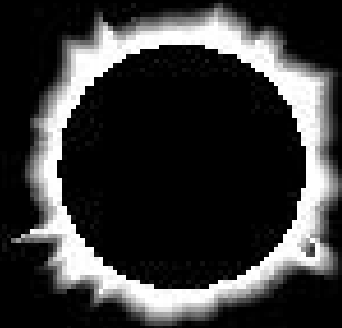


A Java Front-End To CGI/Perl



Gunther Birznieks
Selena Sol

gunther@clark.net

<http://www.extropia.com/Scripts/>

Overview

- Problems with CGI/Perl alone
- Problems with Java alone
- Solution: combine Java with CGI/Perl
- A look at some sample code
- An example CGI/Perl application enhanced by Java

Problems with CGI/Perl

- HTML Forms have a minimal interface
 - not interactive (JavaScript does not solve the whole problem)
 - cannot be extended natively
- Hard to maintain state between forms
- Redundant data frequently needs to be re-retrieved between HTML forms-- need to create fresh interface each time

Problems with Java Applets

- Bloated Applet Problem
 - Placing all application code on the applet makes for a huge download
- Java is hard!
 - Browser incompatibilities
 - Complicated maze of class libraries to learn

Problems with Java Applets

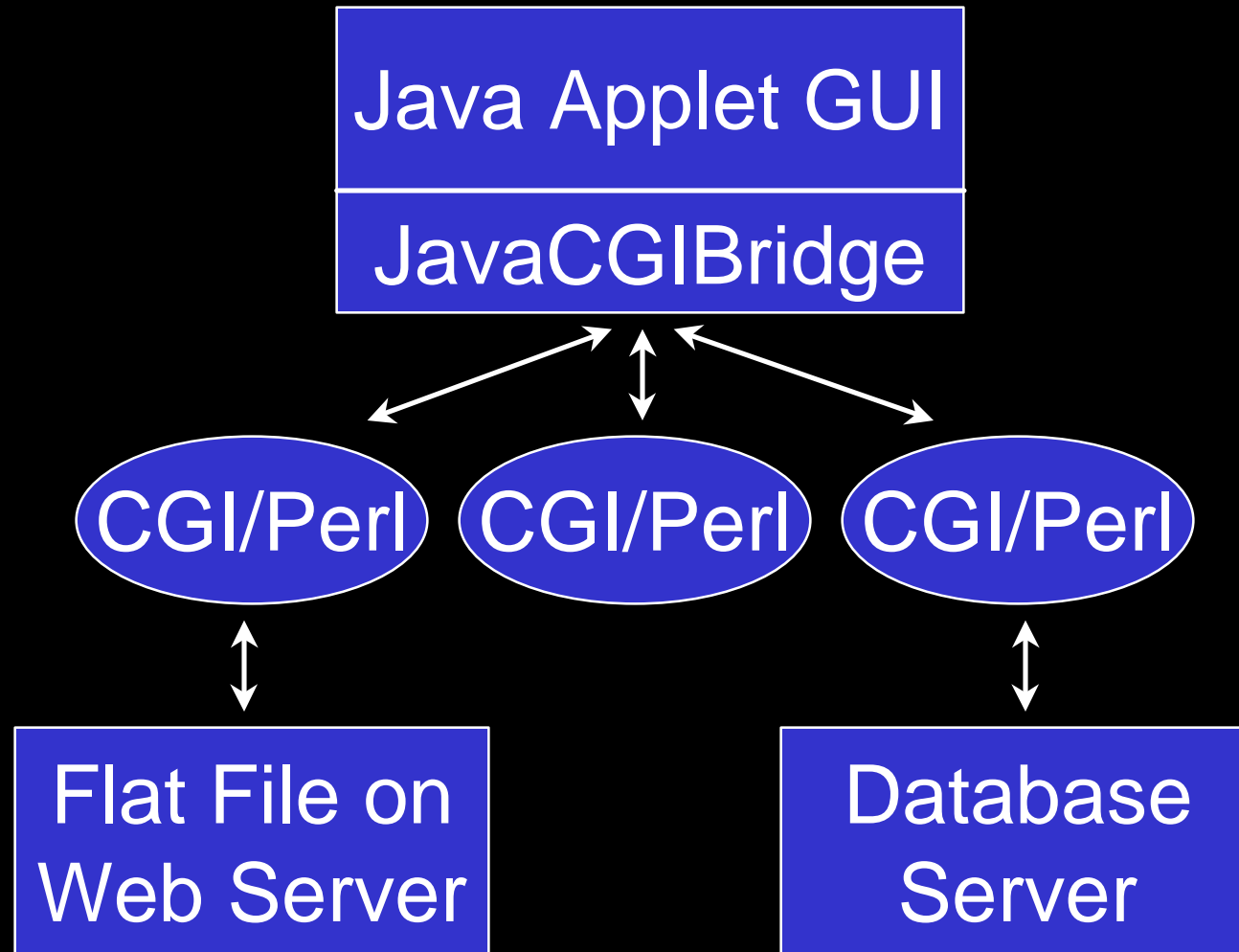
- Security “Sandbox” restricts what applets can do.
 - cannot read/write local files
 - cannot open sockets to non-originating server (RDBMS problem)

Problems with Java Applets

- No good, easy connectivity options
 - JDBC requires lax firewall for talking to DBMS directly or complex middleware
 - RMI is not supported by browsers
 - CORBA is not supported by all browsers

Let's look at the solution...

