

JMS

Java Message Service

Sadržaj

- Uvod
- Arhitektura
- Vrste komunikacije
- Načini primanja poruka
- JMS Api programski model
- Poruke
- Podešavanja

Uvod

- JMS je deo Java Platform, Enterprise Edition (Java EE)
- Razmena poruka između Java EE komponenata
- Standard za pristup Messaging middleware serveru
- Slabo spregnuta arhitektura koja omogućava olakšan i jednostavan način za komunikaciju, tj. razmenu poruka. Primalac i pošiljalac ne moraju postojati u isto vreme. Primalac ne mora čekati na poruku u trenutku kada je pošiljalac šalje kako bi je primio.
- Jednostavan i intuitivan API za ove svrhe

Messaging middleware server

- **Pouzdanost** - posrednički softver za razmenu poruka garantuje isporuku. Ako potrošač ne radi iz nekog razloga, sistem za razmenu poruka skladišti poruku dok ponovo ne proradi.
- **Asinhroni** - JMS je asinhroni, pošiljalac i primalac ne moraju da čekaju jedni druge.
- **Jednostavnost integracije** - Više aplikacija može biti napisano na različitim programskim jezicima ili pokrenuto na različitim serverima. Sve dok koriste isti protokol, posrednički softver za razmenu poruka može se koristiti za uspostavljanje komunikacije između njih.
- **Skalabilnost** - razmena poruka pomaže u emitovanju na više prijemnika što pomaže u skalabilnosti.
- **Fleksibilnost** - različitim porukama možete dodeliti različite protokole, što vam omogućava veliku fleksibilnost u dizajniranju vaše arhitekture.

