PKUAS User Manual

Institute of Software

School of Electronics Engineering and Computer Science

Peking University

2006.03

License

Contents

List of Figures

Chapter 1. Installing and Configuring PKUAS

About

This chapter aims to be a guide for administrators and maintainers of PKUAS system, who will learn how to install, configure, run and maintain PKUAS. The readers will make sense of the entire application server and have a primary understanding of PKUAS through step by step introduction of hardware and software requirements, system installation and configuration. They will also learn how to configure services and to start or halt system. The documents information and files' structure of PKUAS before and after installation are listed in the end. Along with the upgrade of system, this user manual will be revised. Welcome to report the bug and additional materials with complementary information while using PKUAS.

1.1 Introduction

1.1.1 Background

PKUAS (PeKing University Application Server) is designed and implemented independently by Software Institute, School of Electronics Engineering and Computer Science of Peking University, sponsored by the National High-Tech Research and Development Plan of China (863)"Domain Oriented Component Operating Platform" and the Major Project of Science and Technology Research of Ministry of Education P.R.C."Domain Oriented Application Middleware".

1.1.2 Acronyms

AS	Application Server
API	Application Program Interface
CORBA	Common Object Request Broker Architecture

EJB	Enterprise Java Bean
IIOP	Internet Inter-ORB Protocol
JDBC	Java Database Connectivity
J2EE	Java 2 Platform Enterprise Edition
JAAS	Java Authentication and Authorization Service
JMX	Java Management Extensions
JNDI	Java Naming and Directory Interface
JRMP	Java Remote Method Protocol
JSP	Java Server Page
JPS	Java PetStore
JTA	Java Transaction Architecture
RMI	Remote Method Invocation
SOAP	Simple Object Access Protocol
SPI	Service Provider Interface
XML	Extensible Markup Language

1.2 Summarize

PKUAS is an open-structured component operating platform across Internet based on middleware technology and compliant with Java 2 Platform Enterprise Edition. It implements all the functionality of J2EE 1.3 and EJB 2.0. It provides containers for four standard types of EJB: stateless session bean, stateful session bean, entity bean and message-driven bean, as well as integrating Tomcat as its web container. It also provides various common services, such as naming, security, transaction, communication, logging, etc. All of them make up a perfective running environment for EJB and Web components. Besides, PKUAS supports RMI-IIOP interoperability protocol and enables accessing EJB via Remote/Local Interfaces.

1.3 Runtime Environments

1.3.1 Hardware Prerequisites

PKUAS should run on X86 hardware instruction set machines. Minimum requirements are: Pentium 166MHZ, more than 48M memory. At least 70M hard disk space is needed to install Java 2 SDK. More than 200M free space is recommended.

1.3.2 Software Prerequisites

PKUAS supports various operating systems, including Microsoft Windows 95,98(1st or 2nd Edition), Windows NT4.0（Service Pack 5）, Windows ME, Windows 2000 Professional/Server/Advanced Server, Windows XP，Windows Server 2003 and Unix/Linux/Solaris System. In addition, PKUAS needs Java 2 SDK 1.4.2 or above.

1.4 Installing and Building

PKUAS is an Open Source J2EE application server provided by SEI, EECS, PKU. The highly flexible and easy-to-use server architecture has made PKUAS the ideal choice for users just starting out with J2EE, as well as senior architects looking for a customizable middleware platform. The server binary and source code distributions are available from the ObjectWeb repository. (http://orientware.objectweb.org/). The ready availability of the source code allows you to debug the server, learn its inner workings and create customized versions for your personal or business use.

This chapter is a step-by-step tutorial that will show you how to install and configure PKUAS. Specifically, you will learn how to:

- Obtain binaries and source code from the PKUAS project site
- Install the binary
- Test the installation

You will also learn about:

- The installation directory structure
- Key configuration files an administrator may want to use to customize the PKUAS installation

1.4.1 Getting the Binary Files

1.4.1.1 Prerequisites

Before installing and running the server, check your system to make sure you have a working JDK 1.4 installation (1.4 only). The simplest way to do this is to execute the java -version command to ensure that the java executable is in your path, and that you are using Version 1.4. For example, running this command with a 1.4.2 JDK would produce version number like the following.

```
$ java -version
Java(TM) 2 Runtime Environment, Standard Edition
(build 1.4.2\_06-b03) Java HotSpot(TM) Client VM (build
1.4.2\_06-b03, mixed mode)
```

It does not matter where on your system you install PKUAS. Note, however, that installing PKUAS into a directory that has a name containing spaces causes problems in some situations with Sun-based VMs. This is caused by bugs with file URLs not correctly escaping the spaces in the resulting URL. There is no requirement for root access to run PKUAS on UNIX/Linux systems because none of the default ports are within the 0-1023 privileged port range.

1.4.1.2 Installing the Binary Package

After you have the binary archive you want to install, use the JDK jar tool (or any other zip extraction tool) to extract the archive contents into a location of your choice. The extraction process will create a pkuas2005 directory. The following section explores the contents of this directory.

Directory Structure

As mentioned above, installing the PKUAS distribution creates a directory which contains server start scripts, jars, server configuration sets and working directories. Figure 1.1 below illustrates the installation directory of the PKUAS server.

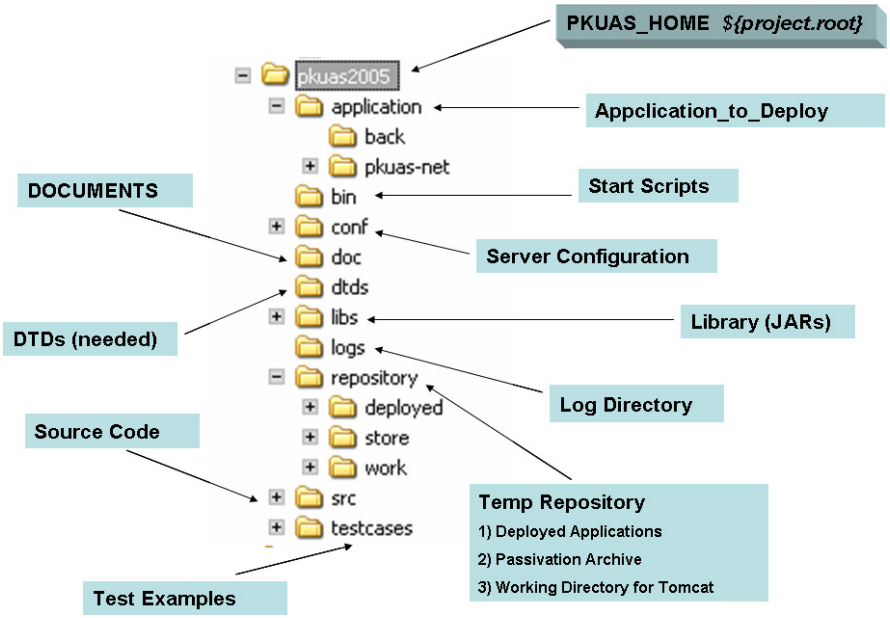

Figure 1.1 PKUAS server installation directory structure

Throughout the book we will refer to the top-level pkuas2005 directory as the PKUAS_HOME directory. In Figure 1.1, the default server configuration file set is shown expanded. It contains a number of subdirectories: application, bin, conf, doc, dtds, libs, logs, repository, src, and testcase. In a clean installation, logs and repository will not exist.

1.4.1.3 The Default Server Configuration File Set

All configuration files of PKUAS are located in the $PKUAS_HOME/conf directory. System Administrators should know their usage.

- jms directory
  Including all configuration files for PKUAS Java Message Service.
- pkuas.properties
  PKUAS server configuration file. Details are described in section 5.2.
- keystore
  Default certificate for https/SSL transport.
- auth.conf and j2se.conf
  Security authentication configuration files. PKUAS uses JAAS for authentica-

tion. The file ”auth.conf” includes only one default configuration item ”other” and this item defines ”pku.as.security.login.UsersRolesLoginModule” as the default authentication module. It is enough for most applications. If you need change authentication technology，just modify item ”other” or add a new item.

```
// The default server login module
other {
   pku.as.security.login.UsersRolesLoginModule required;
};
```

- users.db

  Database for user information.It is used to store users' name and password. If this file does not exist, system will create a new file which includes only two users: admin (password: admin123) and guest (password: any) when PKUAS starts. System Administrator could edit this database through the class pku.as.security.admin.SecurityAdminImpl. The file bin/sadmin.sh is the executable script on Unix.

- cmp.xml

  Define mapping between Java data type and database data type for CMP（Container Managed Persistence）.

- jrmp.policy

  Define security policy while using JRMP as IIOP.

- soapconfig.xml

  Define schemas for parsing and encapsulating SOAP message.

- Tomcat Configuration Files

  catalina.conf.xml，catalina.policy，server.xml，web.xml，tomcat-user.xml are all used to configure tomcat. Please consult the tomcat configuration documents for details.

- services.xml

  Define all the services of PKUAS.

1.4.2 Building the Server from Source Code

1.4.2.1 CVS Access

Anonymous CVS Access

This PKUAS project's CVS repository can be checked out through anonymous (pserver) CVS with the following instruction set. The module you wish to check out

must be specified as the modulename (if you don't know which module is defined or if no module is set, just remplace modulename by .). When prompted for a password for anonymous, simply press the Enter key.

```
cvs -d:pserver:anonymous@cvs.forge.objectweb.org:/cvsroot/pkuas
login
```

```
cvs -z3 -d:pserver:anonymous@cvs.forge.objectweb.org:/cvsroot/pkuas
co modulename
```

Developer CVS Access via SSH

Only project developers can access the CVS tree via SSH 2 (SSH 1 has been disabled for security reason). Substitute modulename and developername with the proper values (if you don't know which module is defined or if no module is set, just remplace modulename by .). Enter your site password when prompted.

```
export CVS\_RSH=ssh
```

```
cvs -z3 -d:ext:developername@cvs.forge.objectweb.org:/cvsroot/pkuas
co modulename
```

SSH Key Fingerprints

DSA Key: 1024 e9:ba:f0:e6:a5:b9:85:f7:48:21:48:51:15:27:74:8b RSA Key: 1024 4a:e0:24:de:dd:e3:5b:6c:eb:be:7c:8f:32:fc:72:bb

1.4.2.2 Build the Source Code

The CVS repository contains all of the modules required to build the server. To perform the build, cd to the %PKUAS_HOME%/src directory and run the compilation. Before running the compilation, you should hava install the Java build tool "Apache Ant" ahead, and set the ANT_HOME environment variable. You will need to set the JAVA_HOME environment variable to the location of the JDK you wish to use for compilation.

```
%PKUAS_HOME%/src $ ant
    to build the source code and generat the executable batch program
%PKUAS_HOME%/src $ ant clean
    to clean the mostly generated binaray file except the web-server module
%PKUAS_HOME%/src $ ant clean-all
    to clean all the generated binaray file
%PKUAS_HOME%/bin $ run.bat
    to run the PKUAS Server
```

1.4.3 Basic Installation Testing

1.4.3.1 Management Console

go to the directory %PKUAS_HOME%/src $ ant manage to generate the Management Tool Application: jsr77.ear

Start your PKUAS server and use one web browser to access web sites http://127.0.0.1:8080/manage

1.4.3.2 Hello Example

```
%PKUAS_HOME%/testcase/HelloExample $ ant
    to compile the source code and pack into an Enterprise
Archive and deploy it
%PKUAS_HOME%/testcase/HelloExample $ ant client
    to generate the client
%PKUAS_HOME%/testcase/HelloExample/client $ run.bat
    to run the test
Access the web site: http://127.0.0.1:8080/hello
    to run test through web server
```

1.4.3.3 Java PetStore

Go to the sub-directory ”src” and execute ”ant” in a command window to build the petstore.ear in this directory. You can also execute directly ”ant deploy” command to deploy generated petstore.ear into application directory.

