CA Workload Automation iXp

Administration Guide

Release 7.1 SP1

TABLE OF CONTENTS

1 INTRODUCTION

iXp is a Java™ based graphical interface than can be launched from Java Web Start. With iXp users can monitor, control, update, forecast, simulate, report on and print CA Workload Automation AE job streams across multiple instances. The iXp Admin Tool enables administrators to manage privileges for end users by defining authorization roles.

The iXp Command Line interface enables users to view, report, control, create, update, and delete CA Workload Automation AE jobs. The commands interface with the iXp Security model and do not require a local installation of the CA Workload Automation AE software.

This guide assumes familiarity with CA Workload Automation AE, and it assumes you have CA Workload Automation AE running, and you install iXp using the procedures described in the *Installation* section.

This ***iXp Administration Guide*** is intended for use with the companion ***iXp User Guide***.

2 PHILOSOPHY

Systems control and operations management will rapidly become web-based. In iXp, CA Technologies, Inc. delivers the next generation in graphical job control, while moving this critical systems task safely to the web. These guiding principles are central to the development of iXp:

1) The architecture should scale up as well as down. Users requiring minimal data should have minimal bandwidth requirements. Users requiring access to large volumes of data should expect quick response with conventional desktop hardware in a local area network.

2) The client should be easy to maintain. The only current requirement for the client to run iXp is to install the Java™ Plug-in. With respect to CA Workload Automation AE monitoring, the client does not require a CA Workload Automation AE client license. In addition, iXp administrators can configure user-defined commands that enable clients to perform custom server-side executions.

3) Management by exception is facilitated throughout iXp.

   - Alarms are selected to the top of the Console View.
   - Failures within boxes are visible while boxes are still running.
   - Sort by Status is available.
   - Impact analysis including critical path filtration facilitates response to exceptions.

4) Job filtration should be powerful and flexible. Efficient and intuitive filtration enables the defining of granular security roles and enables users to easily pinpoint subsections of job streams.

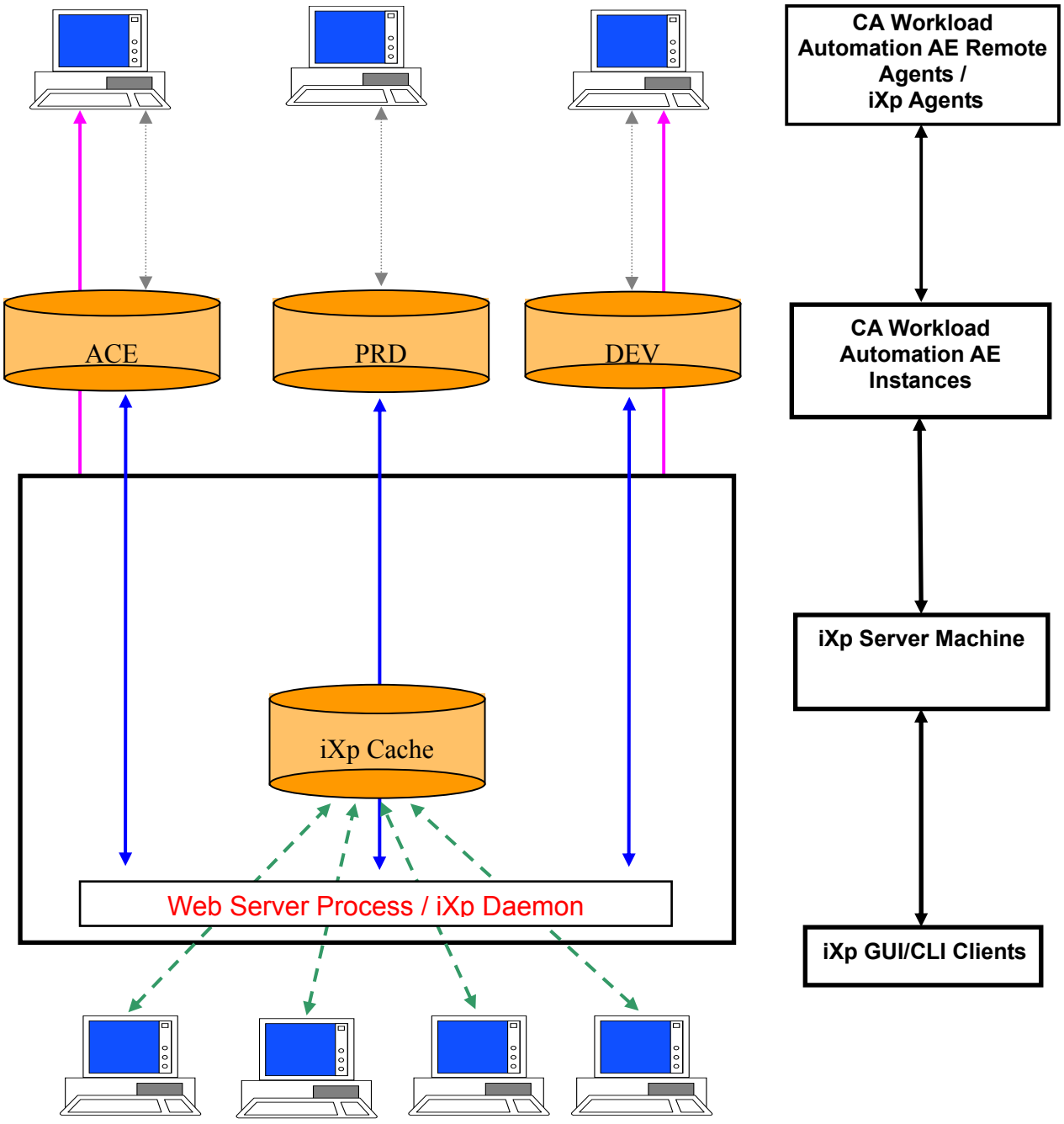
5) Management is consolidated across multiple instances and cross-instance relationships are depicted.

