

TJV – BI-WSI-SI-29

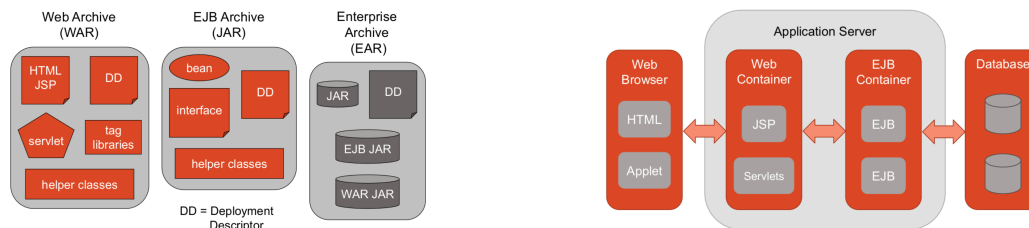
Aplikační server a JNDI služba: popis aplikačního serveru, webový kontejner, EJB kontejner, využití JNDI ke konfiguraci datových zdrojů.

Obsah

1	Aplikační server	2
2	JNDI	3

1 Aplikační server

- Serverovou aplikaci zastřešující všechny knihovny, které dle specifikací Java EE platformy zajišťují požadovanou funkcionalitu, označujeme pojmem aplikační server.
- Tyto knihovny implementují veškerá API obsažená v Java EE.
- Kromě toho poskytuje aplikační server další klasické služby jako např.
 - administrátorskou konzoli,
 - logování atp.
- Ze známých implementací Java EE platformy můžeme zmínit např. JBoss od firmy Red Hat či JRun od firmy Adobe Systems. Referenční implementace Oracle je šířena pod hlavičkou projektu GlassFish.
- Zajišťuje:
 - Vývoj webových aplikací - Java Servlets, Java Server Pages (JSP), JavaServer Faces (JSF)
 - Contexts and Dependency Injection - vkládání závislostí
 - Přístup k relačním databázím - Java Persistence API (ORM)
 - Vývoj sdílené business logiky - Enterprise Java Beans (EJB)
 - Přístup k legacy systémům - Java Connector Architecture (JCA)
 - Přístup ke zprávovému middleware - Java Messaging Services (JMS)
 - Komponenty zajišťující integraci webových aplikací a portálů - Portlety
 - Podpora technologií Webových služeb- SOA, REST



Aplikační server obsahuje dva kontejnery - webový a EJB. Ty obsahují komponenty, instantizují je a řídí jejich životní cyklus. Komponenty nemohou žít bez kontejneru.

Výsledkem je .ear balíček, který je vytvořen třeba vývojovým prostředím za pomoci Maven, Antu nebo novějšího Gradlu. Všechny tři zmíněné nástroje jsou tzv. build systémy. Další ze známějších je Make (známe z BI-PA2), ale ten se tady nepoužívá.

EJB komponenty

- Odpovídá business vrstvě z otázky architektura aplikací
- Více o EJB v otázce EJB
- Balíček jar

Webový kontejner (podobně jako EJB kontejner - viz EJB otázka)

- Mapuje URL adresy na servlety
- Zajišťuje bezpečnost - může uživatel přistoupit k danému kontejneru?

- Řídí životní cyklus servletů - vytváření, mazání
- Pracuje s dotazy a odpověďmi serveru
- Spravuje pool servletů

Webové komponenty

- odpovídá definici webové vrstvy z otázky architektura aplikací
- Balíček .war

2 JNDI

Java Naming and Directory Interface

Aplikační server zajišťuje službu JNDI. Ta na základě klíčového slova (unikátního názvu zdroje) zajistí vyhledání a předání existující konfigurace žádající aplikaci. Využívá se třeba pro získání instance EJB nebo DataSource (pro připojení k databázi).

Aplikační server dále zajišťuje připojení ke zdroji a obnovení komunikace, pokud se něco podělá. Pro získání instancí slouží třída `InitialContext`, kterou si můžeme kdykoliv vytvořit.

Pomocí metody `InitialContext#lookup(Name)` lze získat instanci objektu. Pokud pracujeme s databází, tak je potřeba v nastavení aplikačního serveru nastavit příslušný JDBC Resource a dát mu JNDI jméno (String) - viz otázka JDBC.

Jméno EJB je generováno automaticky ze jména třídy a umístění.

```
InitialContext ic = new InitialContext();
DataSource source = (DataSource) ic.lookup("jdbc/my_resource_name");
MyBean bean = (MyBean) ic.lookup("java:com/env/MyBean");
```