Java Front-End To CGI/Perl



JavaCGIBridge Solution

- CGI/Perl advantages
 - CGI/Perl can provide database connectivity more easily
 - Application related “business-rules” are centrally maintained and executed in CGI/Perl scripts

JavaCGIBridge Solution

- More CGI/Perl advantages
 - Browsers that don't support Java can still use the application
 - Leverage existing "legacy" CGI/Perl code
 - Many Internet Service Providers only allow CGI/Perl access (no daemons)

JavaCGIBridge Solution

- Java advantages
 - Java applet becomes a truly thin client -- only requires GUI code and GUI logic (JavaCGIBridge class adds ~5k overhead)
 - No need to learn the entire Java class library. You can get by with AWT (Forms) and some utility classes

JavaCGIBridge Solution

- More Java advantages
 - Java applet can maintain state between all CGI script calls
 - Java applet can cache data retrieved during previous CGI calls eliminating the need to constantly get redundant data from the server

But how do we actually use it...

Nuts And Bolts

- Modify CGI/Perl programs to return parsed data
- Add JavaCGIBridge class to Java applet implementation

Nuts And Bolts (CGI/Perl)

- Data to be returned should be identified
- Top separator (<!--start of data-->)
- Bottom Separator (<!--end of data-->)
- Pipe delimited fields
- (Optional) Set up form variable to tell script to send data using separators defined above (javacgibridge=on)

Sample CGI/Perl Code

- The code on the next slide shows how a CGI/Perl application would be modified to return sample data for the Java applet if the form variable “javacgibridge” is set to “on”.

Sample CGI/Perl Code (Continued)

```
# If javaccgibridge is on, send data specially formatted for
# java applet, otherwise send regular HTML data to the user's browser
If ($form_data{"javaccgibridge"} eq "on") {
    print "<!--start of data-->";

    # Loop through data here
    while (@row_of_data = &getRowOfData()) {
        printf("%s|%s\n", @row_of_data);
    }
    print "<!--end of data-->";
} else {
    print "<TABLE><TR><TH>Last Name</TH><TH>First Name</TH></TR>";

    # Loop through data here
    while (@row_of_data = &getRowOfData()) {
        printf("<TR><TD>%s</TD><TD>%s</TD></TR>\n", @row_of_data);
    }
    print "</TABLE>";
}
```

Nuts And Bolts (Java Applet)

- Instantiate JavaCGIBridge class
- Set up form variable, form value pairs inside Hashtable
- Set up URL by instantiating URL class
- Get parsed data back using the URL and form variable Hashtable as a Vector of Vectors (otherwise known as an array of arrays in Perl 5)

Sample Java Code

- The code on the next slide calls Address Book search script setting “firstname” and “javacgibridge” HTML form variables. The records will be returned as a Vector containing all the fields in each record as a Vector of Strings.

Sample Java Code (Continued)

```
URL u = new URL("http://www.yourdomain.com/cgi-bin/address_query.cgi");

JavaCGIBridge jcBridge = new JavaCGIBridge();

// firstNameTextField is a text field on a Java applet that the user fills
// in to query.
jcBridge.addFormValue(formVars, "firstname", firstNameTextField.getText());
jcBridge.addFormValue(formVars, "javacgibridge", "on");

Vector addressBookResults = jcBridge.getParsedData(u, formVars);
```

AddressBook Example

- Plain CGI/Perl version of the AddressBook
- Java Applet with CGI/Perl version of the AddressBook

Query Screen (Just CGI/Perl)



The screenshot shows a Netscape browser window titled "NHGRI ADDRESS LOOKUP - Netscape". The browser's menu bar includes "File", "Edit", "View", "Go", "Communicator", and "Help". The main content area displays the "NHGRI Address Lookup" page. Below the title, there is a section labeled "Enter Search criteria:" followed by several input fields: "First Name:", "Last Name:", "Phone:", "Group:", "Building:", "Room:", "E-mail:", and "Fax Number:". There are two checkboxes: "Perform Exact Match (Search is normally done as a case insensitive match based on a closest match criteria)" and "Return results in a condensed table format (desirable for long lists)". At the bottom of the form area, there are two buttons: "Do Search" and "Clear Criteria Fields". The browser's status bar at the bottom shows "Document: Done" and several system icons.

NHGRI ADDRESS LOOKUP - Netscape

File Edit View Go Communicator Help

NHGRI Address Lookup

Enter Search criteria:

First Name: Last Name:

Phone: [Group:](#)

Building: Room:

E-mail: Fax Number:

Perform Exact Match (Search is normally done as a case insensitive match based on a closest match criteria)

Return results in a condensed table format (desirable for long lists)

Document: Done

Query Results (Just CGI/Perl)

AddressList Query Results - Netscape

File Edit View Go Communicator Help

Last Name	First Name	Phone #	Group	Bldg	Room	E-Mail	Fax Number
Birznieks	Gunther	555-1212	NHGRI			gunther@nhgri.nih.gov	
Sol	Selena	555-1212	EFF			selena@eff.org	

[In-House Home Page](#) [Public Home Page](#)

Document: Done

Java with CGI/Perl

- All search data AND modification code on ONE applet!

Last Name	First Name	Phone #	Group	Bldg
Birnieks	Gunther	555-1212	NHGRI	
Sol	Selena	555-1212	EFF	

Search Add Modify Delete

First Name:

Last Name:

Phone #:

Group:

Building:

Room:

E-Mail:

Fax #:

Search The Database

Java Applet Window

More Information...

- Visit <http://www.extropia.com/Scripts/>
 - Selena Sol CGI/Perl Scripts Archive
 - JavaCGIBridge class examples and source code
- Acknowledgements
 - Joseph Ryan, Erik Ferlanti