If you have built the petstore.ear, you should follow five steps to run it on PKUAS:

1. Find the following settings in

```
<Service Class="pku.as.message.JmsService">
        <Attribute Name="connectionFactory" value="jms/ConnectionFactory"/>
        <Attribute Name="queueConnectionFactory"
            value="jms/QueueConnectionFactory"/>
        <Attribute Name="topicConnectionFactory"
            value="jms/TopicConnectionFactory"/>
        <Attribute Name="queueList" Value="jms/queue/myqueue "/>
        <Attribute Name="topicList" Value="jms/topic/mytopic"/>
</Service>
```

Change the red marked line into the following one:

```
<Attribute Name="queueList"
           Value="jms/queue/myqueue,jms/queue/AsyncSenderQueue,
                  jms/queue/opc/OrderQueue"/>
```

2. Copy one of the following two data source settings to your services.xml and modify it according to your real database environment for Petstore 1.3, and you must keep the data source name ”jdbc/CatalogDB” unchanged. We give two settings for oracle9i and mysql 5.0.

```
<!--for oracle 9i--> < Service Class="pku.as.datasvc.PoolManager"
Name="JdbcPool:name=jdbc/CatalogDB" >
    < Attribute Name="expirationTime" Value="300"/>
    < Attribute Name="minCapacity" Value ="5"/>
    < Attribute Name="user" Value ="jps"/>
    < Attribute Name="password" Value ="jps123"/>
    < Attribute Name="URL" 2
                Value ="jdbc:oracle:thin:@***.***.***.***:1521:***"/>
    < Attribute Name="driverClassName"
                Value ="oracle.jdbc.driver.OracleDriver" />
    < Attribute Name="maxCapacity" Value ="1000"/>
</SERVICE>

<!--for mysql 5.0--> <Service Class="pku.as.datasvc.PoolDataSource"
Name="JdbcPool:name=jdbc/CatalogDB" >
    <Attribute Name="expirationTime" Value="300"/>
    <Attribute Name="minCapacity" Value="5"/>
    <Attribute Name="maxCapacity" Value="100"/>
    <Attribute Name="URL"
               Value="jdbc:mysql://***. ***.***.***:3306/jps"/>
    <Attribute Name="user" Value="root"/>
    <Attribute Name="password" Value="****"/>
    <Attribute Name="driverClassName" Value="org.gjt.mm.mysql.Driver" />
</Service>
```

3. Please execute the right following SQL statements according to the real database for Petstore 1.3 before you run Petstore 1.3.

for oracle 9i:

create table CustomerDB(userID varchar(100));

create table CustomerAcc(userID varchar(100),pkAccountEJB number(20));

create table CustomerPro(userID varchar(100),pkProfileEJB number(20));

create table ProfileDB(pkProfileEJB number(20),prelang varchar(100),mylp number(1),fc varchar(100),bp number(1));

create table CreditCardDB(pkCreditCardEJB number(20),cardType varchar(100),expiryDate varchar(100),cardNumber varchar(100));

create table ContactInfoDB(pkContactInfoEJB number(20),fName varchar(100),tele varchar(100),gName varchar(100),email varchar(100));

create table ContactAdd(pkContactInfoEJB number(20),pkAddressEJB number(20));

create table AddressDB(pkAddressEJB number(20),zipCode varchar(100),state varchar(100),streetName2 varchar(100),streetName1 varchar(100),country varchar(100),city varchar(100));

create table AccountDB(pkAccountEJB number(20),status varchar(100));

create table AccCon(pkAccountEJB number(20),pkContactInfoEJB number(20));

create table AccCre(pkAccountEJB number(20),pkCreditCardEJB number(20));

create table UserDB(password varchar(100),userName varchar(100));

create table CounterDB(counter number(20),name varchar(100));

for mysql 5.0

create table CustomerDB(userID varchar(50));

create table CustomerAcc(userID varchar(50),pkAccountEJB INT);

create table CustomerPro(userID varchar(50),pkProfileEJB INT);

create table ProfileDB(pkProfileEJB INT,prelang varchar(50),mylp bool,fc varchar(50),bp bool);

create table CreditCardDB(pkCreditCardEJB INT,cardType varchar(50),expiryDate varchar(50),cardNumber varchar(50));

create table ContactInfoDB(pkContactInfoEJB INT,fName varchar(50),tele varchar(50),gName varchar(50),email varchar(50));

create table ContactAdd(pkContactInfoEJB INT,pkAddressEJB INT);

create table AddressDB(pkAddressEJB INT,zipCode varchar(50),state varchar(50),streetName2 varchar(50),streetName1 varchar(50),country varchar(50),city varchar(50));

create table AccountDB(pkAccountEJB INT,status varchar(50));

create table AccCon(pkAccountEJB INT,pkContactInfoEJB INT);

create table AccCre(pkAccountEJB INT,pkCreditCardEJB INT);

create table UserDB(password varchar(50),userName varchar(50));

create table CounterDB(counter INT,name varchar(50));

4. Copy the petstore.ear to

5. Open URL: http://your server's IP:8080/petstore to access the Pet Store. You must forcefully initiate the petstore's database by clicking the link "repopulate the demo's database" on this page before you enter the pet store for the first time.

1.5 PKUAS Configuration

1.5.1 System Configuration

After JDK and PKUAS have been installed, you should set the following system environment variables correctly:

- Set the environment variable JAVA_HOME to the JDK installation directory. (JDK1.4.2 is required.)
- Set the environment variable PKUAS_HOME to the PKUAS installation directory.

1.5.2 Properties Setup

PKUAS could configure its parameters through the property file "pkuas.properties" (located in the $PKUAS_HOME/conf directory):

PKUAS has two independent thread pools, which serve for Web Container and EJB Container separately. When Web Container works (JSP or Servlet), the thread pool for Web Container will be used; and When EJB Container works (EJB), the thread pool for EJB Container will be in use. When Web Container and EJB Container work together (JSP or Servlet calls EJB), only the thread pool for Web Container will be used.

The configuration of the thread pool for EJB Container includes the maximum size, the minimum size, and the initial number of threads in the Thread Pool:

- thread.pool.min：the minimum size of the thread pool, default value is 5
- thread.pool.max：the maximum size of the thread pool, default value is 10
- thread.pool.init：the initial number of thread in the thread pool, default value is 5

Besides properties referred above, the configuration of the thread pool for Web Container also includes the maximum length of the waiting queue：

- web.thread.pool.max=75
- web.thread.pool.min=5
- web.thread.pool.init=20
- waiting.max=20

The use of the waiting queue is that when the thread pool is full, client requests will be put in the waiting queue. If the waiting queue is also full, the server will refuse to receive any new request from client.

- pkuas.localhost：If the computer has more than one network card or IP address, you must configure this property to tell PKUAS which IP address it could use. Otherwise，PKUAS can not make sure which IP address it should use. But if the computer has only one network card or IP address, this is not necessary. Default value is localhost.
- communication.iiop.backlog：the server socket backlog, it specifies the number of connections to queue up before denying connections at the socket level.

- communication.iiop.buffersize：the buffer size of the Sockets that used to receive client request.
- communication.iiop.reconnectmax：the maximum number of reconnect attempts when sockets failed to get connected.
- communication.iiop.communicationmodel：the communication model between client and server. Two kinds of model is supported: IIOP and HTTP. In IIOP model, every JVM uses the same connection to communicate with server，that is all clients share this connection；In HTTP model，a new connection is created to receive each client request, and disconnected after the request finishes. Default model is IIOP.
- communication.soap.port：the port used to receive SOAP requests.
- communication.iiop.requestdispatcherbuffersize：the maximum number of client requests that PKUAS could cache.
- communication.iiop.retrytimes：the maximum number of retries after socket connection has been built and requests sending fails.
- communication.iiop.port.from and communication.iiop.port.to：PKUAS will automatically allocate a free communication port between the two properties for the application if the port is not specified by the application.

1.5.3 Services Configuration

1.5.3.1 Communication Services Configuration

Communication service is prerequisite for a distributed application server. The configuration of PKUAS communication service is as follows:

```
<SERVICE CLASS="pku.as.communication.CommService" > </SERVICE>
```

1.5.3.2 Log Services Configuration

Users can configure console logger and file logger respectively in pkuas.properties. Use ”pkuas.logger.consolelogger.enable=true” to enable console logger, and

”pkuas.logger.filelogger.enable=true” to enable file logger. Use filter element to specify the kinds of information you want to log. There are 4 kinds of information in PKUAS, ”info”, ”warn”, ”debug” and ”error”. For example: Snippet to configure console logger:

```
pkuas.logger.consolelogger.enable=true
pkuas.logger.consolelogger.filter=info,warn
```

If pkuas.logger.consolelogger.enable=false pkuas.logger.consolelogger.filter is ignored as is the case with file logger. Snippet to configure file logger:

```
pkuas.logger.filelogger.enable=true
pkuas.logger.filelogger.filter=info,warn,debug,error
pkuas.logger.filelogger.maxfilesize=0
pkuas.logger.filelogger.maxbackupindex=1
pkuas.logger.filelogger.logname=pkuas.log
```

PKUAS use rolling files for logging, so ”maxfilesize” specifies the maximum size that the output file is allowed to reach before being rolled over to other backup files. If maxfilesize¡=0, the size is unlimited. And ”maxbackupindex” specifies the number of rolling files to be used. Thevalue of the ”logname” element is the name-pattern of rolling files.

1.5.3.3 Naming Services Configuration

Administrators can set the port and entry factory class of naming service. For example:

```
<SERVICE    CLASS="pku.as.naming.NamingService">
    <ATTRIBUTE NAME="Port" VALUE="2000"/>
    <ATTRIBUTE NAME="ICFactory"
        VALUE="pku.as.naming.SmartCtxFactory"/>
</SERVICE>
```

In this configuration, the port is set to 2000 while the entry class is

pku.as.naming.SmartCtxFactory which PKUAS has provided.

1.5.3.4 Security Services Configuration

Administrator can configure the port of LoginServer and SecurityManager：

```
<SERVICE    CLASS="pku.as.security.SecurityService" >
    <ATTRIBUTE NAME="LSPort" VALUE="2020"/>
    <ATTRIBUTE NAME="SMPort" VALUE="2022"/>
</SERVICE>
```

1.5.3.5 Transaction Services Configuration

In the configuration of transaction service, the administrator can set the default timeout and the port of transaction proxy:

```
<SERVICE    CLASS="pku.as.transaction.TransactionService">
    <ATTRIBUTE NAME="DefaultTimeout" VALUE="6000"/>
    <ATTRIBUTe NAME="TAPort" VALUE="8082"/>
</SERVICE>
```

In this configuration, the port is set to 8082 while the default time out is set to 6000.

The “VALUE” amount of “Default Timeout” attribute defines the timeout of a transaction count in second. System will set the rollback mark after such a time. The “VALUE” amount of “TAPort” attribute defines the port to recerive remote transaction requests when do the distributed transaction service.

