Deployment Document

Wisconsin Hoofers Puddles System

The Puddles application is written to be compatible with standard J2EE deployment technologies and SQL databases. As such, it should be relatively straightforward to set up by anyone who has experience with other similarly architected programs. There is also a wealth of available information on the internet about such applications. A good starting point is the Sun J2EE documentation at: http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/ J2eeTutorialTOC.html

The steps for installing and deploying Puddles is as follows:

1.0 Overview of deliverables

For deployment purposes, the deliverables for the Puddles project consist of just two files.

The first file is called puddles.war. This file conforms to the Java Web Application Archive format, which is and contains all of the code and resources that reside on the application server in a single deployable package. Information about war files can be found at this URL: http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/WebComponents3.html

The second file is called create_tables.sql. This is a script that is designed to be run once during installation, and will populate a database with schemas and static data expected by the Puddles application.

2.0 Installation and configuration of the database software

A SQL complient database should be selected for storing all persistent data used by the system. The PostgresSQL database version 7.2 was used during the development and testing of this project, but it should be possible by design to substitute other standard databases, such as Oracle. Substituting a different database will likely require regeneration of classes produced by an object/relational mapping tool called sql2java. A script to regenerate these classes is provided with the source distribution of Puddles. A QA phase, especially checking for and fixing minor data format inconsistencies should also be performed during this migration.

Installation and administration of database software is outside of the scope of this document. Documentation for PostgresSQL can be found at: http://www.postgresql.org

Once the database software has been selected and installed, a database should be created to store all of the Puddles persistent data. This database should be populated with tables and static data by running the distributed sql script create_tables.sql. Since the database will eventually contain a lot of sensitive personal data about individuals, it is important to

locate this database on a secure network, where unauthorized connections not originating from the application server can not be made.

3.0 Installation of the J2EE Application Server

First, choose a suitable J2EE container that supports the Java Servlet API version 2.3 and JavaServerPager version 1.2. The application was developed and tested using Apache Tomcat 4.0, but due to the portable nature of J2EE, many other compatible servers can be used as well.

Installation and administration of application server software is outside of the scope of this document. Full documentation for Apache Tomcat can be found at this URL: http://jakarta.apache.org/tomcat/

4.0 Deployment of the Puddles application.

Once the application server software has been deployed, the remaining steps are to deploy the application in the application server, and configure the database connection.

The exact steps for deploying a WAR application into a J2EE server is application specific. In Tomcat 4.0, this is as simple as copying the puddles.war file into the directory called "webapps" in the Tomcat installation directory.

The final step is to configure the application to communicate with the database. This mechanism relies on Servlet parameters. Again, the details of how to set these are application-server specific, but the properties that need to be set are as follows:

jdbc.driver=org.postgresql.Driver

This first parameters specificies the Java class name of the JDBC driver used to communicate with the database. This particular driver is used to communicate with Postgres, and is included in the Puddles distribution. To use another database, an appropriate JDBC driver will need to be installed and added the web applications's classpath.

Addition parameters are as follows:

jdbc.url=jdbc:postgresql=//postgres.cs.wisc.edu:49173/puddles_database jdbc.user=username

These parameters specify the URL and username to use to connect to the database The correct values of these parameters are database specific and tell the JDBC driver how to connect to the database. Note that password authentication is not supported in this release, although it is possible to add this by changing the code.

In Apache Tomcat 4.0, these parameters can be specified by adding the following lines inside the <Host></Host> section of the conf/server.xml

```
<!-- Puddles Manager Context -->
<Context path="/puddles" docBase="puddles" debug="0" reloadable="true">
```

```
<Parameter name="jdbc.driver" value="org.postgresql.Driver" override="false" />
<Parameter name="jdbc.url" value="jdbc:postgresql://postgres.cs.wisc.edu:49173/lpkruger_706" over-
ride="false" />
<Parameter name="jdbc.user" value="lpkruger" override="false" />
<Logger className="org.apache.catalina.logger.FileLogger" prefix="puddles_log." suffix=".txt" times-
tamp="true"/>
</Context>
```

5.0 Setup SSL (optional)

For increased security, SSL can be used to encrypt communication between users browsers and the application server. There are two ways to configure SSL. The first way is through configuration settings on the application server. Refer to the documentation for your application server for more information on whether or not this is supported and the details of how to do it. The second way, which is application server independant, is to set up an external SSL gateway. An example of a software package which provides this functionality is stunnel, which can be found at this URL: http://www.stunnel.org/

6.0 Start and test.

The application server is now ready to be started. With Tomcat, this is performed with a script called startup.sh or startup.bat. Using a web browser, navigate to http://server-name:port/puddles/splash.jsp and begin testing the system.