# Enterprise Java (BI-EJA) Technologie programování v jazyku Java (X36TJV)

Ing. Zdeněk Troníček, Ph.D.

Katedra softwarového inženýrství

Fakulta informačních technologií ČVUT v Praze

Letní semestr 2010/2011, přednáška č. 6
https://edux.fit.cvut.cz/courses/BI-EJA
https://edux.feld.cvut.cz/courses/X36TJV

## Agenda

- Java Message Service (JMS)
- Message Driven Beans (MDB)
- Java Web Start (JWS)

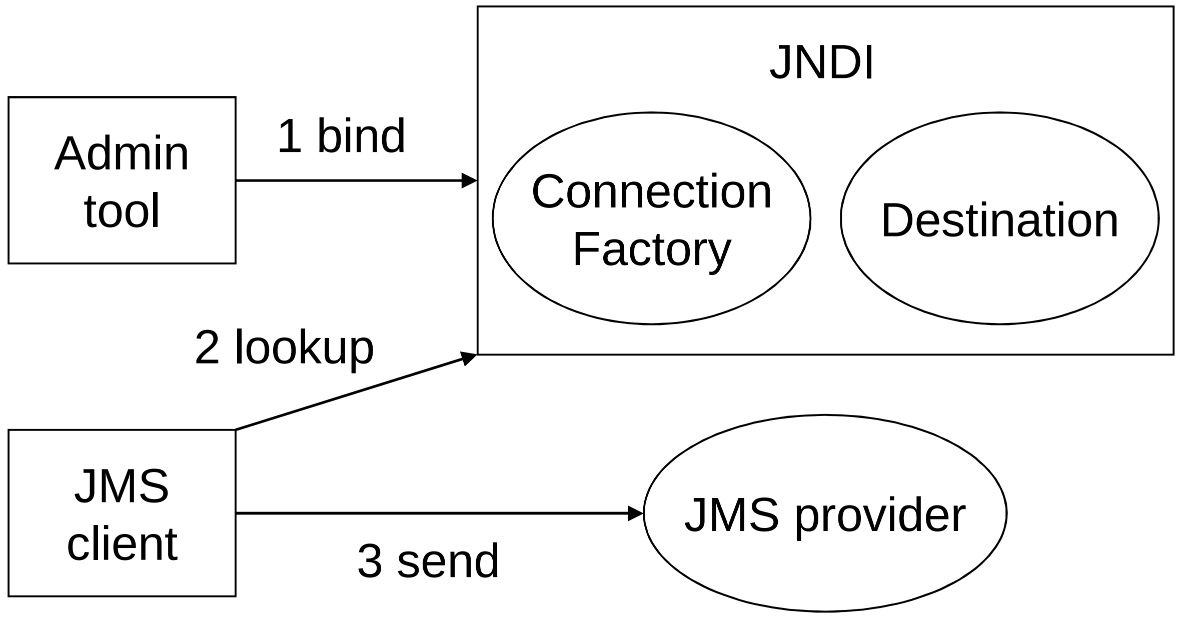
# Java Message Service

Vlastnosti

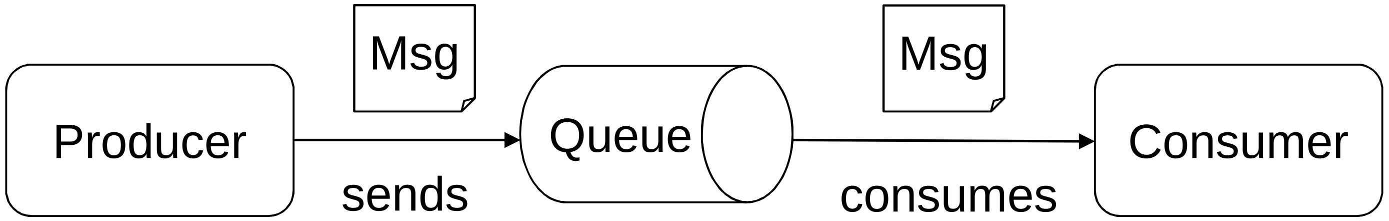
- reliable
- asynchronous
- transactional
- loosely coupled

