

# Java WebStart

Marcin Werla

([marcin.werla@man.poznan.pl](mailto:marcin.werla@man.poznan.pl))

# Agenda

- What is Java WebStart
- JNLP benefits
- Configuration details
- JNLP API
- Summary

# Agenda

- What is Java WebStart
- JNLP benefits
- Configuration details
- JNLP API
- Summary

# What is Java WebStart?

- Helper application
- Comes with JRE
- Just a JNLP client
- So what is JNLP?

# JNLP is...

- Java Network Launching Protocol
- Something like Applets ...

...but better :-)

- In general it allows to „run“:

```
java -cp http://mycompany.com/jars/myApplication.jar  
com.mycompany.app.Main
```

# Agenda

- What is Java WebStart
- **JNLP benefits**
- Configuration details
- JNLP API
- Summary

# JNLP benefits

- No typical installation phase
  - No prompts about install directories etc.
  - Possible user desktop integration
  - Automatic JRE installation on demand
  - Installation activities are possible
  - Easy application removal

# JNLP benefits

- Transparent update
  - Automated checking for newer versions
  - Automated updates download

# JNLP benefits

- Incremental update
  - Depends on application structure
  - Only updated resources are downloaded
  - Incremental update for single jar files - basic and version-based downloads

# JNLP benefits

- Incremental download
  - Depends on application structure
  - Lazy/eager downloads
  - Code initiated downloads