1.5.3.6 Java Message Services Configuration

In the message configuration paragraph, the administrator can set the message destinations for applications.There are two types of destinations: Queue and Topic.

```
<Service  CLASS="pku.as.message.JmsService" >
    <!--Attribute Name="JMSServer" Value="localhost:16010"/-->
    <Attribute Name="connectionFactory"
        Value="jms/ConnectionFactory"/>
    <Attribute Name="queueConnectionFactory"
        Value="jms/QueueConnectionFactory"/>
    <Attribute Name="topicConnectionFactory"
        Value="jms/TopicConnectionFactory"/>
    <Attribute Name="queueList"
        Value="jms/queue/myqueue,jms/queue/queue1"/>
    <Attribute Name="topicList"
        Value="jms/topic/mytopic,jms/topic/topic1"/>
</SERVICE >
```

If you have a standalone JMS provider running out of PKUAS,use the JMSServer attribute to specify the JMS provider's address and here you have to start and close the JMS provider manaully.Otherwise,left the attribute value empty or totally removed this attribute. And in this circumstance, PKUAS will control the JMS provider's lifecycle.

ConnectionFactory，TopicConnectionFactory，QueueConnectionFactory specify the names of the three types of connection factories respectively,all of which should only be used by standalone applications.As for applications in J2EE server environment, there are three other connection factories:JMSAgent_CF，JMSAgent_TCF，JMSAgent_QCF.In other words,if you want to access the JMS provider within J2EE server environment,you should use the latter three factories as substitutes.

The QueueList attribute specifies the names of queues which will be created when server startups; and the TopicList attribute is for topics to be created.Two names are separated by a comma.Each name will be registered in the PKUAS' JNDI system.

1.5.3.7 Data Services Configuration

Data service provides database coonection pool service. Now we give you a concrete configuration to explain the meaning for every attribute:

```
<Service Class="pku.as.datasvc.PoolManager">
    <SERVICE CLASS="pku.as.datasvc.PoolDataSource"
        NAME="JdbcPool:name=jdbc/asdemo" >
        <ATTRIBUTE NAME="ExpirationTime" VALUE="300"/>
        <ATTRIBUTE NAME="MinCapacity" VALUE="5"/>
        <ATTRIBUTE NAME="MaxCapacity" VALUE="30"/>
        <ATTRIBUTE NAME="URL"
            VALUE="jdbc:mysql://192.168.1.66/pkuas"/>
        <ATTRIBUTE NAME="User" VALUE="ecperf"/>
        <ATTRIBUTE NAME="Password" VALUE="ecperf"/>
        <ATTRIBUTE NAME="DriverClassName"
            VALUE="org.gjt.mm.mysql.Driver" />
    </SERVICE>
</SERVICE>
```

In the above configuration, “Expiration Time” sets up the timeout of a concrete connection instance in connection pool, a daemon will check the time mark of the instance in connection pool periodically, close the physical connection of the instance which is timeout and delete the instance in connection pool; “MinCapacity” is the minium number of the instances kept in connection pool; “MaxCapacity” is the maxium number of the instances kept in connection pool; “URL” is the string address for the JDBC; “DiverClassName” is the exact class name of the JDBC Driver; jdbc/asdemo in Name=” JdbcPol:name=jdbc/asdemo” is the JNDI name of the data source, users can get the data source by lookup. For instance:

javax.sql.DataSource ds =
(javax.sql.DataSource)initCtx.lookup(”java:comp/env/jdbc/asdemo”);

1.5.3.8 Web Container Configuration

PKUAS facilitates a Web container integration framework to integrate third-party Servlet engines. Currently Tomcat version 4.1.31 is integrated.

The startup parameters for the Web Container is set in the services.xml configuration file. The following XML code is a typical Web Container configuration using Tomcat as the Servlet engine.

```
<Service Class="pku.as.web.tomcat4.Server">
    <Attribute Name="debug" Value="5"/>
    <Attribute Name="httpPort" Value="8080"/>
    <Attribute Name="httpsPort" Value="8043"/>
    <Attribute Name="defaultDeployer" Value="localhost"/>
    <Deployer Name="localhost"
        Class="pku.as.web.tomcat4.HostWrapper">
        <Attribute Name="appBase" Value="application"/>
        <Attribute Name="autoDeploy" Value="true"/>
        <Attribute Name="unpackWARs" Value="true"/>
    </Deployer>
</Service>
```

The `Service` tag defines a PKUAS service component instance, the implementing class of which is specified in the `Class` attribute. Optionally you can give this instance a name using the `Name` attribute, but this is not necessary if you have only one instance of a service. The `Attribute` tags define additional attributes for this Web Container instance. All the configurable attributes of a Tomcat Engine are supported.

The `Deployer` tag defines a 'Deployer' for this instance of Web Container. For a Tomcat based implementation, this corresponds to a `Host` in the Tomcat component hierarchy. All configurable attributes of a Tomcat `Host` component are supported.

For more information on available Tomcat options, please refer to corresponding Tomcat documentations.

1.5.3.9 User Defined Services Configuration

You can start your customized services when PKUAS starting the common services. Here begins the configuration example:

```
<SERVICE  CLASS="pku.as.usersvc.UserService" >
    <ATTRIBUTE NAME="ClassName"
        VALUE="pku.as.usersvc.TestService" />
    <ATTRIBUTE NAME="MethodName" VALUE="test" />
    <ATTRIBUTE NAME="InNewThread" VALUE="FALSE" />
</SERVICE>
```

You can create a user-defined service by adding XML-format description.The property 'ClassName' defines the name of the class to be loaded, 'MethodName' defines the target method to be invoked, and the 'InNewThread' designates if this services runs in a new thread.

1.5.3.10 JAVAMail Service

To use JavaMail you should add the following configuration:

```
<SERVICE  CLASS="pku.as.mail.JavaMailService">
<ATTRIBUTE NAME="JNDIName" VALUE="Mail" />
<ATTRIBUTE NAME="Password" VALUE="password" />
<ATTRIBUTE NAME="Configuration">
    <configuration>
        <property name="mail.store.protocol" value="pop3" />
        <property name="mail.transport.protocol" value="smtp" />
        <property name="mail.user" value="nobody" />
        <property name="mail.pop3.host" value="pop3host" />
        <property name="mail.smtp.host" value="smtphost" />
        <property name="mail.from" value="mailfrom" />
        <property name="mail.debug" value="false" />
    </configuration>
</ATTRIBUTE>
</SERVICE>
```

The "JNDIName" attribute is the JNDI name when users lookup something while the "Password" attribute is used for validation. In the configuration tag

- the default value of mail.store.protocol is the message store protocal

- the value of mail.transport.protocol is the message transfer protocal
- the value of mail.user is the user name when connecting to the mail server
- the value of mail.pop3.host is the host of the pop3 mail server
- the value of mail.smtp.host is the host of smtp server
- the value of mail.from is the return mail address of current user
- the value of mail.debug is the initial debug model, true means open debug model, false means close it

1.6 PKUAS Boot and Stop

1.6.1 Boot

Execute the boot.bat under the PKUAS_DIST/bin directory to start PKUAS server. Then all the applications under the PKUAS_DIST/application directory will be deployed and started.

1.6.2 Stop

Close the console window or execute the shutdown.bat to halt PKUAS. PKUAS server will stop all the application and the started services, and kill itself finally.

Chapter 2. PKUAS web based management tool

2.1 About

This part will get you started with PMC(Pkuas Management Console).In this part,you will learn how to monitor and configure the applications deployed on PKUAS and services provided by PKUAS with PMC. At present,PMC only has radical ability to monitor and configure.We'll perfect it in the following release.Welcome to submit Bug report and suggestions to us.

2.2 Management Framework

The newest PMC framework accords with JSR77(Java Specification Request 77),JSR77 is J2EE management specification which based on the resources-management ability of JMX.This specification abstracts mutual managable information from J2EE architecture,and use metadata object (they are also called managed object)model to describe these information.And based on this,a well-defined management model is provided to monitor and configure applications and resources on J2EE platform and also the platform itself.JSR77 defines a standard framework of management and so make management independent form any application server. In JSR77 ,it stipulates Management EJB as external operation interface of JSR77 Management Model,and mapping ralation to Common Information Model(CIM) and SNMP MIB.Management EJB and Managed Object Model are obligatory in JSR77,Event Model,State Model and Statistic Model are optional. JSR77 Management Framework: 2.1：

2.3 PMC Bootstrap

PMC is deployed on PKUAS as an application(jsr77.ear),and will start when PKUAS starts;Users can use it with the URL:http://localhost:8080/manage. Figure 2.2 is the login page of PMC:

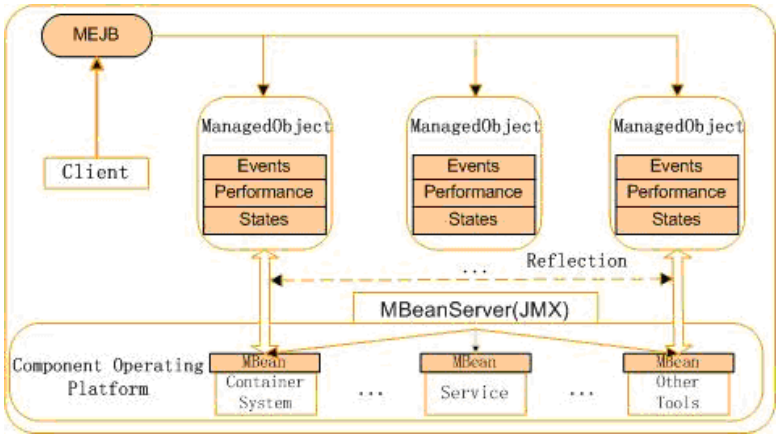

Figure 2.1 Management Framework

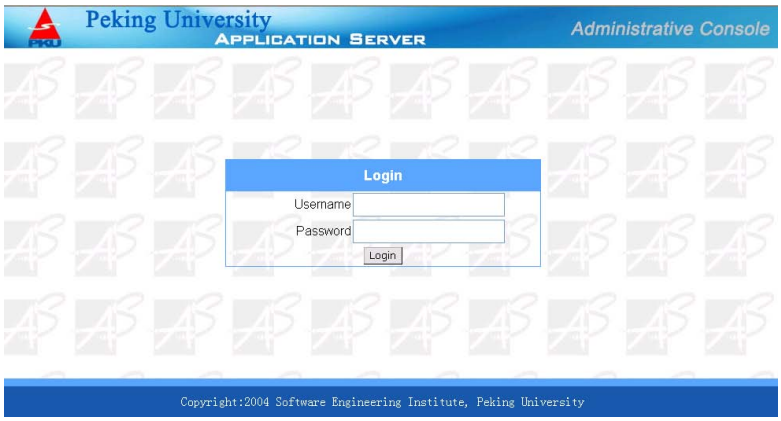

Figure 2.2 Login Interface

At present,PMC has only one administrator account (username : admin & password : admin123).Only administrator can add or delete accounts of PKUAS(physical),and other accounts just has common privilege.This is the exclusive defference between administrator and other accounts (common users). Figure 2.3 the is main Interface of PMC:

In Main Interface of PMC,there is a tree menu on the left.It shows 3 parts to manage:

1.server:configure localhost,thread pool attributes here

2.applications:deploy redeploy and undeploy applications.