Módy

- point-to-point
- publish/subscribe

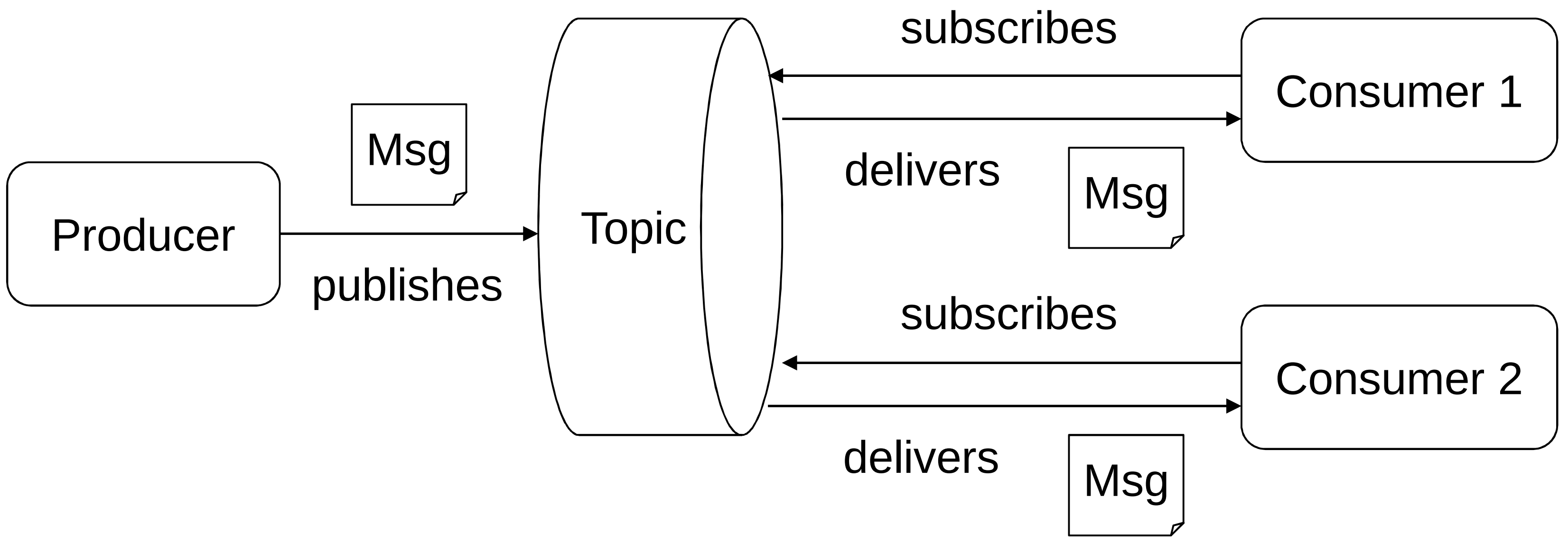


# Point-to-point



každou zprávu zpracuje 1 příjemce

## Publish/subscribe

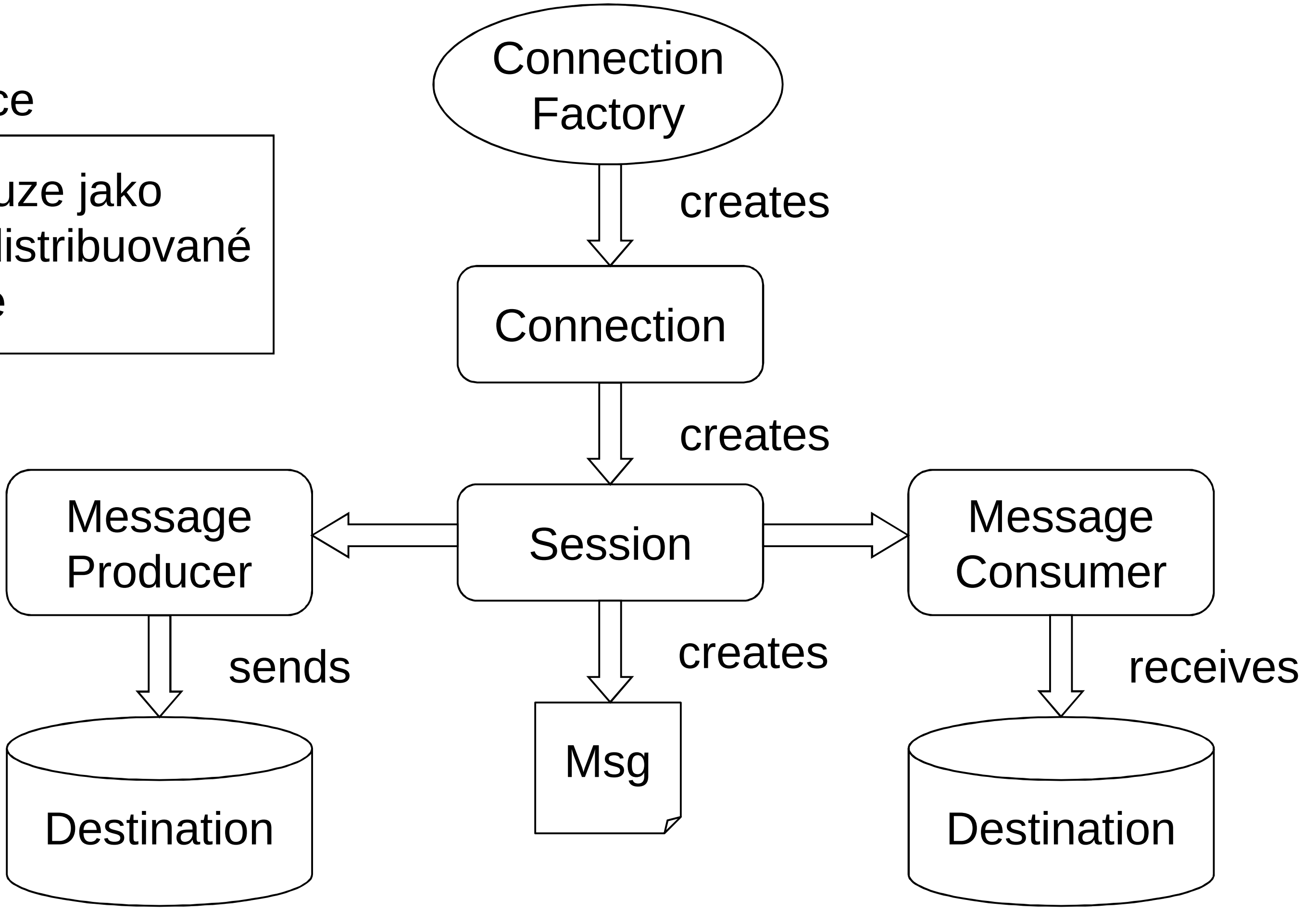


zprávu může dostat více příjemců

# Programový model

Transakce

v JEE pouze jako součást distribuované transakce



## Příklad

```
@Resource( name = "jms/statementQueue" )
private Queue statementQueue;

@Resource( name = "jms/statementQueueFactory" )
private ConnectionFactory statementQueueFactory;
```

```
Connection con = statementQueueFactory.createConnection();
Session session = con.createSession( false, Session.AUTO_ACKNOWLEDGE );
MessageProducer producer = session.createProducer( statementQueue );
TextMessage tm = session.createTextMessage( msg );
producer.send( tm );
```

