Fujitsu and Java

- One of the oldest Java related products vendor
  - Including Java for mainframe

- Interstage Application Server since 1998
  - Java EE certified GlassFish base application server

- Interstage Studio
  - Eclipse base
  - classic GUI library for client desktop
History of Client Java

Client & Server

Server

Client

Web

Server

Applet/Flash

Brower

Server

JSP/JSF

Browser

Server

HTML5・SPA

Browser

Reduce Server Load

Increase Client Maintenance Cost

Reduce Client Maintenance

Increase Security Risk

Reduce Security Risk

Poor GUI

Rich GUI

Improve pc performance

W3C HTML5 recomendation

Copyright 2019 FUJITSU LIMITED
Applet Technology

- Bundle with Netscape Navigator
- Java Applet runs on HotJava browser
- J2SE 1.2/Java Plugin
- WebStart

NPAPI: Netscape Plugin Application Programming Interface

Copyright 2019 FUJITSU LIMITED
Applet Technology

- 1995: J2SE 1.2/Java Plugin
- 1996: Bundle with Netscape Navigator
- 1998: Java Applet runs on HotJava browser
- 2001: WebStart
- 2013: US DHS recommendation (disable Plugin)
- 2015: Google Chrome disable NPAPI
- 2017: Edge quit NPAPI support
- FireFox quit NPAPI except Flash
- Oracle Java 9 deprecate Applet
- Oracle Java 6/7 remove Plugin

※NPAPI: Netscape Plugin Application Programming Interface
Applet Resources

How much are there Applet?

<table>
<thead>
<tr>
<th>Fujitsu Customers</th>
<th># of client pc</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Company</td>
<td>90,000</td>
</tr>
<tr>
<td>B Company</td>
<td>70,000</td>
</tr>
<tr>
<td>C Company</td>
<td>54,000</td>
</tr>
<tr>
<td>D Company</td>
<td>21,000</td>
</tr>
<tr>
<td>E Company</td>
<td>5,000</td>
</tr>
<tr>
<td>F Company</td>
<td>3,500</td>
</tr>
<tr>
<td>G Company</td>
<td>3,000</td>
</tr>
<tr>
<td>H Company</td>
<td>3,000</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>249,500</strong></td>
</tr>
</tbody>
</table>
Migration Plans

1. Use Applet
   until end of IE11

2. Use JSP/JSF
   Not so much used?

3. use HTML5/JavaScript
   favorite!?
Jakarta EE survey (used API)

- Java Persistence API (JPA): 76.23%
- Java API for Restful Web Services (JAX-RS): 75.73%
- Servlet: 59.39%
- Enterprise JavaBeans (EJB): 51.14%
- Java Transition API (JTA): 47.31%
- JavaServer Pages (JSP): 39.39%
- JavaServer Faces (JSF): 33.63%


JSP/JSF are not used so much
<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
<th>Website</th>
</tr>
</thead>
<tbody>
<tr>
<td>DukeScript</td>
<td>Java app runs on browser without Plugin</td>
<td><a href="https://dukescript.com/">https://dukescript.com/</a></td>
</tr>
<tr>
<td>jpro</td>
<td>JavaFX app runs on browser</td>
<td><a href="https://www.jpro.io">https://www.jpro.io</a></td>
</tr>
<tr>
<td>JSweet</td>
<td>Transpiler from Java to JavaScript</td>
<td><a href="http://www.jsweet.org/">http://www.jsweet.org/</a></td>
</tr>
</tbody>
</table>
Transpiler

Change source code in a language to same code in a different language

Difference between transpiler and compiler

COBOL source \(\rightarrow\) transpiler \(\rightarrow\) C source

Machine code \(\rightarrow\) compiler

Copyright 2019 FUJITSU LIMITED
Examples of transpiler

- **Legacy language**
  - Fortran → C (f2c)
  - C++ → C (cfront)