3.services:manage all services(JMS,JTS,Javamail etc.)Click the service's name,and it will show information about that service,and you can modify these information if supported.

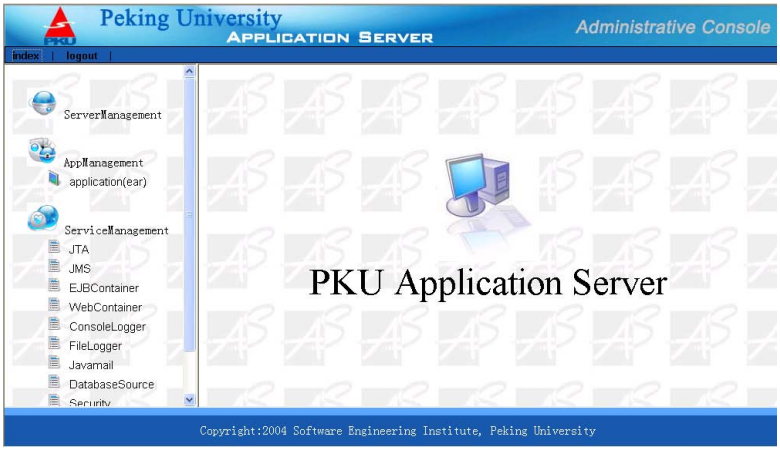

Figure 2.3 Main Interface

2.4 Manage Applications with PMC

2.4.1 Deploy/Undeploy Applications with PMC

PKUAS supports hot-deploy.Place application module(*.ear,*.jar,*.war) in the deploy directory（defaut is PKUAS_HOME/application）and PKUAS will deploy it when restarting.(If auto-deploy is on,need not restart).In application management page,you can deploy *.ear without restarting.

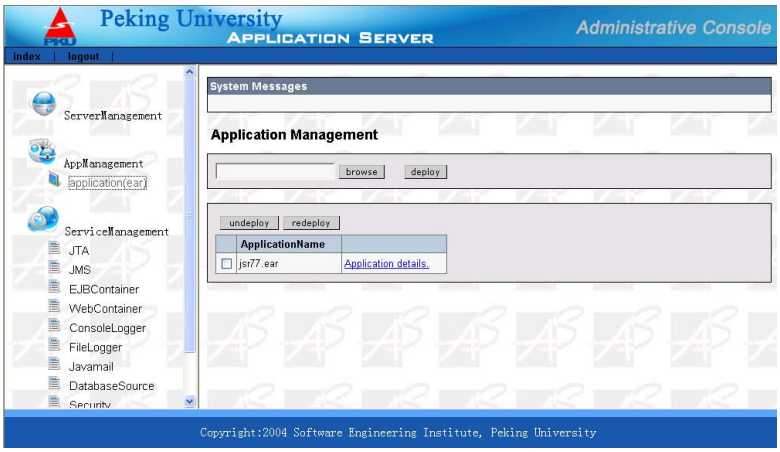

Figure 2.4 Application Management Page

Check the check-box before application's name,and click 'redeploy' or 'undeploy' button so that these applications will be redeployed or undeployed.Click 'browse' but-

ton ,choose your application on local,and click 'deploy' button so that it will be transferred to server and then deployed.

2.5 Browse Services Information

2.5.1 Browse Services Runtime Information

Click the service listed in the tree menu,and details of this service will be showed in the main page on the right.All these services listed in the tree menu can be monitored and managed.

2.5.1.1 Message Service

The following figure 2.5 shows JMS management page.In this page,all destinations in PKUAS are listed by 2 category :Queue and Topic.And users can also add or delete Queue or Topic destinations in this page.

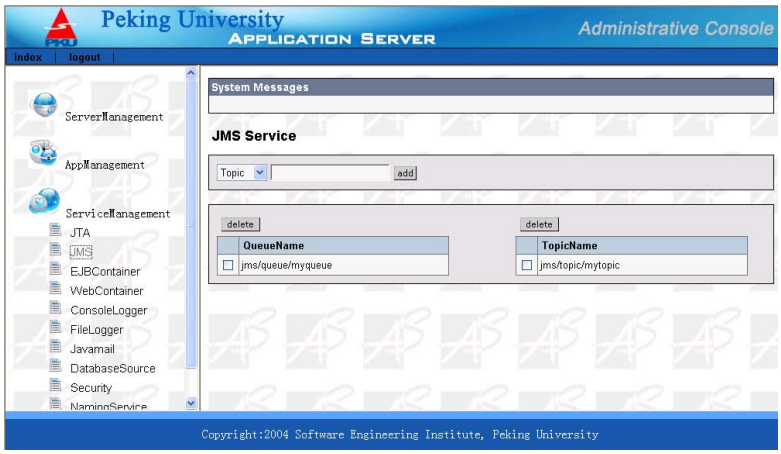

Figure 2.5 Message Service

2.5.1.2 Transaction Service

In JTA management page 2.6 ,DefaultTimeOut and TAPort can be managed.

2.5.1.3 Security Service（User Management）

In security management page2.7,administrastor can manage all accounts in PKUAS.(add,delete,update password).

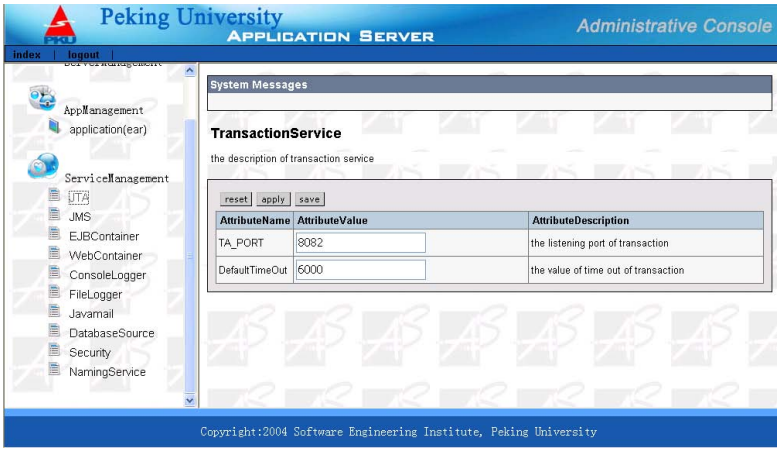

Figure 2.6 Transaction Service

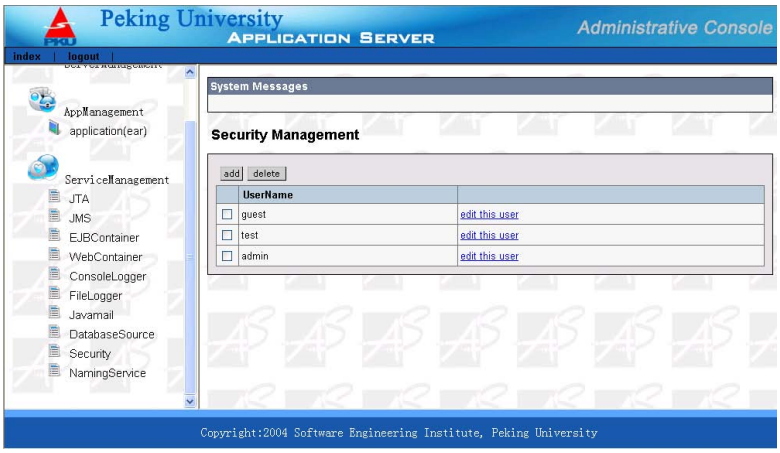

Figure 2.7 Security Service

2.5.1.4 Naming Service

Naming service maintains the bindings between names and addresses of resources in PKUAS.In NamingService management page2.8,PMC shows which class Naming Service uses ,and types and values of binding names.

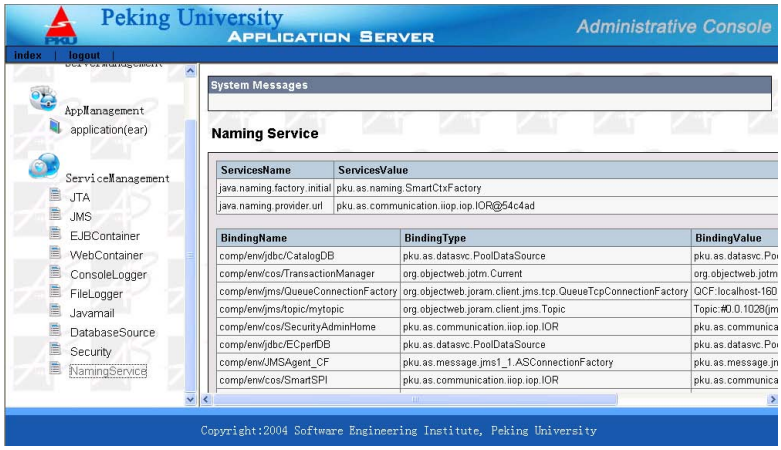

Figure 2.8 Naming Service

2.5.1.5 Web Container

In WebContainer management page2.9,PMC shows runtime information of web container,including http-port,https-port and so on. All these attributes can be modified.

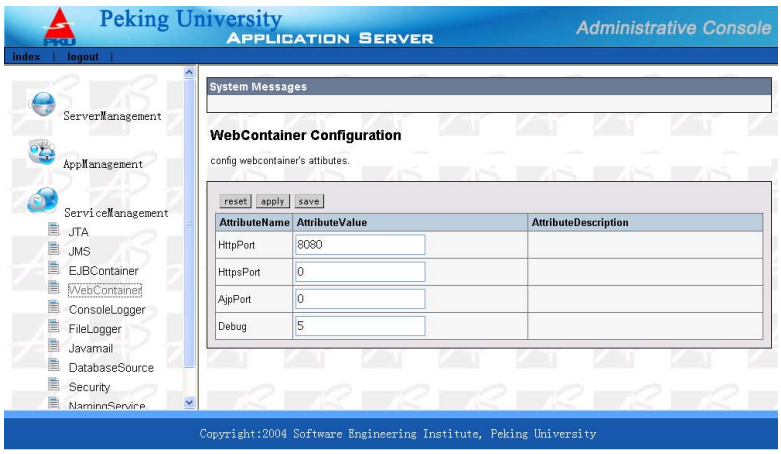

Figure 2.9 Web Container

2.5.1.6 EJB Container

In EJBContainer management page2.10,PMC shows runtime information of EJB container,including BeanInterval,TaskInterval and so on. All these attributes can be

modified.

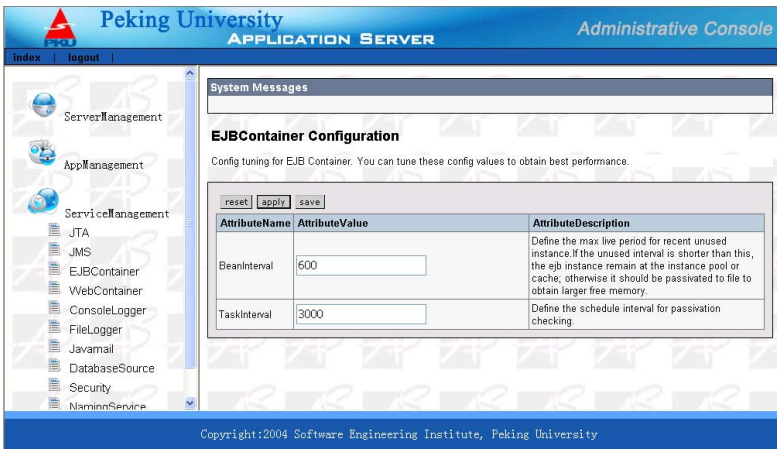

Figure 2.10 EJB Container

2.5.1.7 Mail Service

2.11JavaMail Service is similar to DatabaseSource Service.