## Synchronní a asynchronní příjem

synchronní příjem

interface MessageConsumer

- metoda receive()
- metoda receive( timeout )
- metoda receiveNoWait()

asynchronní příjem

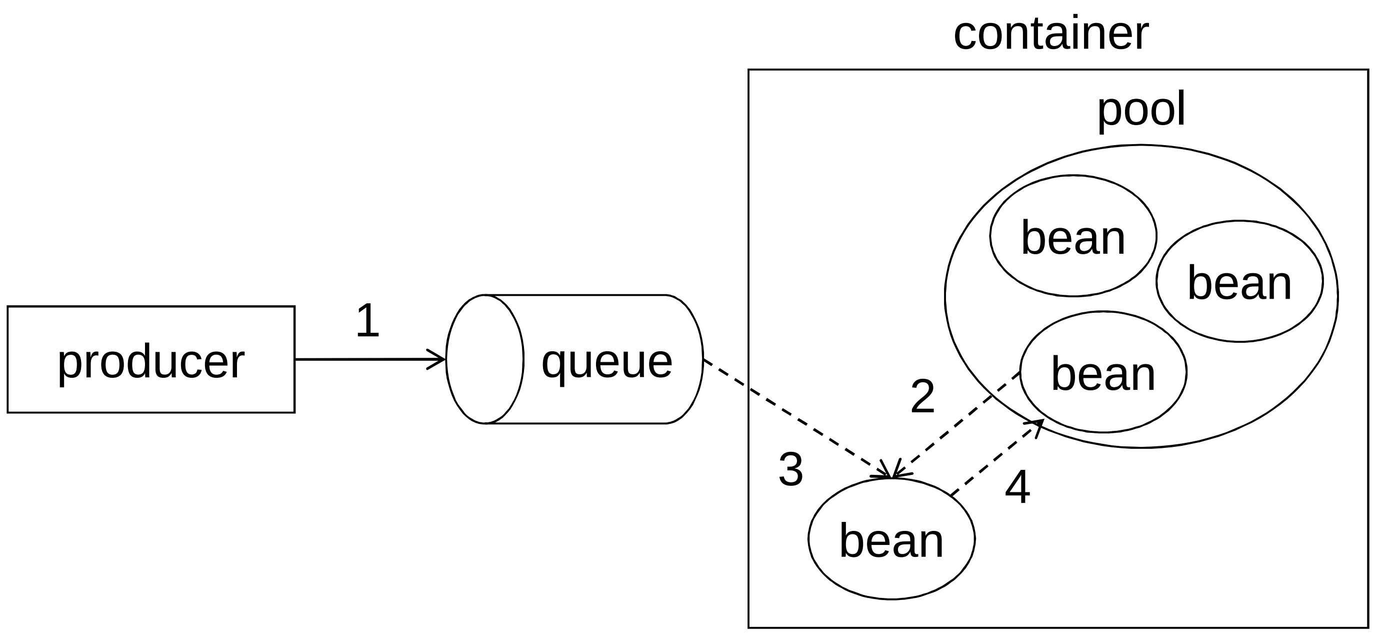
interface MessageListener

- metoda onMessage()

interface MessageConsumer

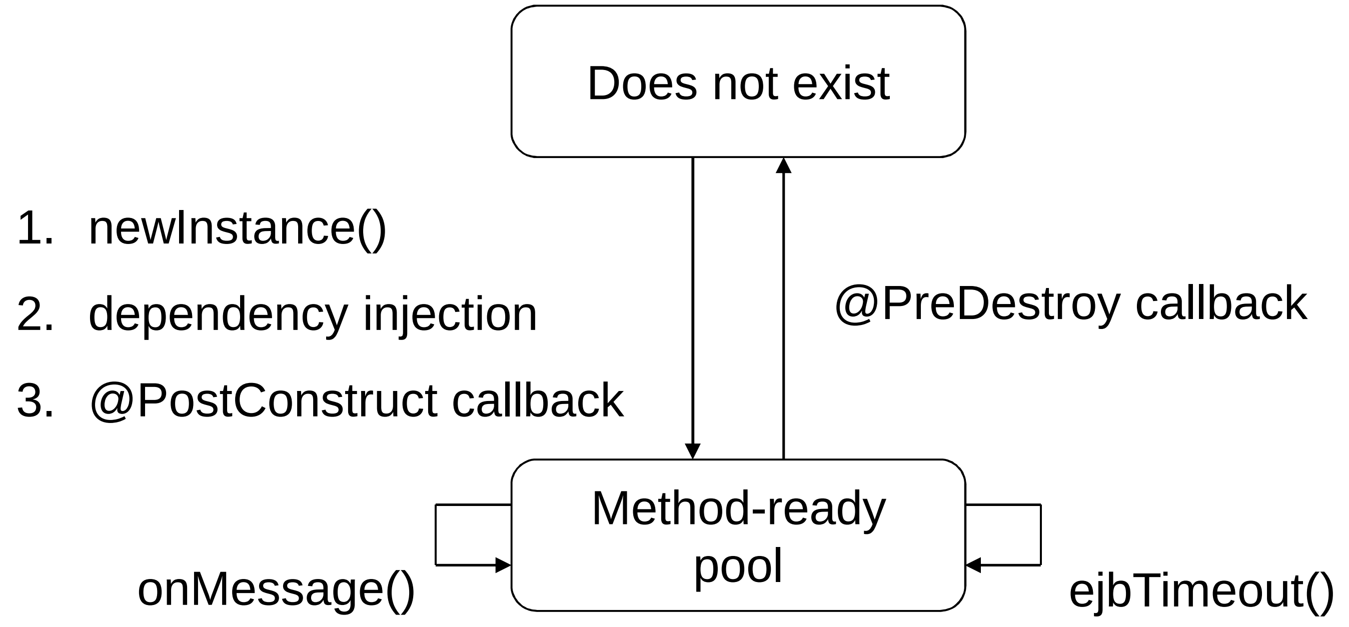
- metoda setMessageListener()

