

# App Start

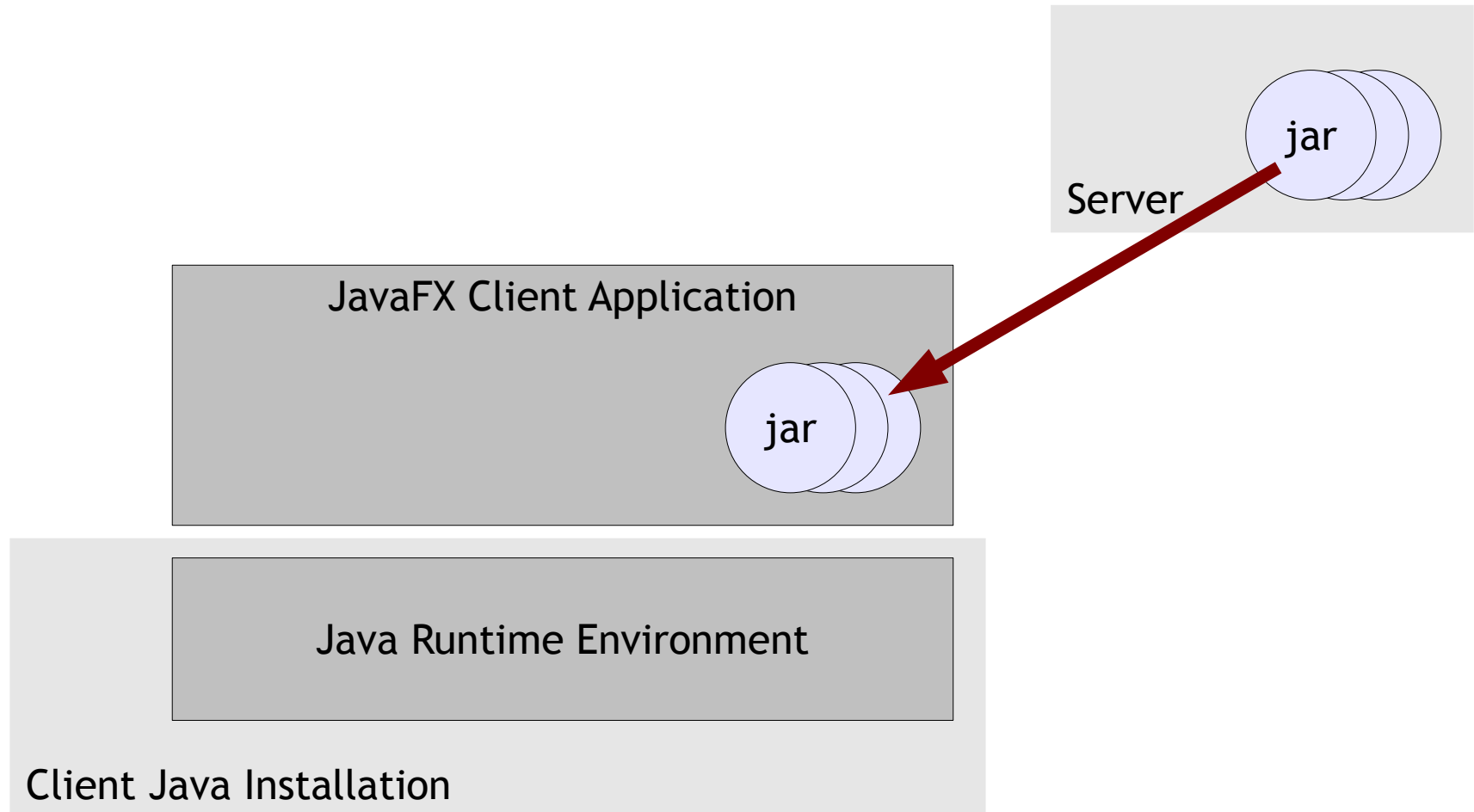
Launching JavaFX Applications



# Running a JavaFX Client Application



# Applet/Web Start



# Applet/Web Start

## + PRO

Automated Download of JavaFX Application  
(both initial load and continuous)

## - CON

Requires a client side Java Installation  
(security issues..., policy issues...)