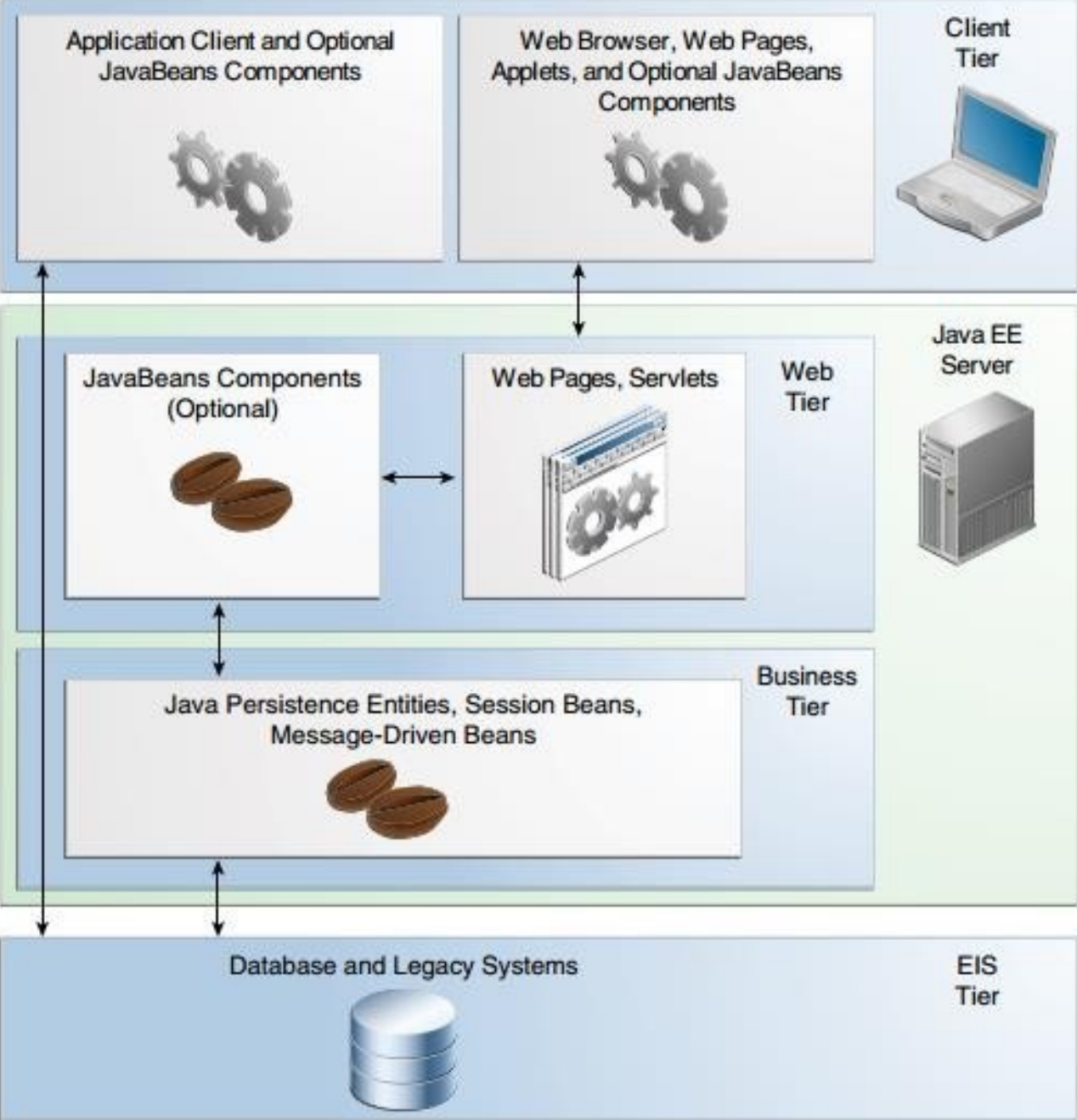
Java Message Service

- JMS pruža standardni Java API i time uobičajeni način Java aplikacijama da kreiraju, šalju, primaju i čitaju poruke između aplikacija.
- Neki od Messaging middleware servera koji ih implementiraju JMS su Apache ActiveMQ, Oracle WebLogic Server, **Glassfish**, IBM MQ, Jboss Wildfly,...

Okruženja, alati, serveri...

- Apache Netbeans 12 ili noviji
 - Development Environment
 - Tooling Platform
 - Application Framework
 - Trenutno je aktuelna verzija 19
- Glassfish Server 5.1
 - aplikativni server