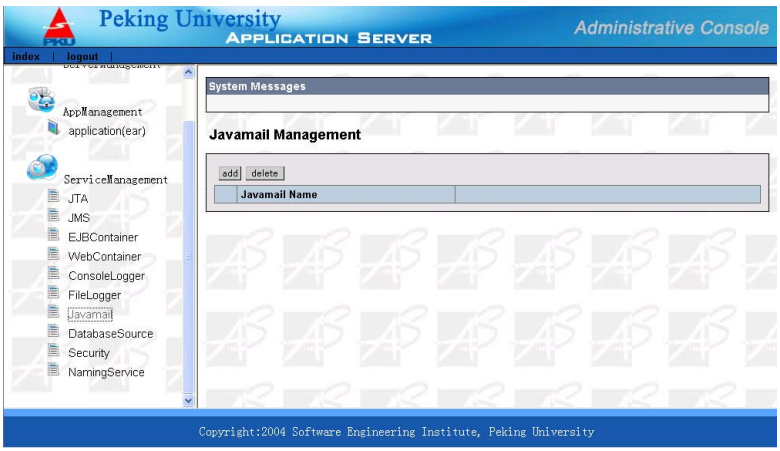

Figure 2.11 Mail Service

2.5.1.8 Datasource Service

2.12 DatabaseSource Service provides hot-modify of database sources,including add,delete and update them. In the following figure,all database sources are listed in this page.You can delete database source in this page by checking the box before its name.And you can see and modify its attributes in detail page after clicking the link behind its name.Click 'add' button will link to add page.

Add page:2.13

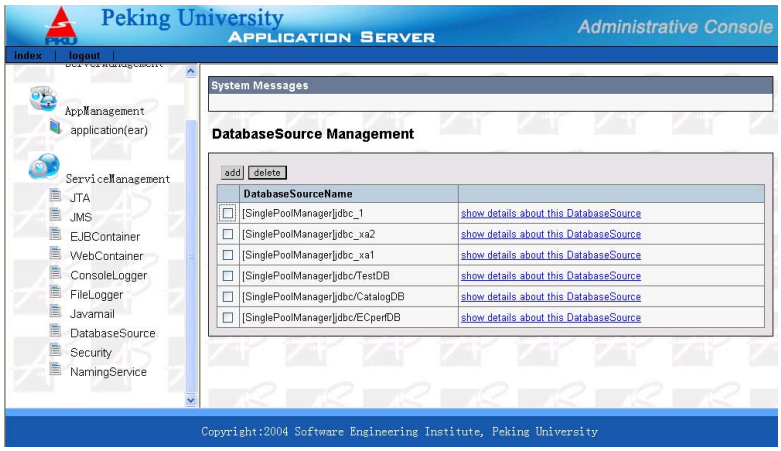

Figure 2.12 Datasource Information

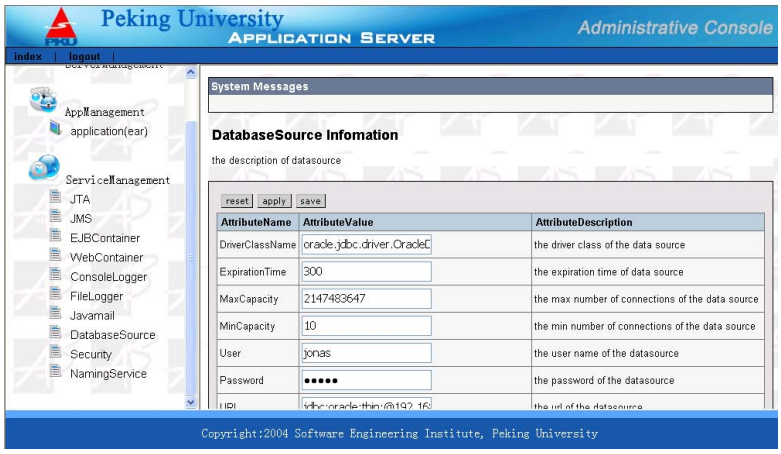

Figure 2.13 Add Datasource Page

2.5.2 Server Management

2.14Localhost and thread pool configuration can be modified in this page.

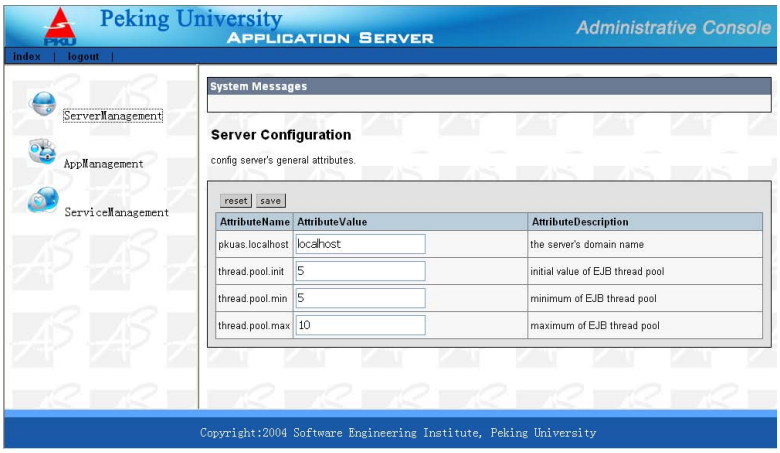

Figure 2.14 Server Management

2.6 tag

In the preceding part, it introduces how to use management tool(PMC) based on web page.This tool that accords with JSR77 is useful for administrator managing PKUAS.

Chapter 3. PKUAS JBuilder plugin

About

This section is for J2EE application developers.The purpose is to let them know and master how to develop PKUAS-sepefic J2EE applications within JBuilder's integrated development environment.This document gives detailed description for the plugin's installation and settings,but as to how to develop standard J2EE applications we give little explanation,because Borland has illuminate this in details in JBuilder's manual.Our emphasis is to explain the PKUAS-specic settings.

3.1 Installation

The JBuilder's Plugin for PKUAS follows the JBuidler opentool development framework and PKUAS' deployment descriptors. So far, it supports JBuilder 9.0 and the latest version of PKUAS. The use of the plugin is to translate a Borland Enterprise Server-specific J2EE application developed with JBuilder into a PKUAS-specific application, and provide the mechanism of setting and managing PKUAS in JBuilder.

The prerequisite softwares of JBuilder Plugin for PKUAS are listed below 3.1:

JBuilder	JBuilder 9.0: Enterprise Edition, Development Edition ,Foundation Edition
PKUAS	PKUAS' latest version

Figure 3.1 Installation Requirement

Setup steps:

1) Get the setup file -pkuasplugin.jar from PKUAS' installation disk.

2) Close the running JBuilder 9.0.

3) Navigate and open JBuilder 9.0's setup directory and we will use %JBuilder_ROOT% for this direcory.

4) Copy pkuasplugin.jar to the directory of %JBuilder_ROOT%/lib/ext.

5) Restart JBuilder, a triangle icon exists on toolbar if the installation is successful 3.2.

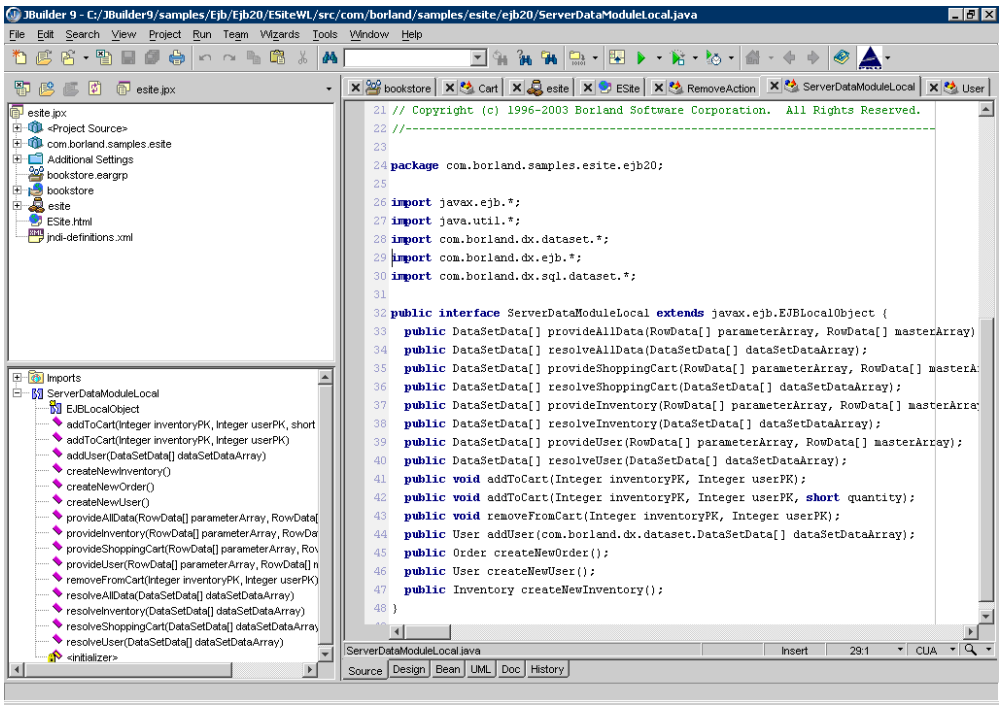

Figure 3.2 JBuilder

3.2 Configuration

JBuilder supports several types of J2EE application servers.The server should be configured before being used.We will focus on the configuration of PKUAS.

Firstly, PKUAS related information should be set.

Steps are:

1) start JBuilder,single-click the menu command Tools—Configure Servers.Select PKUAS in the popup window.See figure 3.3 below.

2) Copy %JAVA_HOME%/lib/tools.jar to %PKUAS_HOME%/libs/.In general panel, you need only choose PKUAS' setup direcory, and other directories will be set automaticly.See figure 3.4.

3) In custom panel, you can configure remote deployment.Select the remote deploy option box and enter the IP address and port of the remote server.See figure 3.5.