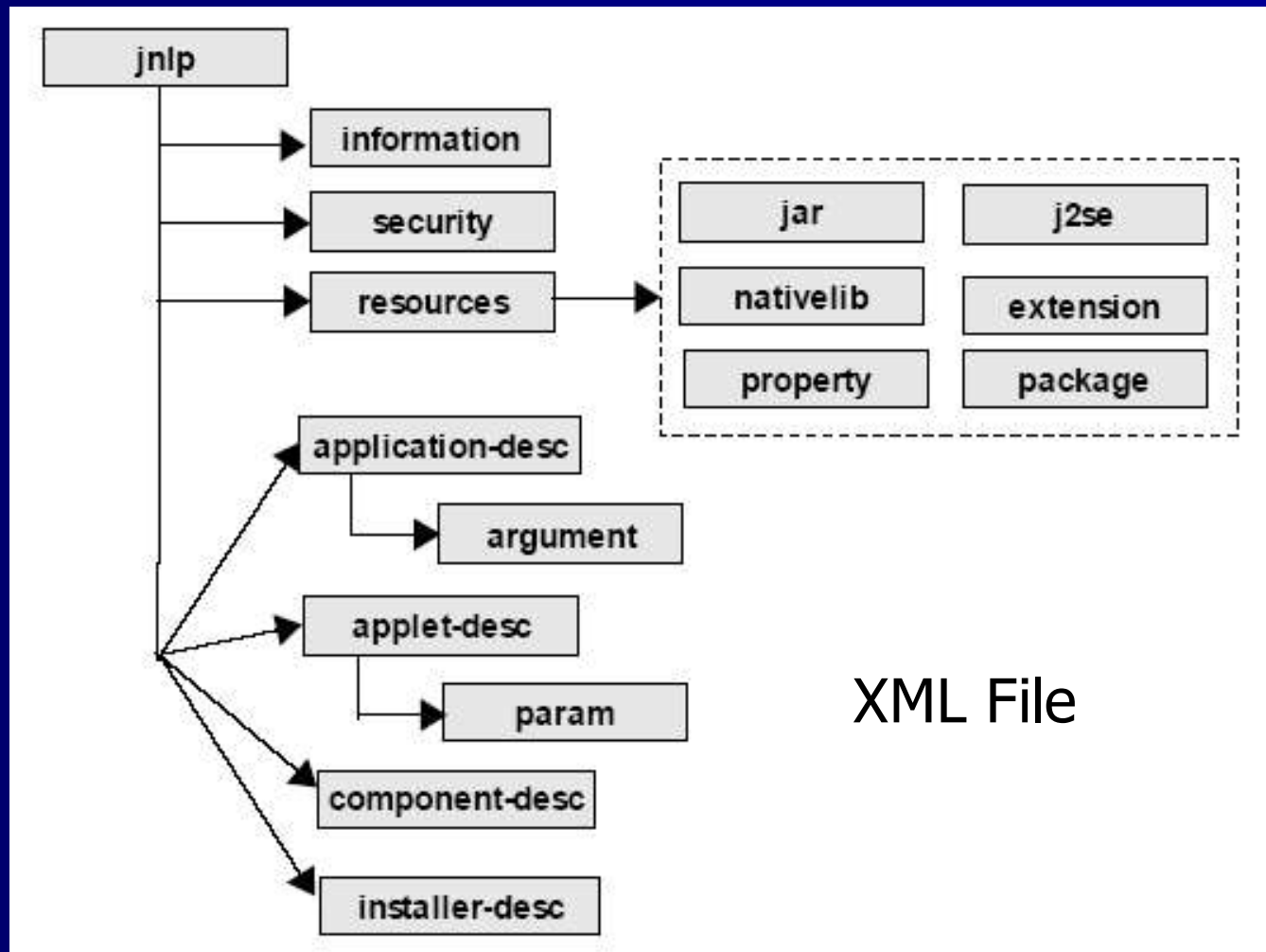
# JNLP benefits

- Offline support
  - JNLP applications can be started offline...
  - ...but it can be forbidden

# Agenda

- What is Java WebStart
- JNLP benefits
- **Configuration details**
- JNLP API
- Summary

# Application Descriptor



# <jnlp>

- Main file element
- Specifies protocol version
- Specifies base application location

# <information>

- Contains basic information:
  - <title>
  - <vendor>
  - <description>
    - kind="one-line"
    - kind="short"
    - kind="tooltip"
  - <icon>
  - <offline-allowed>

# <security>

- Decides about application permissions:
  - <all-permissions/>
  - <j2ee-application-client-permissions/>
- Resources must be signed to have extra permissions

# <resources>

- Describes application resources
- Can depend on:
  - Operating system
  - System architecture
  - Locale
- Can define additional system properties with <property>

# <j2se>

- Describes JRE configuration
  - Version (1.2, 1.2.2+, 1.2\*)
  - Initial heap size
  - Max heap size
  - Installation URL
  - Properties
  - Additional resources

# <jar>

- Describes jar file with:
  - Url
  - Version number
  - Part name
  - Download type (lazy/eager)
- One of jars can be selected as main application jar

# <nativelib>

- Describes native libraries (\*.dll, \*.so)
  - Url
  - Version number
  - Part name
  - Download type (lazy/eager)
- Libraries must be packed in jar
- Libraries must be in jar root directory
- Library access: `System.loadLibrary`

# <extension>

- Reference to resources described in external JNLP file
- Can define download type for resource subparts:

```
<extension name="Sound" version="1.0"  
  href="http://www.myext.com/servlet/ext/sound-extension.jnlp">  
  <ext-download ext-part="MIDI" />  
  <ext-download ext-part="MP3" download="lazy"  
    part="MP3Player" />  
</extension>
```

# <package>

- Shows which classes or packages are in which jar files
- Combination of „name“ and „recursive“ attributes:
  - name="com.mycompany.app.Main"
  - name="com.mycompany.app.\*"
  - name="com.mycompany.app.\*" recursive="true"

# <application-desc>

- Defines application main class
- Defines application execution arguments

# <applet-desc>

- Similar to HTML applet description
- Defines:
  - Main class
  - Document base
  - Name
  - Width, height
  - Execution parameters (name, value)

# <component-desc>

- Defines that jnlp file describes extension
- Does not contain any additional information

# <installer-desc>

- Defines installer main class
- Installer class is executed with:
  - „install“ parameter for installation
  - „uninstall“ parameter for unnistallation

# Agenda

- What is Java WebStart
- JNLP benefits
- Configuration details
- **JNLP API**
- Summary

# JNLP API

- Additional services accessible through `javax.jnlp` package
- Available to both trusted and untrusted applications
- Available through `ServiceManager.lookup`

# JNLP API Services

- **BasicService**
  - Querying and interacting with environment
  - getCodebase, isOffline, ...
- **DownloadService**
  - Allows application to control resource caching
  - isSthCached, loadSth, removeSth, ...

# JNLP API Services

- FileOpenService, FileSaveService
  - Allows loading and saving files
  - Wrapper class for file contents
  - Executed application knows only name of the file

# JNLP API Services

- ClipboardService
  - Gives clipboard access
  - `getContents`, `setContents`, ...
- PrintService
  - Allows printing
  - `showPageFormatDialog`, `print`, ...

# JNLP API Services

- PersistenceService
  - Allows storing data in local system
  - Cookie like mechanism
  - Allows sharing data between different applications from the same host
- ExtensionInstallerService
  - Used to manipulate progress screen
  - Gives access to informations about installed extensions

# Agenda

- What is Java WebStart
- JNLP benefits
- Configuration details
- JNLP API
- **Summary**

# Summary

- JNLP simplifies application distribution
- No code modification required
- Simple usage
- SDK with examples and deployment servlet
- Maven plugin :-)

# Links

- <http://java.sun.com/products/javawebstart/>

**Any questions?**

**Thank you :-)**