# Message Driven Bean



```
@MessageDriven( mappedName = "jms/MyQueue", … )
public class MyMDB implements MessageListener {
    @Override
    public void onMessage( Message message ) { ... }
}
```

# Životní cyklus

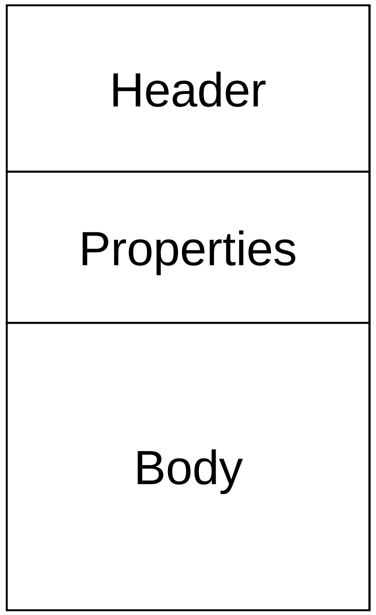


## Příklad

```
@MessageDriven( mappedName = "jms/statementQueue", … )
public class MessageBean1 implements MessageListener {

    @Override
    public void onMessage( Message message ) {
        try {
            TextMessage tm = (TextMessage) message;
            …
        } catch ( Exception e ) { … }
    }
}
```

# Message



služební informace (JMSMessageID, JMSPriority, JMSDestination, JMSExpiration, ...)

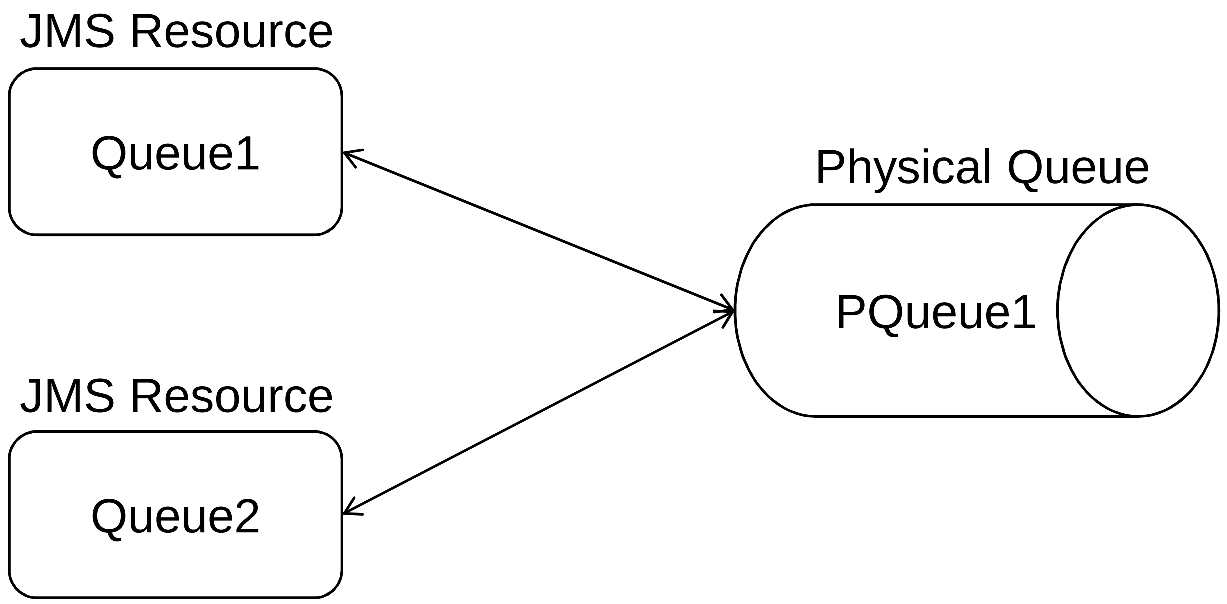
informace pro výběr a filtrování zpráv

obsah zprávy

Typy zpráv

- TextMessage
- MapMessage
- BytesMessage
- StreamMessage
- ObjectMessage

## Physical Destination



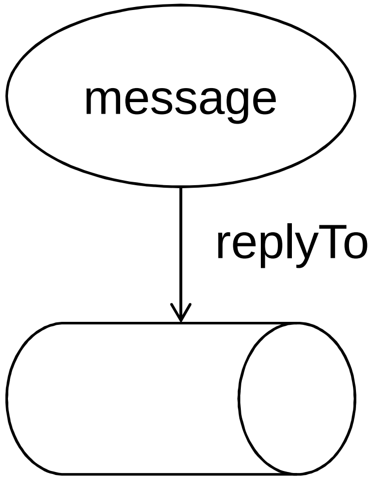
## Temporary Destination

Producer

```
MessageProducer prod = ...
Message msg = …
Queue replyTo =  session.createTemporaryQueue();
msg.setJMSReplyTo( replyTo );
producer.send( msg );
```