- **JavaScript (altJS)**
  - TypeScript → JavaScript
  - CoffeeScript → JavaScript

- **Legacy to JavaScript**
  - Java → JavaScript (GWT, JSweet)
  - Haskell → JavaScript (GHCJS)
Java→JavaScript transpiler

♦ pros
  • Need not know JavaScript
  • Easy to migration of legacy Java app
  • IDE support

♦ cons
  • Rapid frontend technology
  • API mismatch
JSweet

- Java → TypeScript(JavaScript) transpiler
  - http://jsweet.org

- Syntax Mapper
  - Only Syntax conversion
  - Use JavaScript API in Java code

- Resolve existing problems of transpiler

---

Application

JavaScript API(Candy)

JavaScript impl

Java source separation
Existing transpiler (API conversion)

Input Java source

Button button = new Button();
button.setLabel("click me");

transpile

Need not modify Java source

Output JavaScript

var button = document.createElement("button");
button.innerText = "click me";}
JSweet transpiler (Syntax Mapper)

Input Java source
Button button = new Button();
button.setLabel("click me");

Modify Java source

API conversion

Modified Java source
HTMLButtonElement button =
document.createElement(StringTypes.button);
button.innerText = "click me";

transpilie

Syntax Mapper

Output TypeScript
let button : HTMLButtonElement =
document.createElement("button")
button.innerText = "click me";
New problem

Need source modification before transpilation

Resolve by Candy

Application

Java source

Java source separation

Common Library (Candy)

JavaScript API (Candy)

JavaScript impl

No modification

Modified code

Good for systems using common library/framework
JSweet (practical way)

Input Java source

```java
Button button = new Button();
button.setLabel("click me");
```

Modify only library (API conversion)

```java
public class Button extends JButton {
    public void setLabel(String label) {
        this.label = label;
    }
}
```

Transpile

```java
Button button = new Button();
button.setLabel("click me");
```

Syntax Mapper
Candy (JSweet library)

- Bridge between Java and JavaScript
  - There are 1744 Candy registered (as of 2018/5)
    http://www.jsweet.org/candies-releases/
  - Can create your own

- Candy types
  - Java Candy: common Java library can be used by JSweet
  - JavaScript Candy: bridge to existing JavaScript

Application

Common Library(Candy)

JavaScript API(Candy)

JavaScript impl
Summary of system

- 22K LoC Applet
- Created more than 10 years ago
- Use Fujitsu GUI product

Requirements

- Migrate to JavaScript with less cost and time
- Must be same as before for Looks and operation
Fujitsu product for Client

provide button, label, frame and dialog for Java GUI application

Application (Java)

JBKGUI SpreadBean

JRE (AWT/coreAPI)

Java Virtual Machine

JBK Plugin

Web browser

Spread functions similar to Excel for Java application

Run applet on Web browser

*JBK: J Business Kit
Applet transformation

before

Applet(Java)

JBKGUI

SpreadBean

JRE(AWT/coreAPI)

JBK Plugin

Java Virtual Machine

Web browser

after

Application(JavaScript)

JBKGUI (JavaScript)

Spread (JavaScript)

Java base core API

JavaScript framework

AWT

spread library

jQuery

AngularJS ...

SlickGrid

Wijmo ...

Web browser

JavaScript engine
# Result

## LoC of two windows

<table>
<thead>
<tr>
<th></th>
<th>LoC</th>
<th>%auto</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>total</td>
<td>manual</td>
</tr>
<tr>
<td>A window</td>
<td>6,102</td>
<td>63</td>
</tr>
<tr>
<td>B window</td>
<td>4,764</td>
<td>44</td>
</tr>
<tr>
<td>common</td>
<td>11,789</td>
<td>849</td>
</tr>
<tr>
<td>total</td>
<td>22,655</td>
<td>956</td>
</tr>
</tbody>
</table>

manual: need to modify for transpiler

High auto transformation by Candy of Fujitsu product