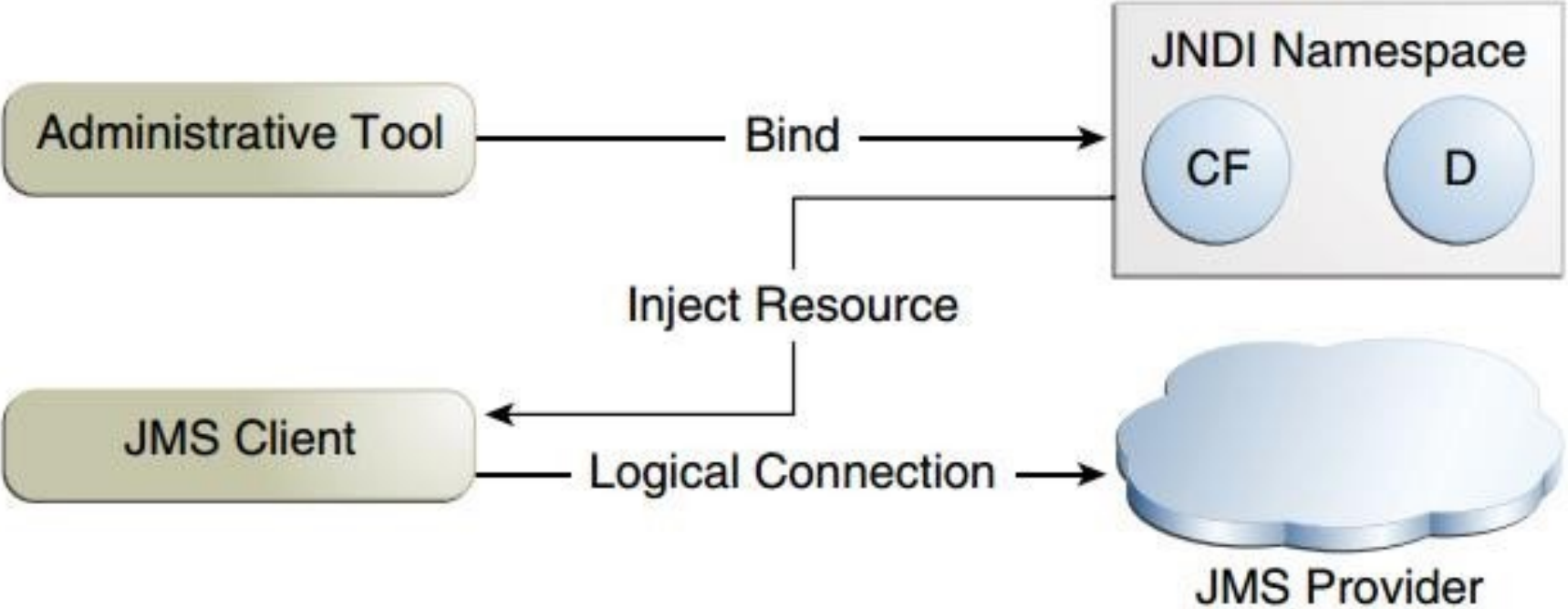
Java EE



JMS arhitektura

- **JMS clients** - Java kod koji šalje i prima poruke.
- **Messages** - Podaci o preduzeću koji se šalju ili primaju.
- **JMS provider (Message Oriented Middleware)** - sistem za razmenu poruka koji implementira JMS kao i druge administrativne funkcije.
- **Administered objects** - odredišta za poruke (redovi, teme) i fabrike veza koje su unapred konfigurisane u okviru JMS provider-a.