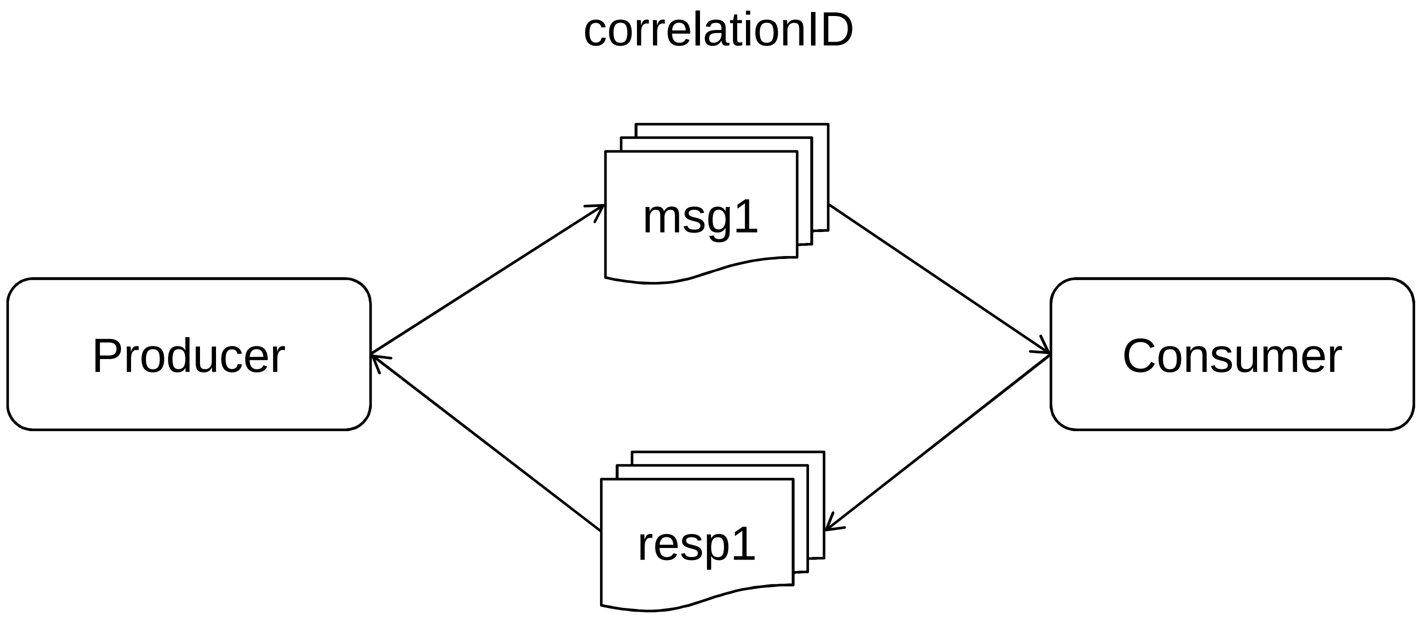
Consumer

```
Destination dest = msg.getJMSReplyTo();
MessageProducer prod = …
Message msg = …
prod.send( replyTo, msg );
```