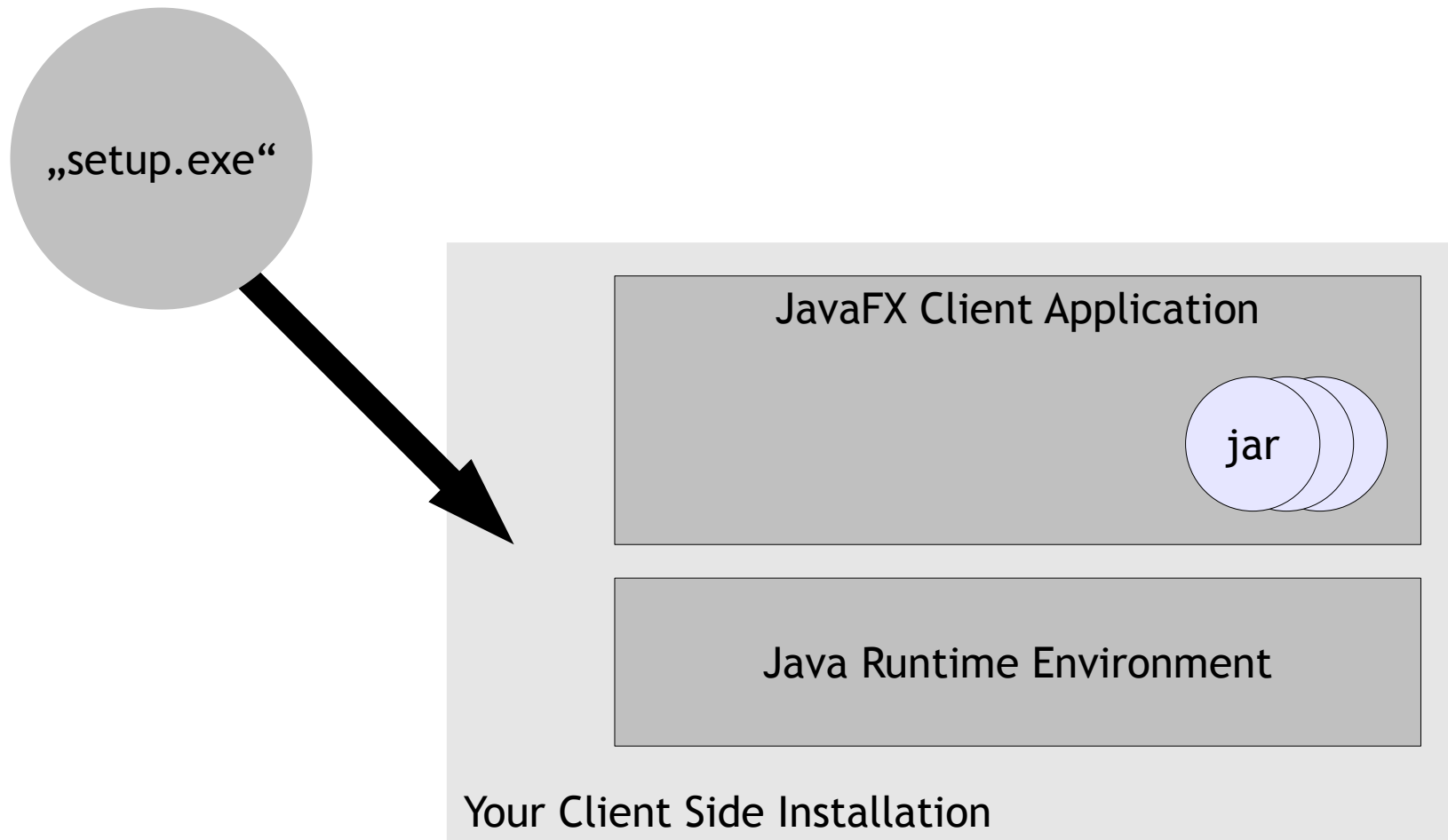
Rollout of a certain Java version is always a problem

Especially JavaFX applications require the user to  
keep up to date with the client side JRE

JRE version is driven by the user - not by the JavaFX  
application. Does your application run on JRE 8.xx?