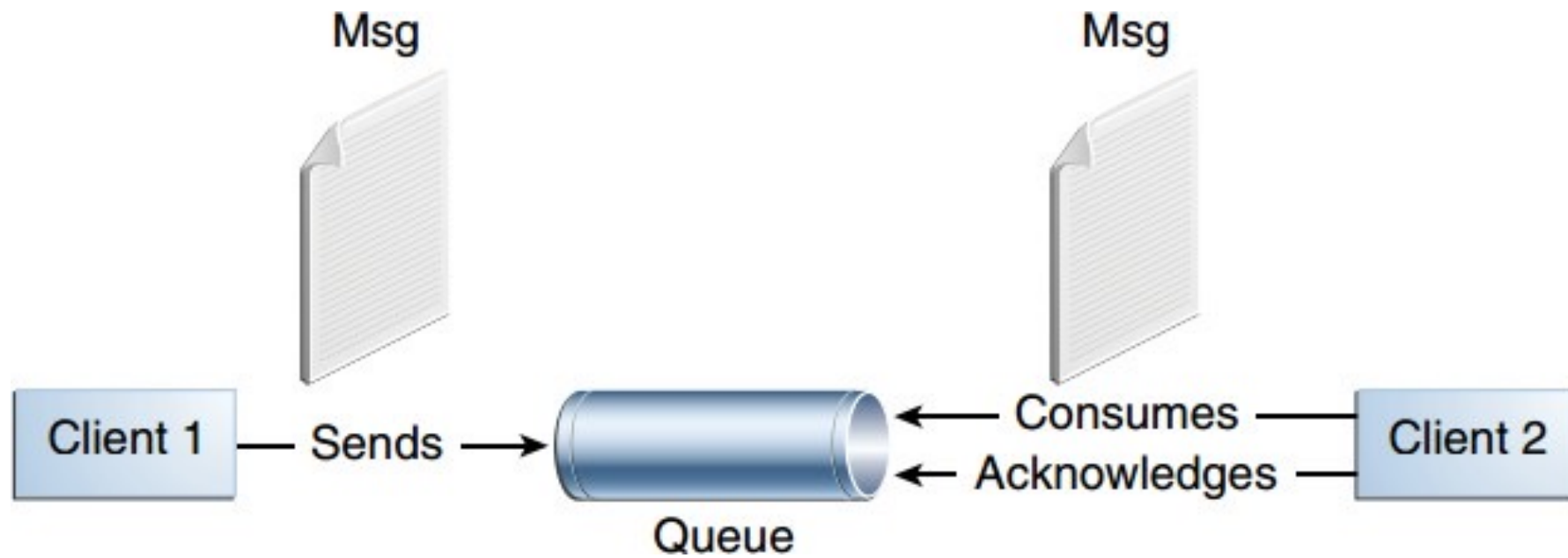
JMS architektura



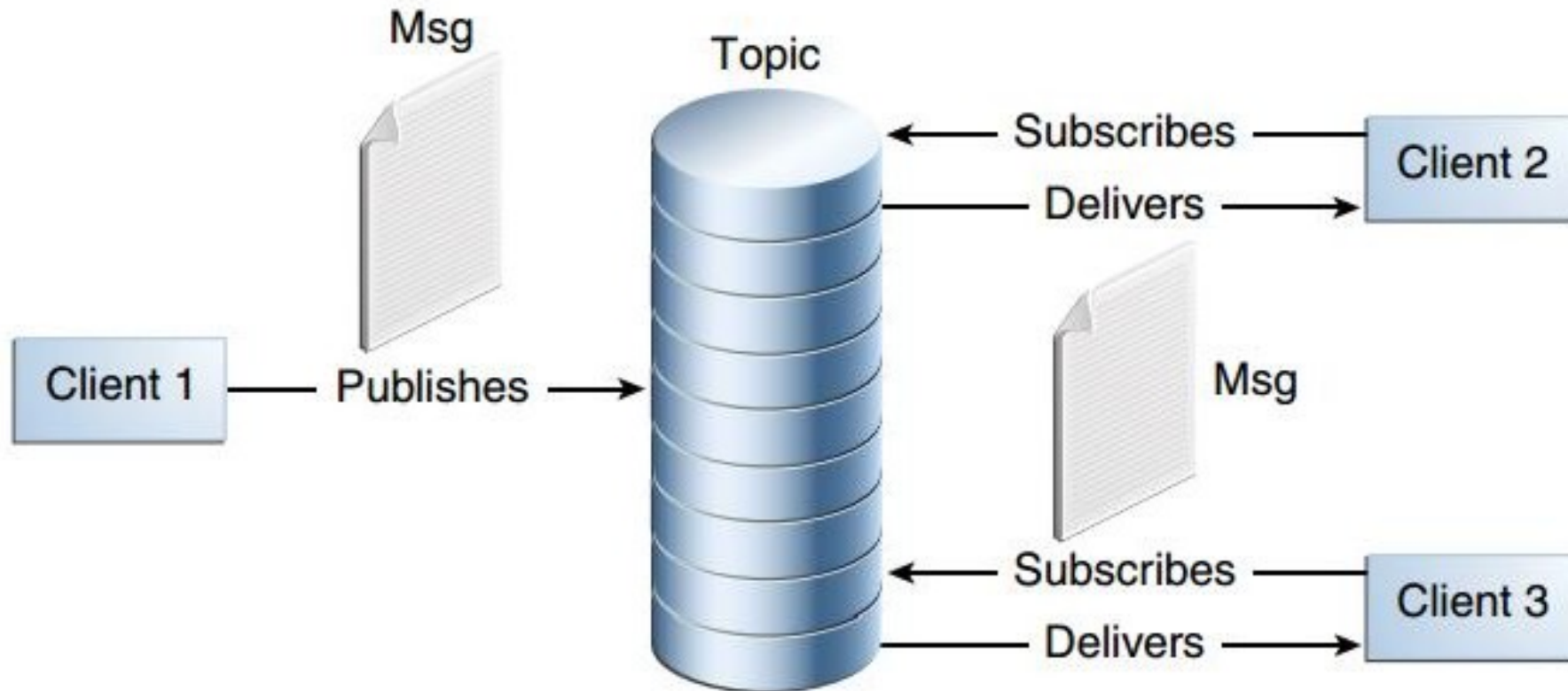
Vrste komunikacije

- Point-to-point (PTP)
- Publish/subscribe (pub/sub)

Point-to-point (PTP)



Publish/subscribe (pub/sub)

