1.2 на JDK (Java Developer Kit), в Java се предоставят все по-сложни и функционални потребителски интерфейси. Тези средства са групирани в един общ пакет, наречен Java Foundation Classes (JFC). JFC предоставя класове, които се използват лесно. В пакета JFC са включени следните основни групи класове:

- Abstract Window Toolkit (AWT);
- Swing;
- Java2D;
- Класове за подпомагане работата на хора с увреждания;
- Класове за осъществяване на drag and drop;

Abstract Window Toolkit (AWT) представляват основата на JFC. Те поддържат цветове, шрифтове, събития, обработчици на събития, подредби на контролите и други. AWT е ядрото за програмиране на графичен интерфейс.

Swing е набор от компоненти, които са олекотени и интегрируеми. Те са базирани на архитектурата Model-View Controller (MVC). Тази архитектура позволява на компонентите на Swing да променят изгледа си при различни обкръжения. Като платформено-независим език компонентите в Java трябва да приемат изгледа на конкретната платформа. В пакета Swing са включени множество графични компоненти: бутони, списъци, диалогови кутии, етикети, полета за въвеждане на информация, панели, менюта и много други. Swing съответства на пакета класове WinForms в платформата .NET Framework на Microsoft, който е в основата на визуалната среда Visual Studio .NET. Така, в J2EE Swing може да се смята като среда за визуално програмиране с Java.

Java2D е пакет от класове, които са включени в Java платформата JDK 1.2 и предоставя множество стилове за рисуване и инструменти за създаване на сложни фигури и изображения. Този пакет е проектиран като разширение на графичните възможности на AWT.

Класове за подпомагане на хора с увреждания (Accessibility API) предлагат технологии за улеснения за представянето на информацията, като четци на текст, промяна на параметрите на екрана и други.

Класове за осъществяване на drag and drop. Тези класове позволяват преместването на данни между програми създадени с Java.