Client Java Installation

# Application



# Application

„setup.exe“

+ PRO

No central JRE installation  
You bundle the JRE that you require, that you tested

- CON

No concept of how to continuously update your  
JavaFX Application

Java Runtime Environment

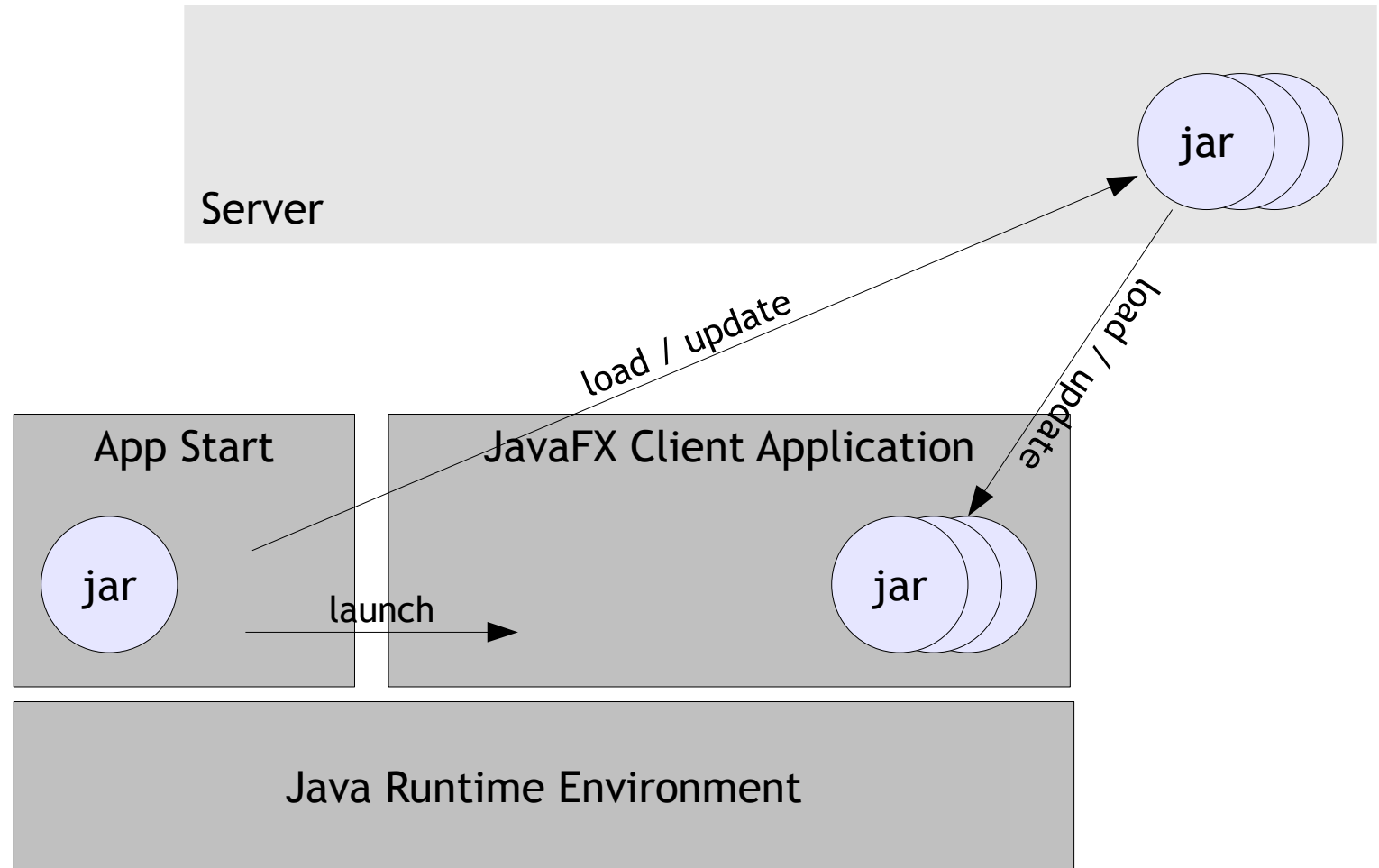
Your Client Side Installation

# Consequence

- Missing framework which runs JavaFX as stand alone application AND...
  - ...loading the JavaFX program from central servers
  - ...continuously checking the client side JavaFX program for server side updates
- **...“App Start“!**
  - ...combining the „pro“s of Webstart with the „pro“s of running JavaFX as stand alone application.