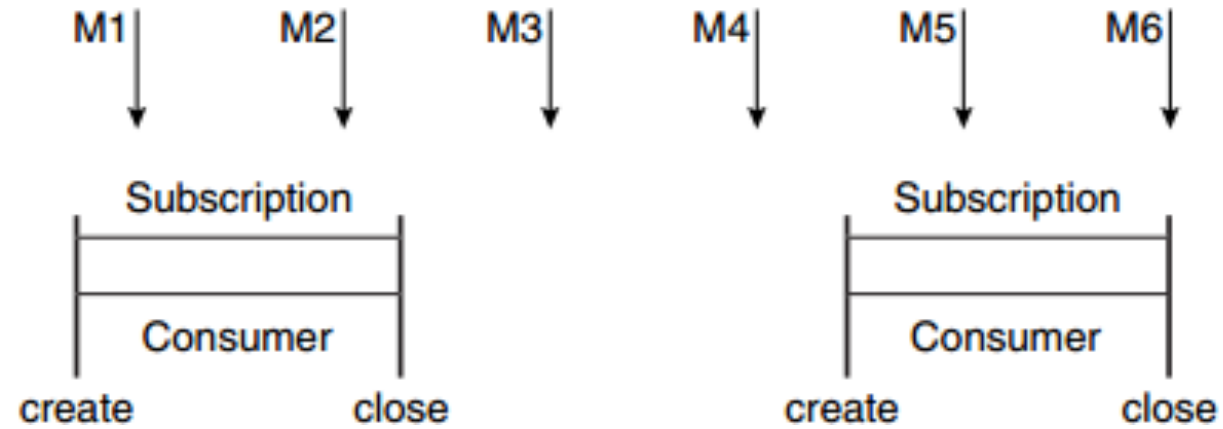


Publish/subscribe- Vrste prijava

	Shared	Not Shared
Non Durable		
Durable		

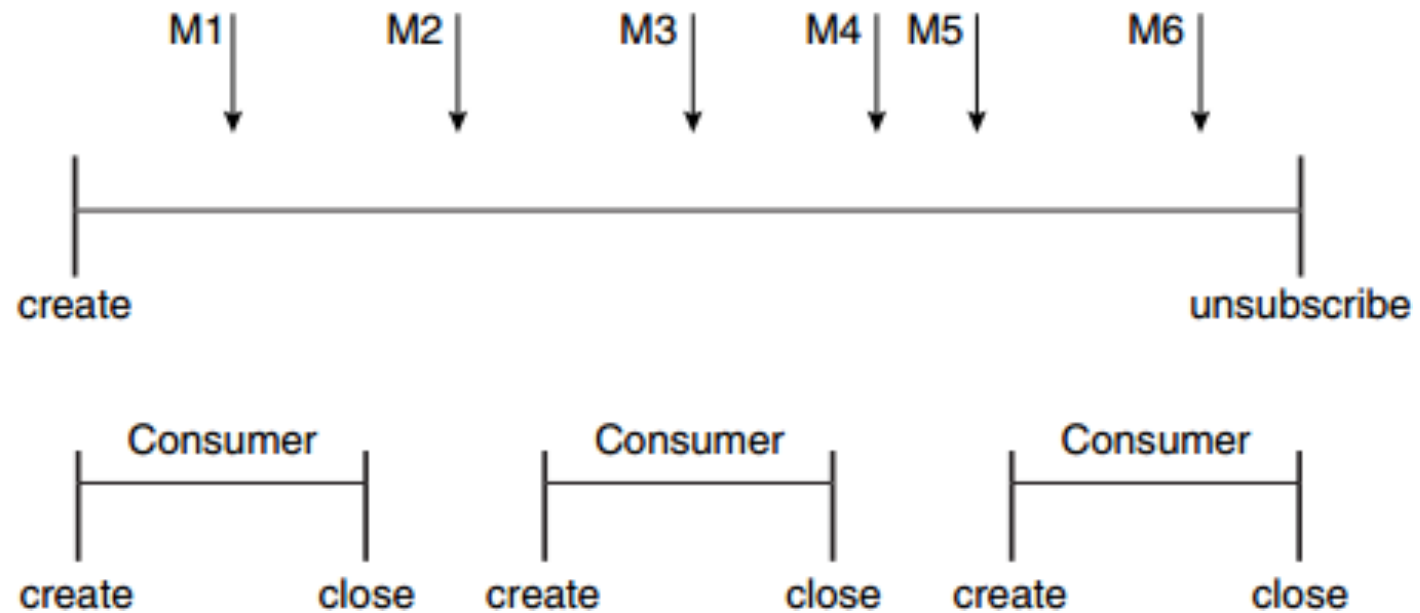
Nondurable Subscription

Figure 45–6 Nondurable Subscriptions and Consumers



Durable Subscription

Figure 45-7 Consumers on a Durable Subscription



Načini primanja poruka

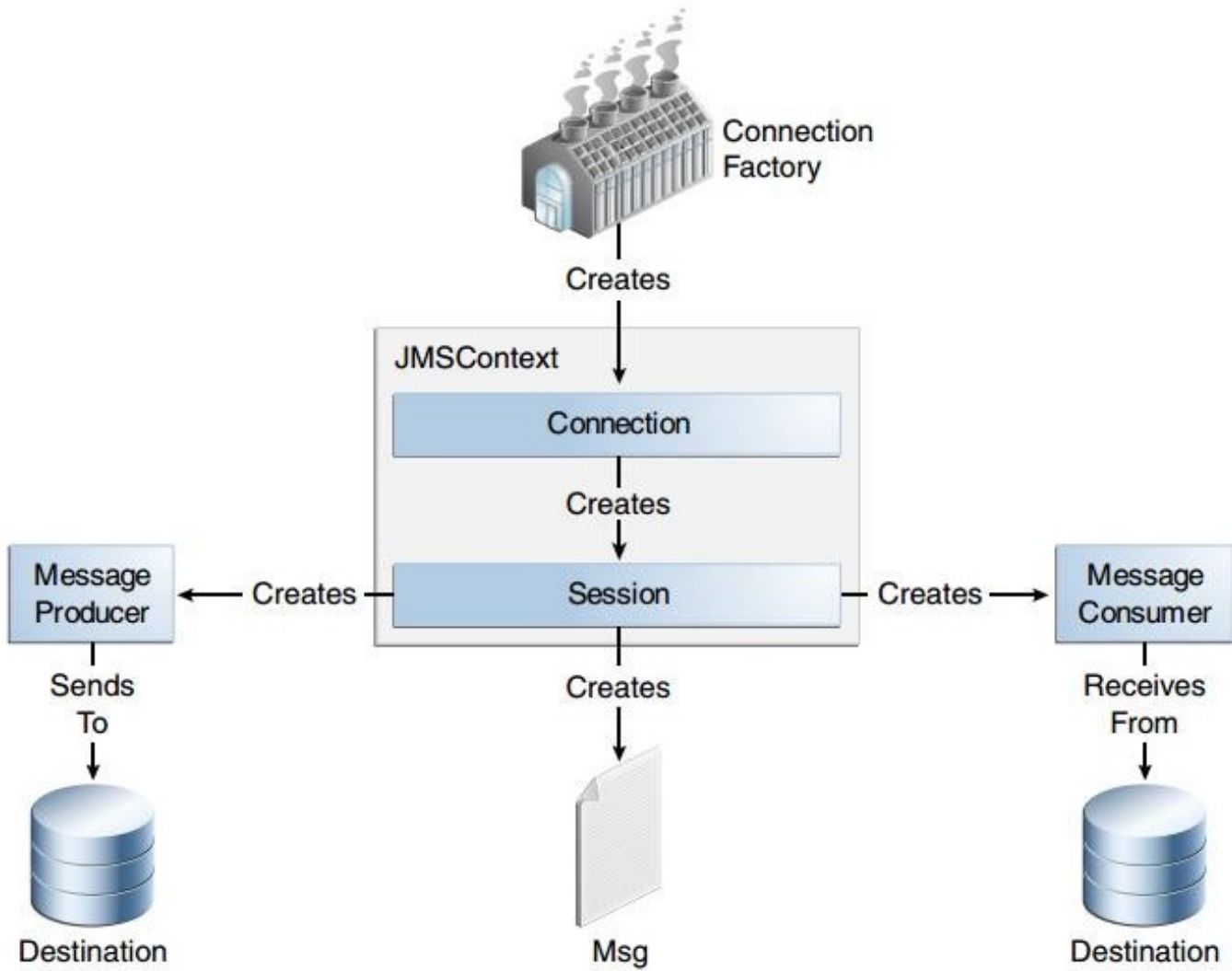
- **Sinhrono** – klijent se blokira dok ne dobije poruku
- **Asinhrono** – klijent koristi osluškivača (MessageListener) kao dodatnu nit u kojoj prima poruku

JMS API programski model

Bazični blokovi JMS aplikacije su:

- **Administered objects** - connection factories i destinations
- **Connections**
- **Sessions**
- **JMSContext objects** – spoj connection i session u jedan objekat
- **Message producers**
- **Message consumers**
- **Messages**

JMS API programski model



Poruke

Delovi poruke su:

- **Headers** (obavezni)
- **Properties** (opcioni) – mogu se dodavati na poruku
- **Body** (opcioni)

Headers

Header Field	Description	Set By
JMSDestination	Destination to which the message is being sent	JMS provider send method
JMSDeliveryMode	Delivery mode specified when the message was sent (see Specifying Message Persistence)	JMS provider send method
JMSDeliveryTime	The time the message was sent plus the delivery delay specified when the message was sent (see Specifying a Delivery Delay)	JMS provider send method
JMSExpiration	Expiration time of the message (see Allowing Messages to Expire)	JMS provider send method
JMSPriority	The priority of the message (see Setting Message Priority Levels)	JMS provider send method
JMSMessageID	Value that uniquely identifies each message sent by a provider	JMS provider send method
JMSTimestamp	The time the message was handed off to a provider to be sent	JMS provider send method
JMSCorrelationID	Value that links one message to another; commonly the JMSMessageID value is used	Client application
JMSReplyTo	Destination where replies to the message should be sent	Client application
JMSType	Type identifier supplied by client application	Client application
JMSRedelivered	Whether the message is being redelivered	JMS provider prior to delivery

Body

Message Type	Body Contains
TextMessage	A <code>java.lang.String</code> object (for example, the contents of an XML file).
MapMessage	A set of name-value pairs, with names as <code>String</code> objects and values as primitive types in the Java programming language. The entries can be accessed sequentially by enumerator or randomly by name. The order of the entries is undefined.
BytesMessage	A stream of uninterpreted bytes. This message type is for literally encoding a body to match an existing message format.
StreamMessage	A stream of primitive values in the Java programming language, filled and read sequentially.
ObjectMessage	A <code>Serializable</code> object in the Java programming language.
Message	Nothing. Composed of header fields and properties only. This message type is useful when a message body is not required.

Resource injection

- Administratorski server može definisati objekte koje će moći da koriste sve komponente jedne Java EE aplikacije (servlet, enterprise bean, managed bean, web component, java client component).
- Resursi bivaju kreirani unutar JNDI (Java Naming and Directory Interface) prostora imena
- Java komponente ih mogu koristiti koristeći sledeću anotaciju `@Resource(lookup = „jndiResourceName“)`

JMS Queue Browsers

- JMS Queue Browsers objekti se koriste radi prolaska kroz sve poruke koje postoje u redu i čitanja informacija vezanih za svaku od njih, bez njihovog uklanjanja iz reda.
- JMS Queue Browsers objekte kreira JMSContext instanca.

JMS Message Selector

- JMS Message Selectors omogućava primaocu poruka da navede poruke koje ga zanimaju i da prima isključivo njih.
- Filtriranje poruka se radi na osnovu Message Headers i Message Properties.
- Način rada selektora zavisi od tipa destinacije:
 - Queue - vraćaju se samo poruke koje odgovaraju selektoru, dok ostale poruke ostaju u redu.
 - Topic - poruke koje se ne odgovaraju selektoru biće zanemarene kao da nisu ni objavljene.

Glassfish server

- Aplikativni server – Java EE aplikacije rade na ovom serveru
- Domen je skup jedne ili više insanci Glassfish servera i domenom upravlja administratorski server. Podrazumevani postojeći domen je „domain1“.
- Elementi vezani za domen su:
 - Glassfish podrazumevani broj porta je 8080
 - Podrazumevani broj porta administratorskog servera je 4848
 - Pristup administratorskom server putem linka <http://localhost:4848/>
 - Podrazumevano administratorsko korisničko ime i lozinka su: „admin“, „“.

Glassfish server - komande

- Pokretanje Glassfish-a servera:
asadmin start-domain domain_name
- Zaustavljanje Glassfish-a servera:
asadmin stop-domain domain_name
- Pokretanje Enterprise Client Application:
appclient -client jarFileInTargetFolderAfterBuild.jar

Glassfish server - komande

- Kreiranje JMS resursa:

```
asadmin --host hostIPAdress --port adminServerPort create-jms-resource --restype javaClassName JNDIResourceName
```

```
asadmin add-resources pathToXMLFileResourceputanja
```

- Pregled JMS resursa:

```
asadmin list-jms-resources
```

- Brisanje JMS resursa:

```
asadmin delete-jms-resource JNDIResourceName
```