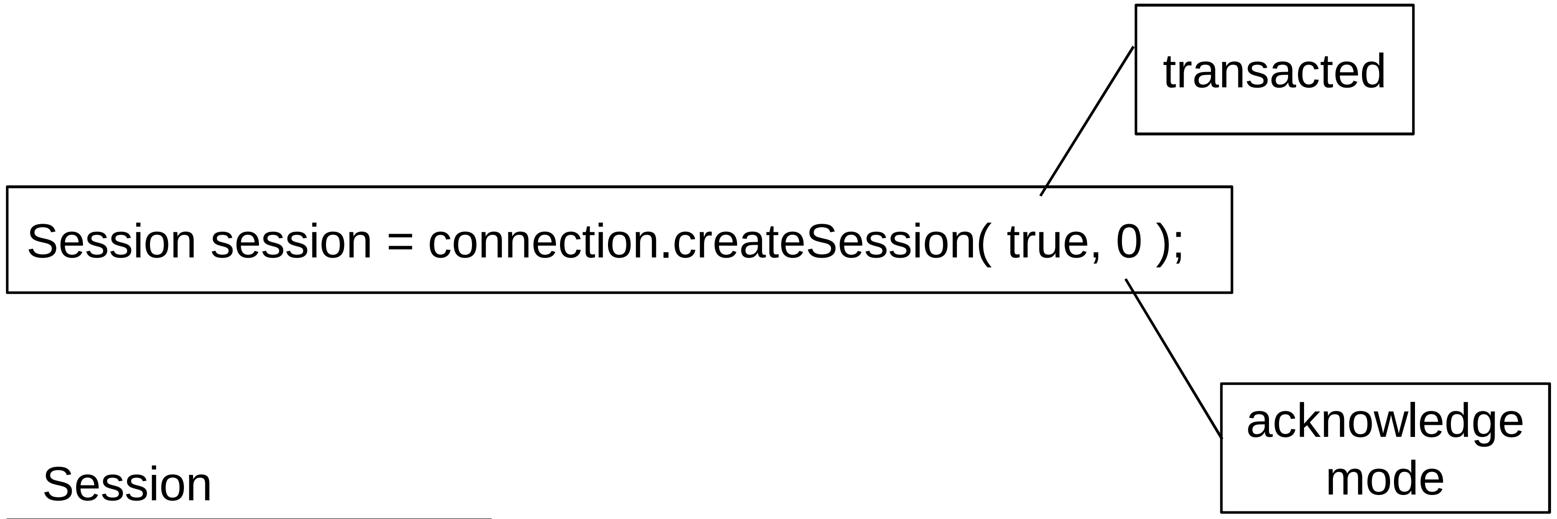
temporary queue

# CorrelationID



```
Message msg = …
msg.setJMSCorrelationID( "msg1" );
```

# Transakce



```
Session session = connection.createSession( true, 0 );
```

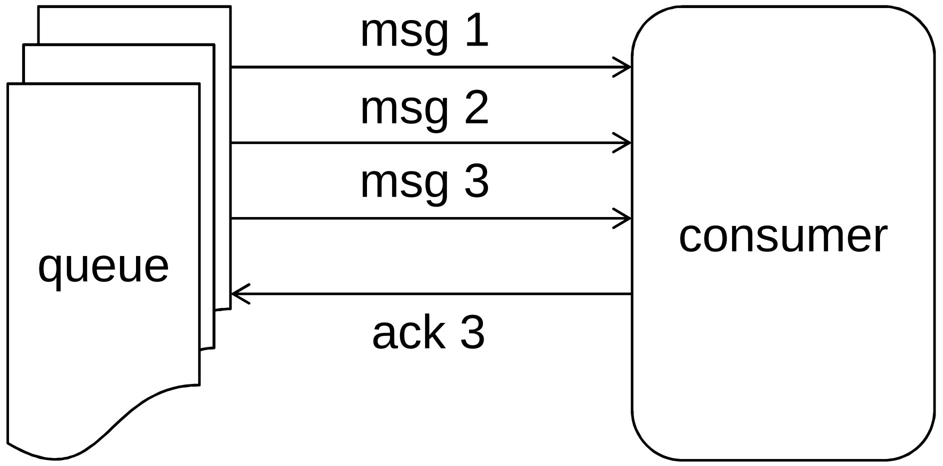
Session

- metoda commit()
- metoda rollback()

## Potvrzování

- AUTO_ACKNOWLEDGE
- CLIENT_ACKNOWLEDGE
- DUPS_OK_ACKNOWLEDGE

DUPS_OK_ACKNOWLEDGE

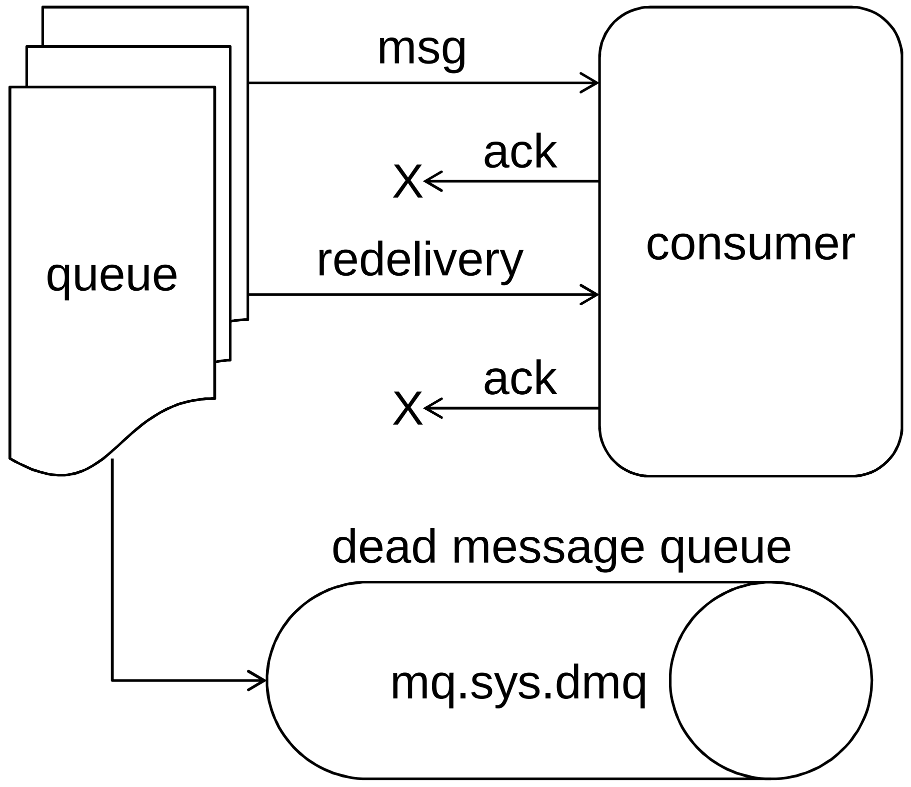


CLIENT_ACKNOWLEDGE

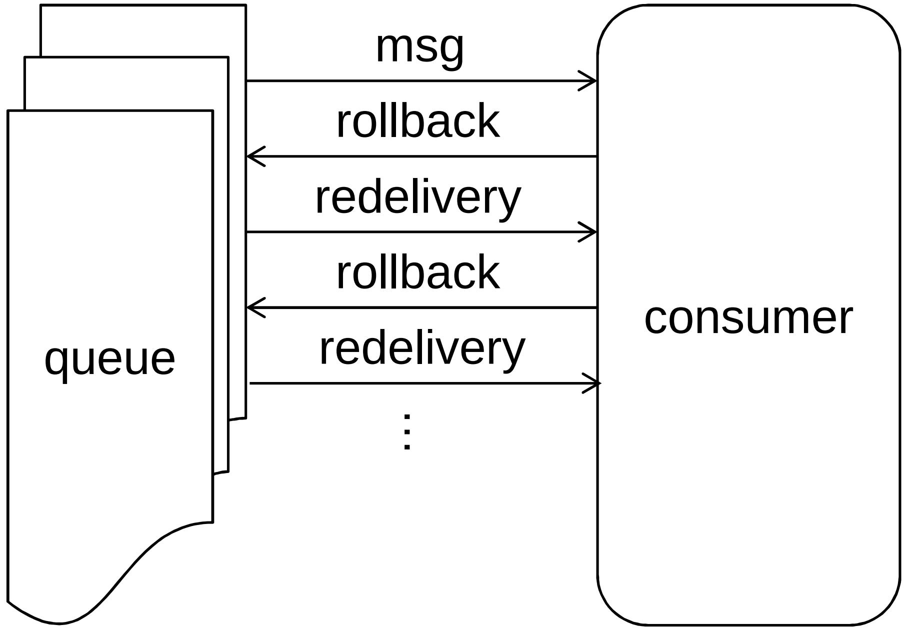
```
Message msg = …
…
msg.acknowledge();
```