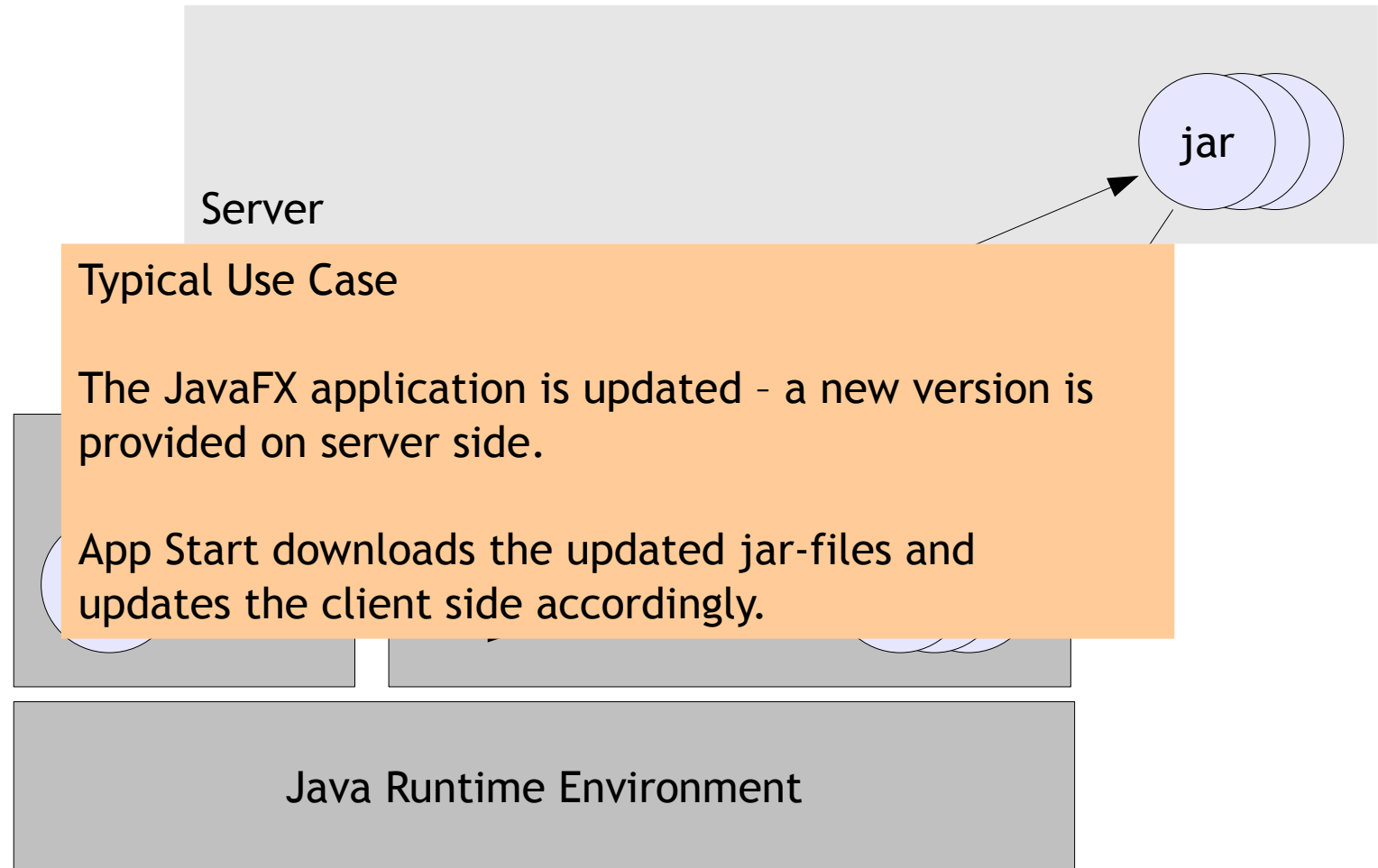


# App Start - Basics

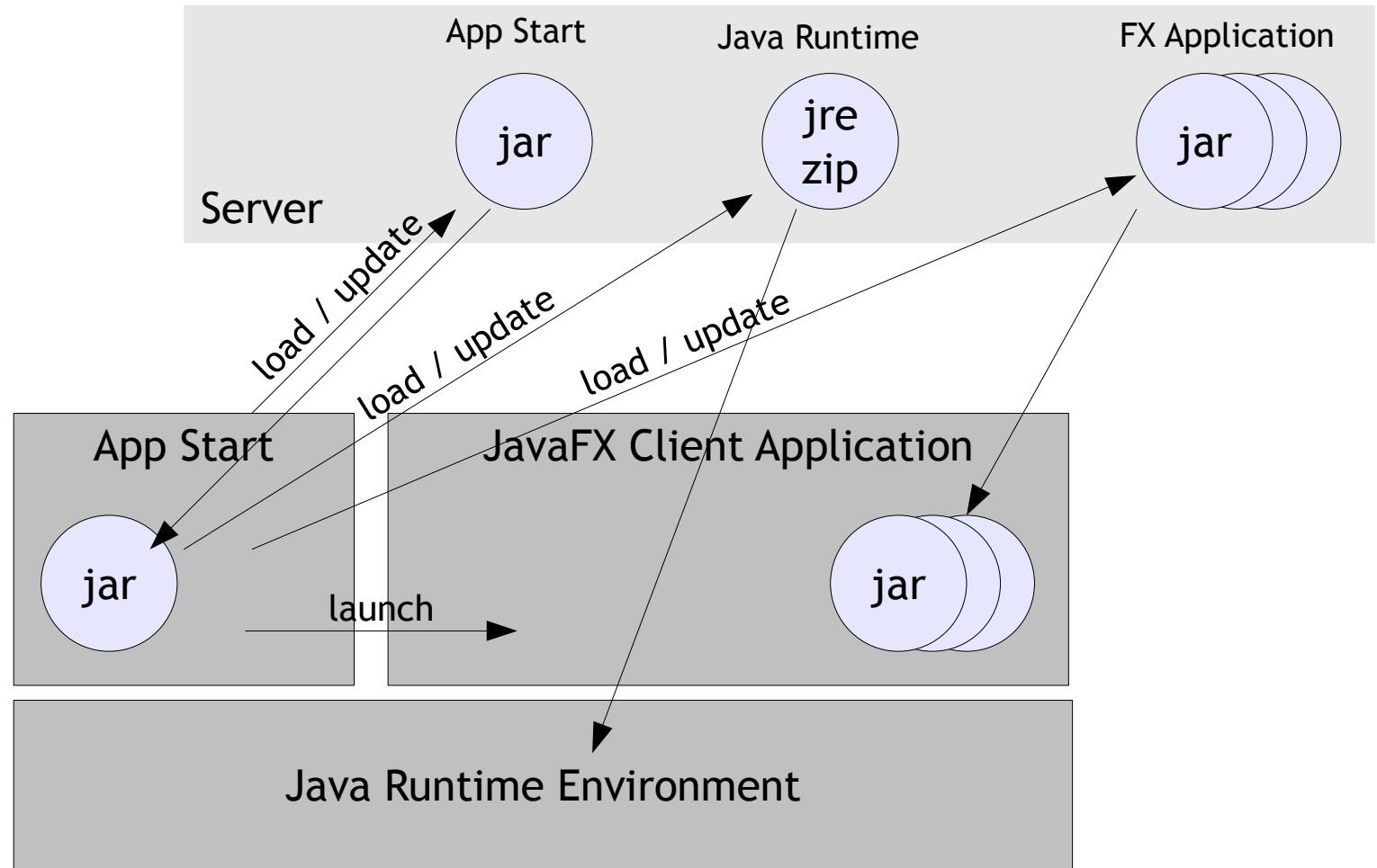




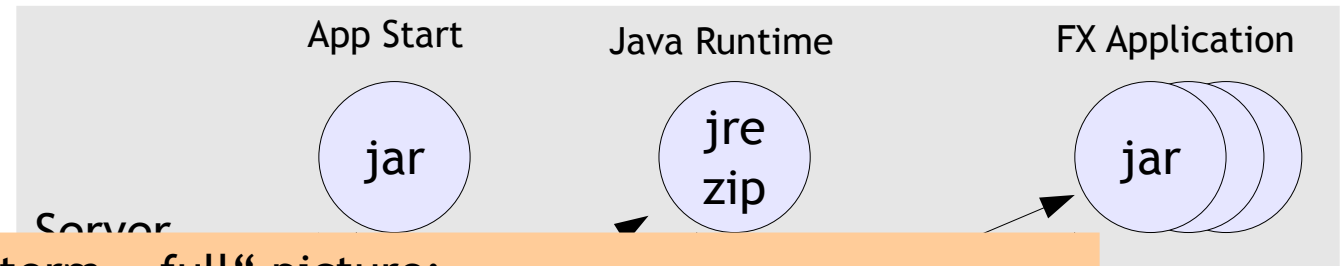
# App Start - Basics



# App Start - Full picture



# App Start - Full picture



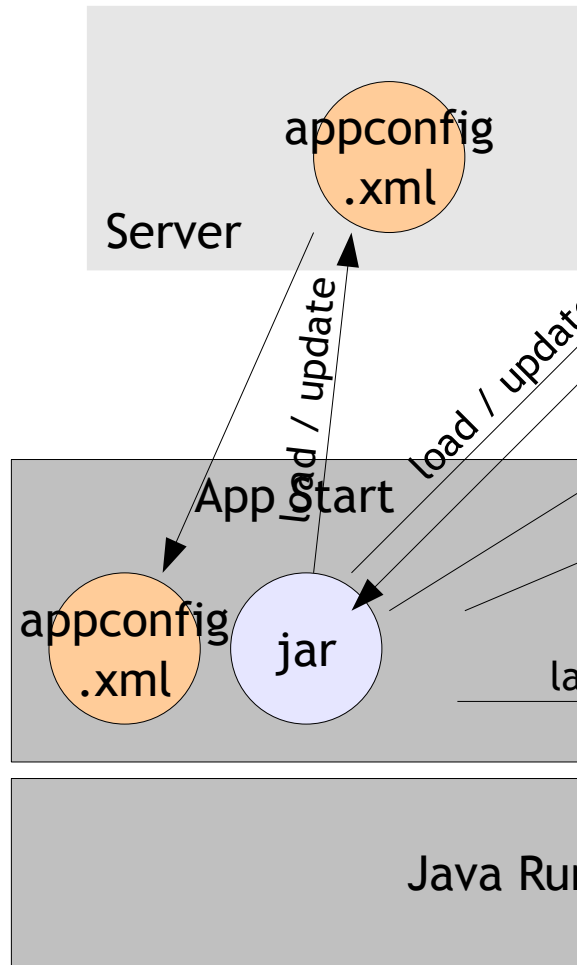
Long term, „full“ picture:

„Everything“ is to be updated during the life cycle of a JavaFX application:

- the application
- the Java Runtime Environment
- the App Start framework

So, everything is versioned and downloaded/updated correspondingly.

# Configuration



```
<app
  appname="CaptainCasa Enterprise Client"