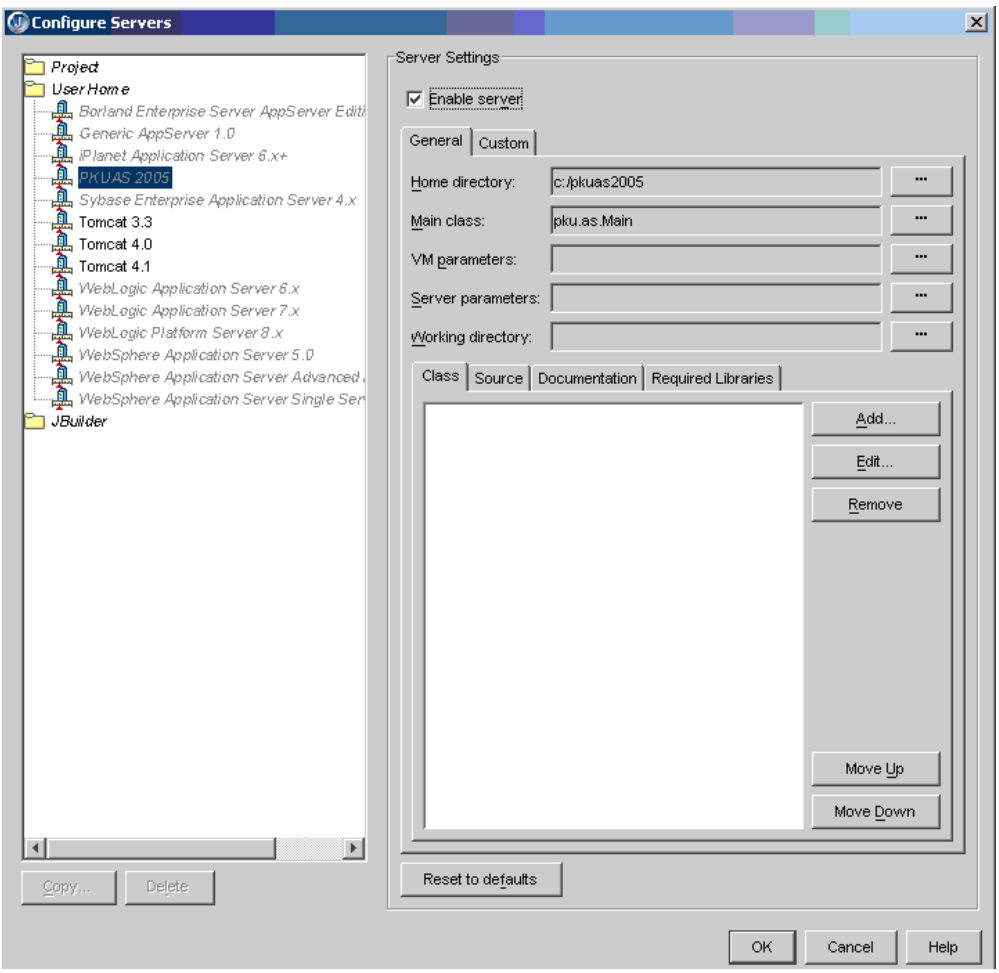

Figure 3.3 JBuilder Configuration 1

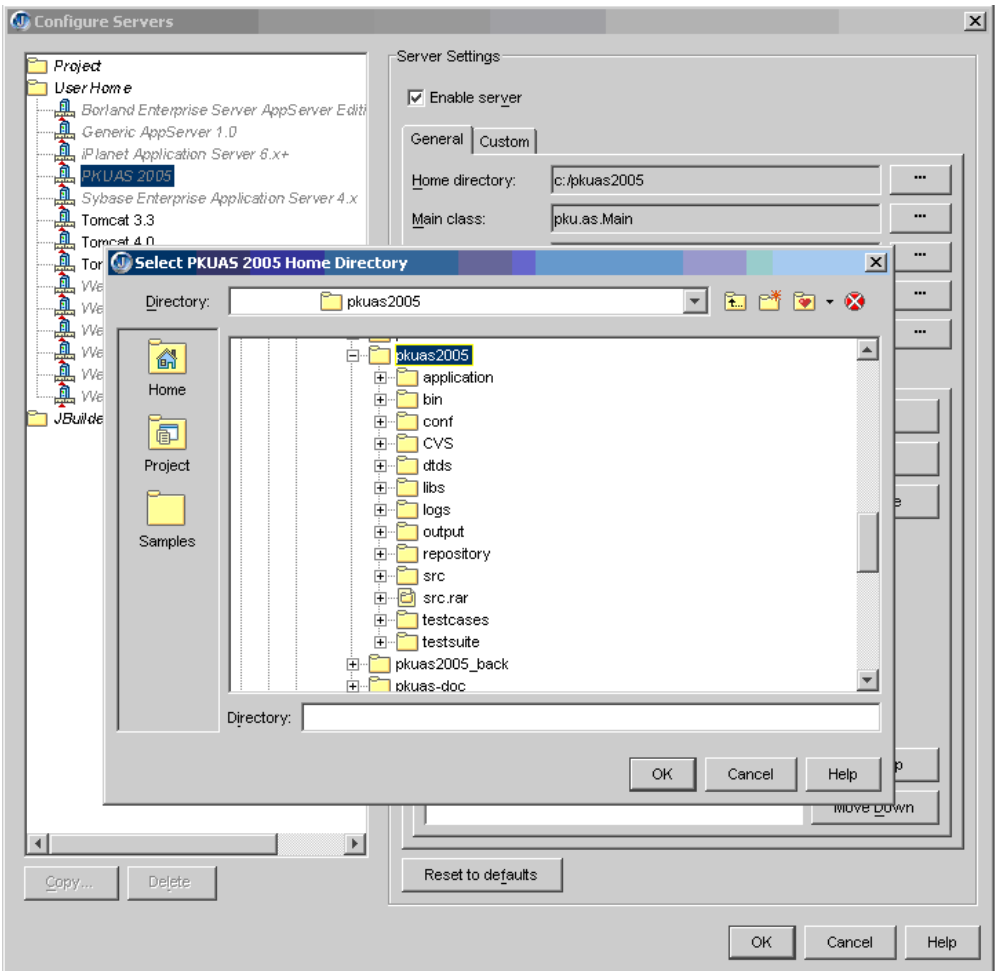

Figure 3.4 JBuilder Configuration 2

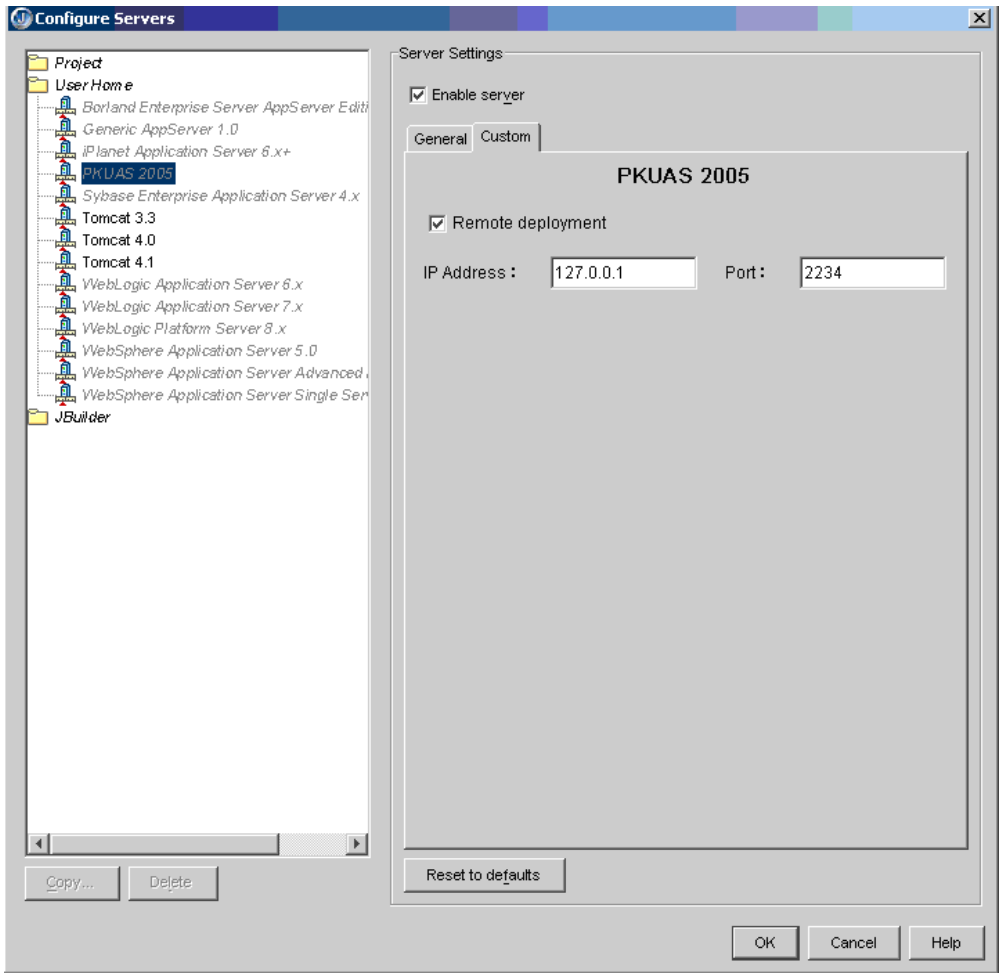

Figure 3.5 JBuilder Configuration 3

In order to use this server, you shoud change the property of the current project after finishing above settings.

Single-click the menu command Project—Project Properties,then select the server option panel,select the radio button of Single Server for all services in project,and select PKUAS 2005 in the list box,at last click the OK button.After doing this,when you rebuild your project,PKUAS-specifi jars,wars and ears will be created under the project directory.See figure 3.6.

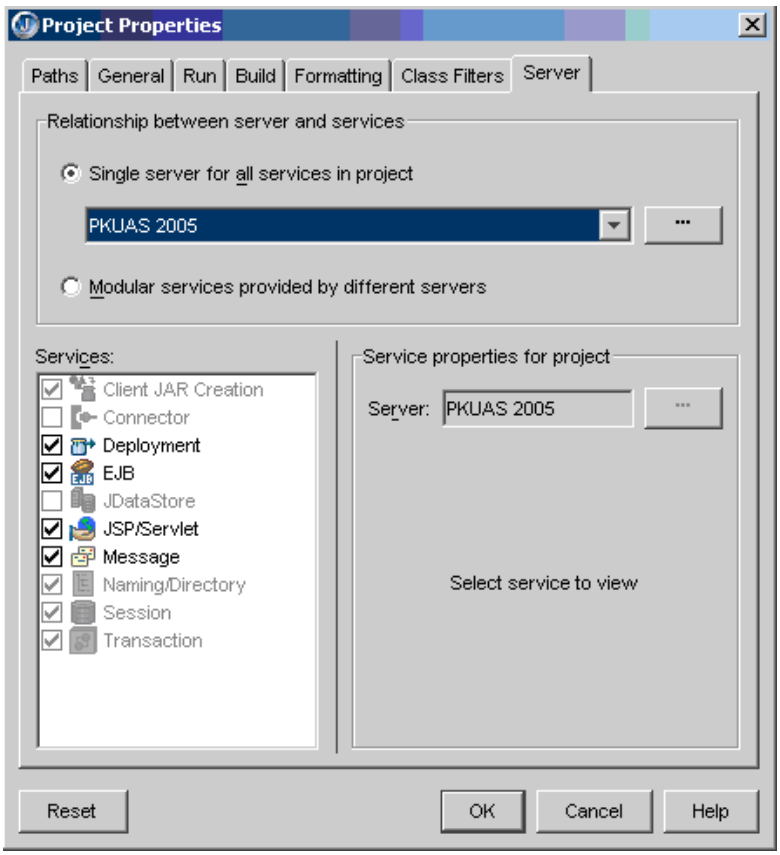

Figure 3.6 JBuilder Configuration 4

3.3 J2EE Application Development

With JBuilder, the way to develop applications based on PKUAS is similar with the way based on other Application Servers. Therefore, we only mention the differences. If you have problems with developing J2EE applications with JBuilder, please refer to JBuilder's developer manual.