6) Management is consolidated across multiple database types.

7) Multi-threaded client architecture maintains access to the GUI during refreshes.

8) Three tiered architecture minimizes the number of direct connections to a CA Workload Automation AE instance. Multiple users connect to one iXp Server

3 ARCHITECTURE



ARCHITECTURE ELEMENT	DESCRIPTION
iXp Daemon	This process runs on the iXp Server machine and is the only process that runs constantly. It is a multi-threaded Java process that collects all the relevant job data from the CA Workload Automation AE instance(s), maintains the iXp Cache and supplies the data to all the iXp clients. It is started by the Web Application Server (e.g. Tomcat) as a servlet.
iXp Cache	These highly compressed data files in proprietary binary format include all the CA Workload Automation AE data needed by the iXp Daemon and the iXp clients.
iXp Client	The iXp Client is the Java™-based GUI that can be launched on any Java supported system, and the CLI that can be launched on supported systems. The GUI provides the capability to monitor, report, forecast, control and update job processing across the CA Workload Automation AE instances. Because the GUI is a Java applet, there is no software installation required on the machine that will launch the client. The CLI provides the capability to run CA Workload Automation AE commands like *sendevent*, *autorep*, *jil, autostatus, job_depends* from any machine without installing the CA Workload Automation AE client software.
iXp Agent	The iXp Agent is a JAVA program that runs on a CA Workload Automation AE Remote Agent. The Agent enables iXp Clients to retrieve the output files created by any CA Workload Automation AE job. iXp also provides a PERL-based agent that has been deprecated. This program can be started automatically by the iXp Daemon when needed, or can be started as a job from CA Workload Automation AE. This program can be installed automatically by the iXp Server or manually by the iXp Administrator. ***The iXp Agent is not required with CA Workload Automation AE r11.3 and higher.***

3.1 Processing Example

3.1.1 Data Refresh Cycle

Following is a detailed description of the processing of each component of the architecture and the interaction between them during a normal data refresh cycle for the iXp Daemon and the Client.

1) The iXp Daemon reads data from the CA Workload Automation AE database(s) for each instance.
2) It updates the iXp Cache with the newly fetched data.
3) The iXp Client contacts the web application server during its refresh cycle, and reads the iXp Cache for the data refresh. Since the iXp application is multi-threaded, it can service multiple client requests simultaneously, if needed.

3.1.2 On-Demand Activity

Following is a detailed description of the processing of each component and the interaction between them when an iXp Client performs an on-demand activity such as putting a job on-hold.

1) The iXp Client contacts the web application server.
2) The web application server associates the request with a thread and asks it to perform the action to put the job on-hold.
3) The thread uses the JDBC connection to the database to perform the action and reports the status back to the web application server.
4) The web application server then reports the status back to the iXp Client and exits.

3.1.3 Log File Retrieval

Following is a detailed description of the processing of each component and the interaction between them when an iXp Client requests retrieval of a log file (e.g. STDOUT) created by a job running in CA Workload Automation AE r11 or older version.