  appversion="20140218_1"

  launcherversion="5_0_20140218"
  launcherurl="http://host/ecIntlaunchfx/launchfx_5_0_20140218.jar"

  javaversion="17_15"
  javaur="http://host/ecIntlaunchfx/jre_17_15.zip">

  <!-- Definition of application jars -->
  <jar jarurl="http://host/ecInt/libfx/app1.jar"/>
  <jar jarurl="http://host/ecInt/libfx/app2.jar"/>

  <!-- Definition of startables -->
  <start id="DW"
    name="Demo Workplace"
    classname="org.ecInt.fxclient.elements.PageBrowserStarter">
    <vmparam value="-Xmx256m"/>
    <vmparam value="-Xms128m"/>
    <param value="url=http://host/faces/workplace/workplaceFX.jsp"/>
    <param value="loglevel=INFO"/>
    <param value="undecorated=true"/>
  </start>

</app>
```

## Please note...

- The App Start framework currently is a „small“ one
  - ...comparing versions
  - ...loading resources via http(s)
  - ...managing resources on client side
  - ...creating scripts (if required) on client side
- It's not rocket science!
- There is room for development...
  - ...checking signatures
  - ...script generation for non-Windows scenarios
- We are open to share everything, please contact us via [info@CaptainCasa.com](mailto:info@CaptainCasa.com) in case of interest.