3.3.1 Setup related properties

PKUAS allows J2EE Module to set some special attributes for PKUAS, these attributes are as follows:

mdb-user	User name of ”Message Driven Bean”
mdb-passwd	Password of ”Message Driven Bean”
mdb-client-id	Client ID of ”Message Driven Bean”
security-realm-name	Name of ”Security Realm”
utility-name	Name of ”Utility”
database-name Entity	Database Name for ”Entity Bean”, so far, we support Oracle(default), MySql and MS SqlServer
clustered	Whether or not using PKUAS cluster
evolution	Whether or not using PKUAS online evolution

Figure 3.7 PKUAS Properties

In JBuilder, the steps to set these attributes are as follows:

1) Open ”EJB Designer”

2) Select ”PKUAS Properties” label

3) Fill in/Select the related fields as shown in the following figure 3.8.

3.3.2 Develop EJB

In JBuilder, the way to develop Session Bean and Bean-Managed Persistent Entity Bean is the same with different Application Servers. However, for Container-Managed Persistent Entity Bean and Message Driven Bean, some attributes have to be set in order to use PKUAS. In the following, ”Plugin for PKUAS” will be introduced to set these attributes.

3.3.2.1 Develop CMP Entity Bean

Some basic settings should be done before following the steps below.Here we will focus on PKUAS-specific settings,as these basic settings mentioned above are similar for different Application Servers and developers may refer to JBuilder’s development manual for details.

- Set ”Data Source” for every Entity Bean

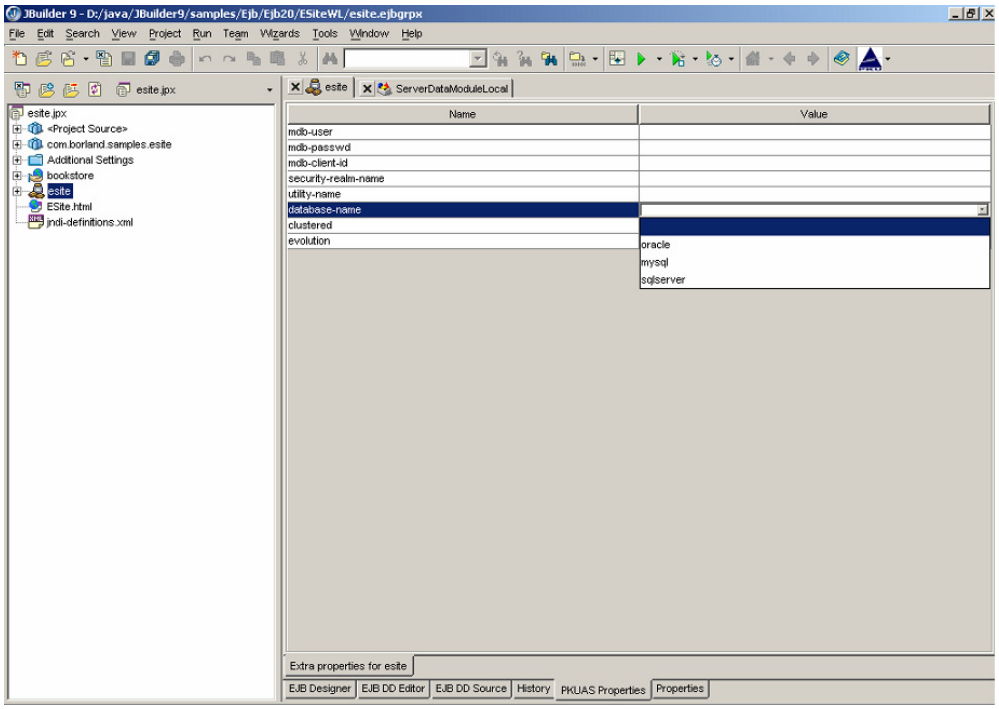

Figure 3.8 JBuilder Properties Setup

Open ”DD Editor” of Entity Bean, select ”Resource Reference” label. Click [Add] button, and add one record whose type is ”javax.sql.Datasource”, then fill in the ”JNDI Name” blank with the Datasource you will use. The results are as follows 3.9.

- Compound Keys

  The combination of several cmp fields can be used as the Primary Key of the Bean. In this case, JBuilder will automatically generate a Primary Key class named BeanNamePK. Generally speaking, users don’t need to pay attention to this class.

- All primary key fields must take part in the relationship if this bean has relation with another bean.

  Suppose : A and B are two Entity Beans,their database tables are table_a and table_b, the cmp fields of A are a1 and a2, that of B are b1,b2 b3, and relationship between A and B is recorded as R.

  1）If you use relation table (RT) to record this relationship, there have to be three

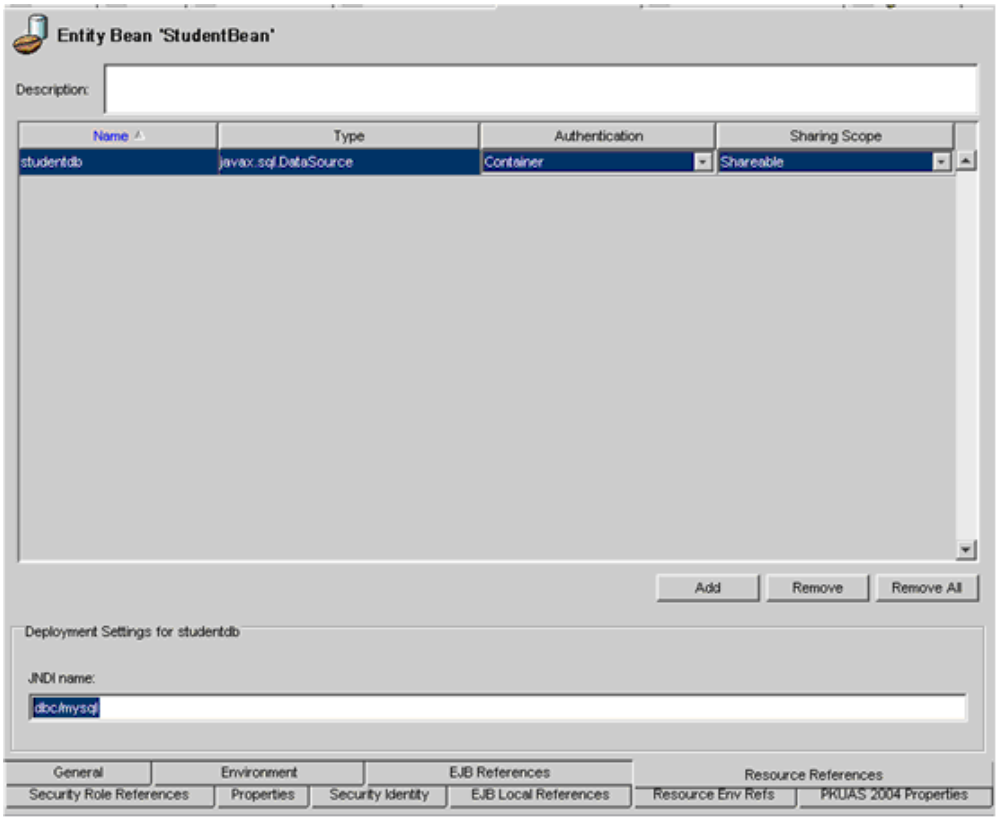

Figure 3.9 JBuilder CMP Development

columns in relation table, which are respectively related to a1, a2, b1, b2 and b3. 2）When you use foreign key table (FK) to record this relationship, if the foreign key table is table_a, table_a must have three foreign key columns related to b1, b2 and b3; if the foreign key table is table_b, table_b must have two columns related to a1 and a2.

3.3.2.2 Develop Message-Driven Bean

The Connection Factory name is jms/TopicConnectionFactory or jms/QueueConnectionFactory. Destination name is the same with that of the message producer, as shown in figure 3.10.

3.3.3 Develop JSP/Servlet

New one JSP, then add EJB and resource reference in file web.xml, as shown in the following figure. Fill in ”Link” blank with the jndi name of EJB or resource.See figure

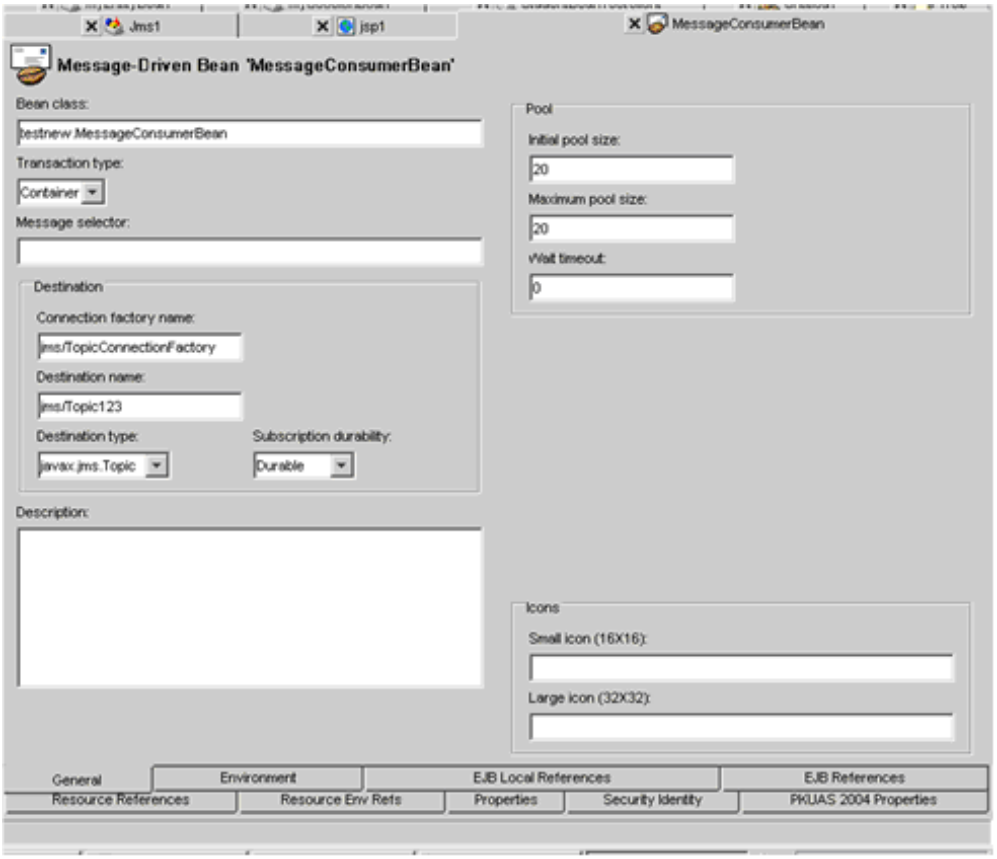

Figure 3.10 JBuilder MDB Development

3.11.