## Dead Message Queue

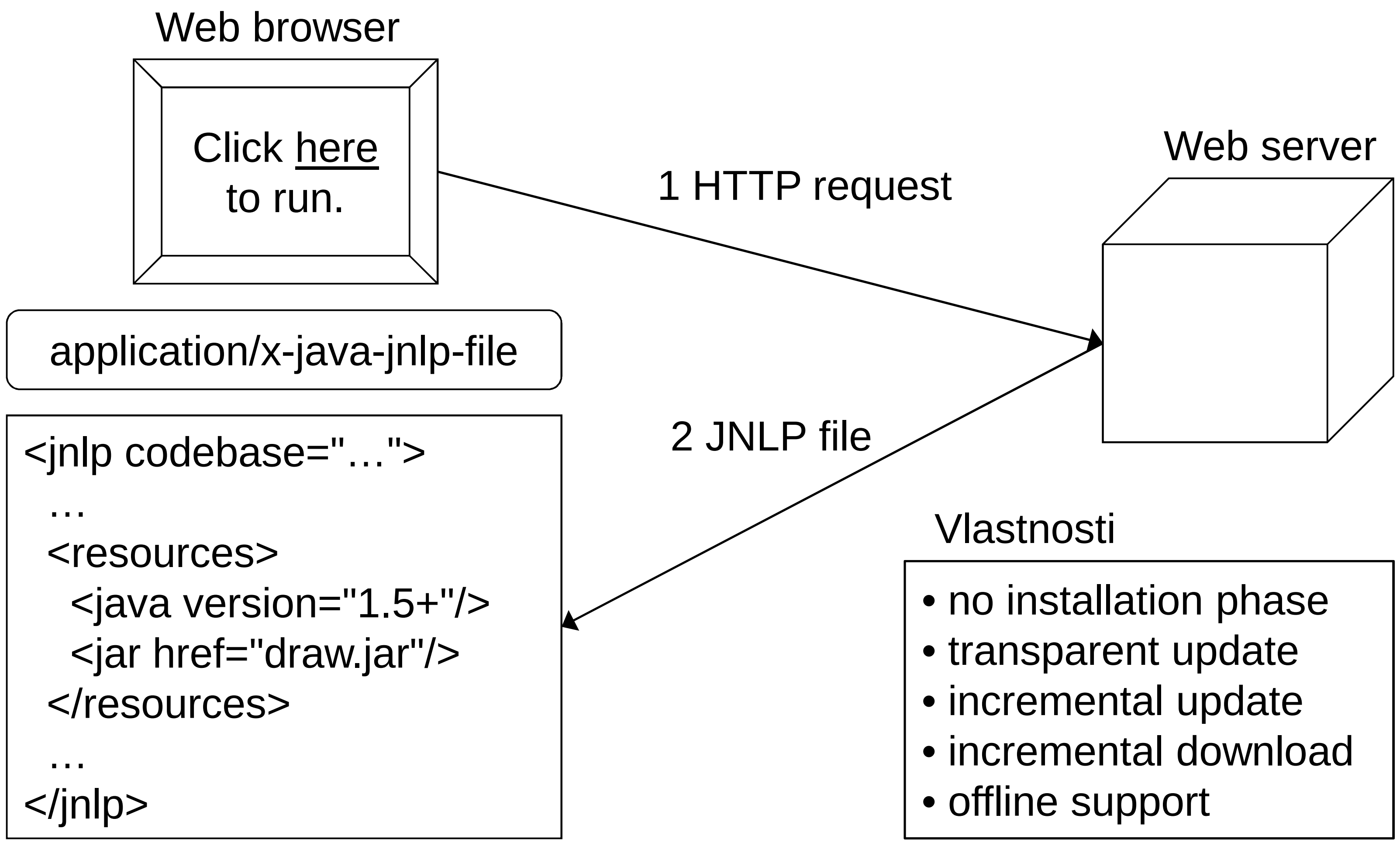
EndpointExceptionRedeliveryAttempts=1
EndpointExceptionRedeliveryInterval=500
SendUndeliverableMsgsToDMQ=true