3.3.4 Develop Standalone EJB test Client

The general process of developing a Standalone EJB Test Client is the same with developing other EJB Test Client in JBuilder, but note that:

In the test program, you should add the following code or code with similar function

```
System.setProperty (”java.naming.factory.initial”,”pku.as.naming.SmartCtxFactory”);
```

Before

```
Context context = new InitialContext ();
```

3.4 Deploy Applications to PKUAS

Deployment of applications to PKUAS should follow these steps:

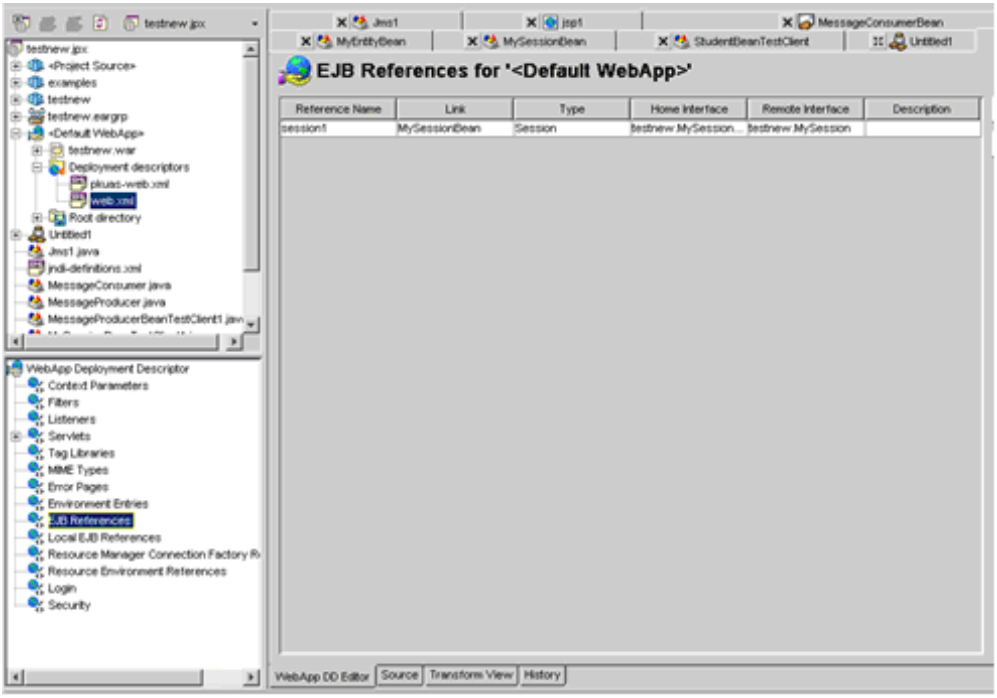

Figure 3.11 JBuilder JSP/Servlet Development

1) Start PKUAS.Click PKUAS' logo on the tool bar, and then select Start PKUAS on the drop-down menu.

2) Wait for PKUAS to finish its startup.PKUAS' startup information is displayed in the bottom output panel of JBuilder. PKUAS starts successfully if you see '[info] [Main] PKUAS V2004 started', otherwise, please check whether PKUAS' configuration is incorrect or whether there is running another PKUAS' instance,as shown in figure 3.12.

3) Compile your project.

4) Right-click the node you want to deploy in JBuilder's project pane, choose 'Deploy options for XXX', and choose Deploy,as shown in figure 3.13.

5) It is similar to redeploy or undeploy an application.

3.5 PKUAS Management

As JBuilder plugin for PKUAS has a mechanism to call PKUAS' management tool within JBuilder environment, you can open PKUAS' management interface in JBuilder.:

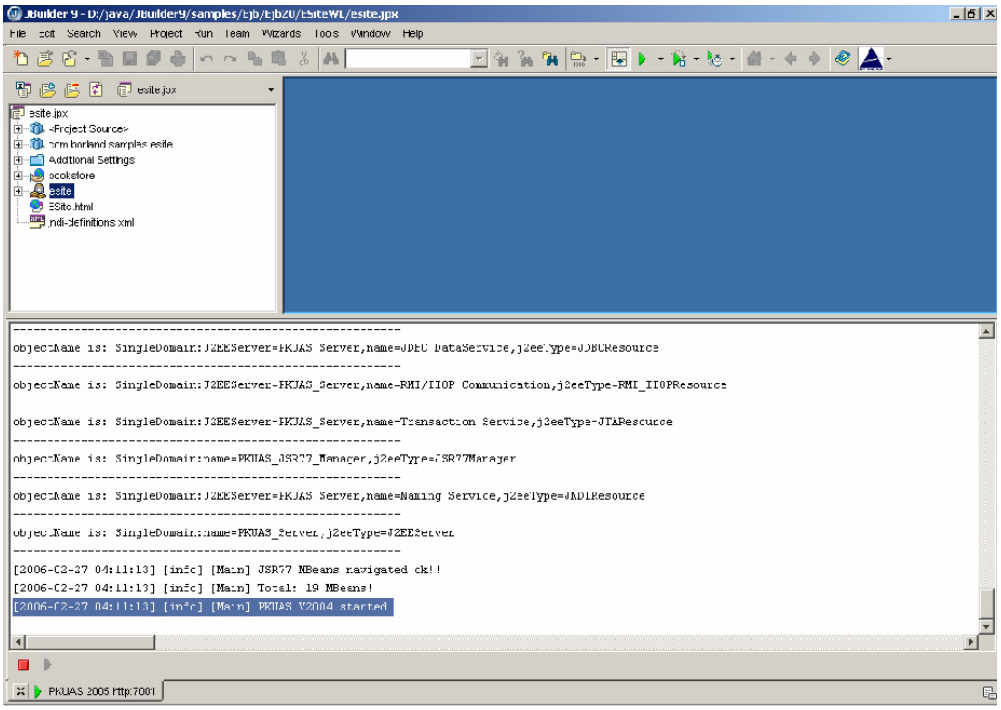

Figure 3.12 JBuilder Deployment 1

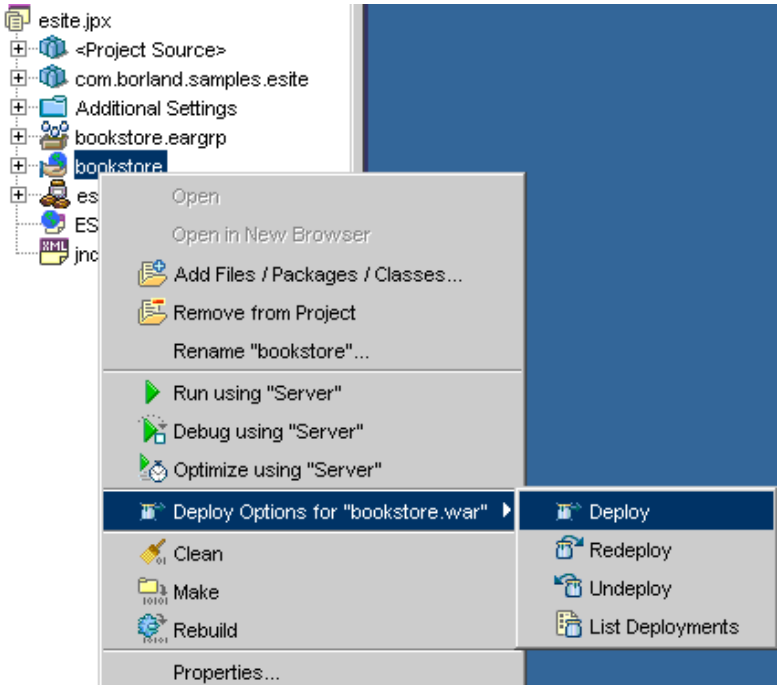

Figure 3.13 JBuilder Deployment 2

1) Click PKUAS' logo on JBuilder's tool bar and then choose 'Open PKUAS Manager' 3.14.

Figure 3.14 JBuilder PKUAS Icon

2) PKUAS' management interface will be opened in JBuilder's main window.See figure 3.15.

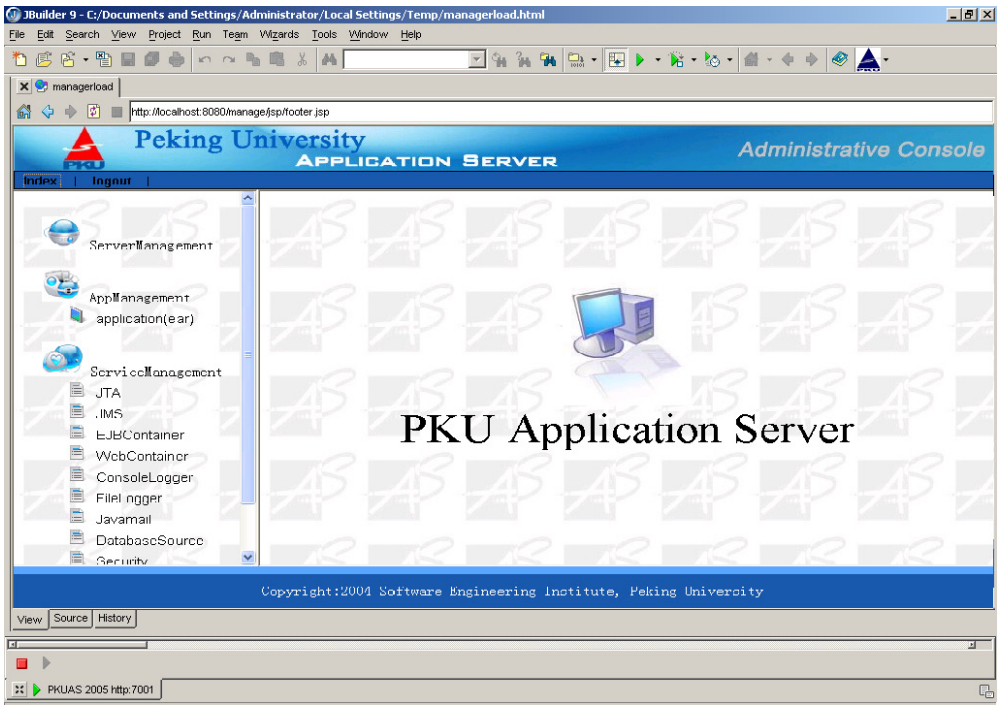

Figure 3.15 JBuilder Management Page

3) Please refer to Part two of this document for more details.

3.6 Caution

There is something you should pay attention to.

- Before you run some test programs to access some EJB, it is necessary to copy the stub files and homestub files located in the directory of %PKUAS_HOME%/repository/deployed/XXX/util to the directory where the class files of your testing program are located.You have an alternative:set the classpath of your testing program to include the stub and home stub directory.
- It is also necessary to copy the stub files and homestub files to the directory of /WEB-INF/classes of your war before the war is accessed.

GNU Free Documentation License

Version 1.2, November 2002

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document ”free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of ”copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The ”Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as ”you”. You accept the license if you copy, modify or distribute the work in a way

requiring permission under copyright law.

A ”Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A ”Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The ”Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The ”Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A ”Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not ”Transparent” is called ”Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The ”Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, ”Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

A section ”Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as ”Acknowledgements”, ”Dedications”, ”Endorsements”, or ”History”.) To ”Preserve the Title” of such a section when you modify the Document means that it remains a section ”Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies.

If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement. C. State on the Title page the name of the publisher of the Modified Version, as the publisher. D. Preserve all the copyright notices of the Document. E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. H. Include an unaltered copy of this License. I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title

of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. M. Delete any section Entitled ”Endorsements”. Such a section may not be included in the Modified Version. N. Do not retitle any existing section to be Entitled ”Endorsements” or to conflict in title with any Invariant Section. O. Preserve any Warranty Disclaimers. If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled ”Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties–for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents,

unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ”History” in the various original documents, forming one section Entitled ”History”; likewise combine any sections Entitled ”Acknowledgements”, and any sections Entitled ”Dedications”. You must delete all sections Entitled ”Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an ”aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Docu-

ment specifies that a particular numbered version of this License ”or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.