# Hot Potato


msg
rollback
redelivery
rollback
redelivery
⋮
queue
consumer

# Java Web Start

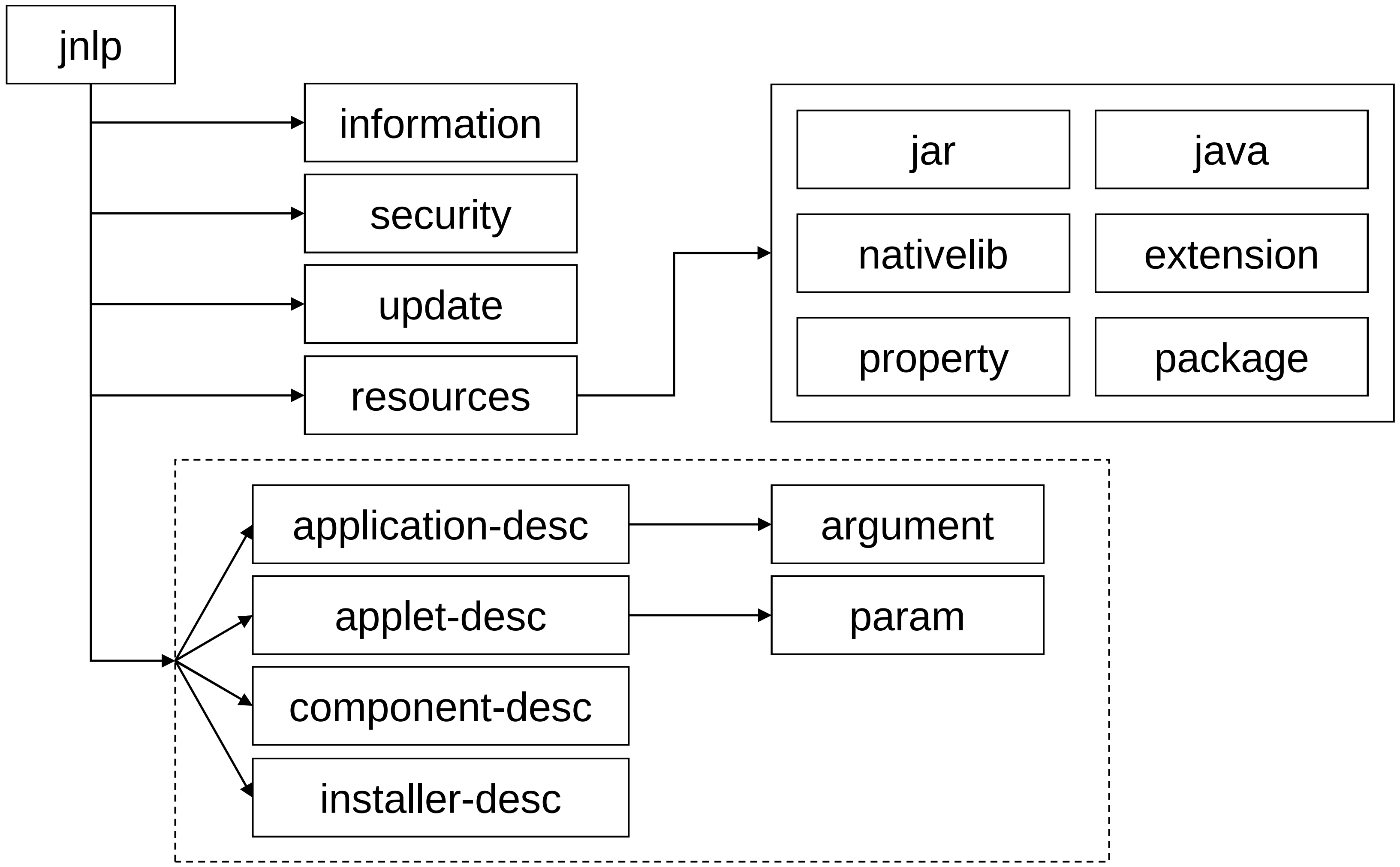


```
<jnlp codebase="…">
  …
  <resources>
    <java version="1.5+"/>
    <jar href="draw.jar"/>
  </resources>
  …
</jnlp>
```

Vlastnosti

- no installation phase
- transparent update
- incremental update
- incremental download
- offline support

# JNLP File


jnlp
information
security
update
resources
jar
java
nativelib
extension
property
package
application-desc
applet-desc
component-desc
installer-desc
argument
param

## Příklad

```
<?xml version="1.0" encoding="UTF-8"?>
<jnlp codebase="http://www.mysite.com/app">
  <information>
    <title>Draw!</title>
    <vendor>My Web Company</vendor>
    <description>Draw your dreams</description>
    <icon href="draw-icon.jpg"/>
    <offline-allowed/>
  </information>
  <information locale="cs">…</information>
  <resources>
    <java version="1.5+"/>
    <jar href="draw.jar"/>
  </resources>
  <application-desc main-class="com.mysite.Draw"/>
</jnlp>
```

# <security> & <update>

```
<security>
  <all-permissions/>
</security>
```

```
<security>
  <j2ee-application-client-permissions/>
</security>
```

always
timeout
background

always
prompt-update
prompt-run

```
<update check="timeout" policy="always"/>
```

## <resources>

```
<resources>
  <jar href="lib/app.jar" version="3.2" main="true"/>
</resources>
<resources os="Windows"/>
  <nativelib href="lib/windows/corelibs.jar"/>
</resources>
<resources os="SunOS" arch="SPARC">
  <nativelib href="lib/solaris/corelibs.jar"/>
</resources>
```

eager
lazy

```
<jar href="sound.jar" download="eager"/>
```

## Launch Sequence

1. retrieve JNLP file
2. parse JNLP file
3. determine (+ download and install) the JRE to use
4. download extension descriptors
5. install any required extension
6. download all eager JAR files
7. verify the signing and security requirements
8. setup JNLP services
9. launch the application/applet/installer

# Downloading Protocols

Basic

- jnlp
- icon
- jar
- nativelib
- extension

Version-based

- icon
- jar
- nativelib

incremental update

Extension

- extension
- java

Příklady: HTTP GET

```
http://www.mysite.com/c.jar
http://www.mysite.com/c.jar?version-id=2.3%2B
http://www.mysite.com/c.jar?version-id=2.3%2B&current-version-id=2.2
http://www.mysite.com/servlet/ext/coolaudio.jnlp?arch=x86&
os=Windows+XP&locale=en_US&version-id=2.3.0+2.3.1&
known-platforms=1.2
```

# JNLP API

- BasicService: getCodeBase(), isOffline(), showDocument(), …
- DownloadService: isResourceCached(), loadResource(), removeResource(),…
- FileOpenService
- FileSaveService
- ClipboardService
- PrintService
- PersistenceService: “cookies”
- ExtensionInstallerService
- SingleInstanceService
- ExtendedService

## Otázky & odpovědi

tronicek@fit.cvut.cz