1) The iXp Client contacts the iXp Daemon via the web application server.
2) The iXp Daemon then requests the iXp Agent to read the desired log file and send the contents back to the iXp Server. If the iXp Agent is not running, then the iXp Daemon sends a message back to the Client indicating that the Agent is not running.

3) Once it receives the contents, the iXp Server sends the contents back to the requesting iXp Client. The iXp Client displays the contents in a browser window.

3.2 Security

There are two security methods available for controlling access to iXp.

1) Native iXp security.
2) Single-Sign On (SSO).

iXp provides a native multi-layered security model that encompasses data encryption, user authorization, and optional user authentication. The same security model is applied to the iXp GUI and the iXp CLI.

In the first model, user access to iXp requires validation of the user identity and password. All user identities and passwords are managed within iXp and have no relation to system/domain IDs and passwords.

In the second model, iXp leverages the user ID already authenticated to the Active Directory Domain or the local machine. This requires no user/password maintenance in iXp.

In either case, iXp security includes the following features.

- Data transmission between the server and client machines can be secured and encrypted through current secure HTTP technology. Also, the sensitive data stored by iXp is encrypted.

- User access to iXp requires validation of the user identity and password. All user identities and passwords are managed within iXp and have no relation to system IDs and passwords.

- Groups can be assigned to the iXp user identities that provide read, update and control privileges on the desired CA Workload Automation AE objects.

- The security model is very dynamic and user and group policies can be changed on the fly. Changes immediately take into effect and a user logout is not required.

- The iXp Administrator can easily manage the security policies through the iXp *Admin Tool*.

4 INSTALLATION

4.1 Overview

As a second-generation companion tool to CA Workload Automation AE, iXp is a Java™ based GUI that can be launched from Java Web Start. With the iXp GUI, users can monitor, control, update, forecast, simulate, report on and print the job processing environment across multiple CA Workload Automation AE instances from a single web browser window. Based on a true web-based 3-tiered architecture developed in Java™, iXp is highly scalable, efficient, fast and minimal in terms of impact on the network bandwidth and the CA Workload Automation AE repository.

The iXp Client CLI enables users to execute CA Workload Automation AE commands from any client machine without having to install the CA Workload Automation AE software. The iXp Client CLI interfaces with the iXp Security Model, hence providing a secure access to the CA Workload Automation AE environment.

This section contains steps to install the iXp Server components on UNIX, Windows and LINUX Web Servers.

4.2 Components

As seen in the *Architecture* diagram, the product has two components that are executed on the iXp Server or iXp Client machines.

4.2.1 iXp Daemon

This is the only persistent daemon in the iXp system. This process is installed on a Web Application Server that can provide access to the different CA Workload Automation AE instance repositories. The iXp Daemon process is a multi-threaded Java™ process that periodically reads job processing data from the different CA Workload Automation AE repositories and maintains the iXp Cache from which data is provided to the iXp Clients. The Web Application Server starts this process as a servlet. The main tasks of installation are:

1) Satisfying software prerequisites.
2) Installing iXp files.
3) Configuring iXp so it can communicate with all the CA Workload Automation AE instances to be managed.
4) Configuring user security.

4.2.2 iXp Client

iXp Client is the Java™-based graphical interface that can be launched on any system that supports Java 1.5 and Java 1.6. The client provides the capability to monitor, report, forecast, control and update job processing across the CA Workload Automation AE instances. Because the client is a Java applet, there is no software installation required on the machine where it will be launched.

The iXp Client CLI can be installed on any supported system. This requires installation and configuration of the iXp Client CLI software. We recommend installing the Client CLI on those machines from which users would like to run certain CA Workload Automation AE commands (e.g. `autorep`, `autocal_asc`, `sendevent`, `jil`). You do not need to install the CA Workload Automation AE Client software on the machines where the iXp Client CLI is installed.

4.2.3 iXp Agent

The iXp Agent is a JAVA program that runs on a CA Workload Automation AE Remote Agent. The Agent enables iXp Clients to retrieve the output files created by any CA Workload Automation AE job.

iXp also provides a PERL-based agent that has been deprecated. This program can be started automatically by the iXp Daemon when needed, or can be started as a job from CA Workload Automation AE. This program can

be installed automatically by the iXp Server or manually by the iXp Administrator.

Note: The iXp Agent is not required with CA Workload Automation AE r11.3 and higher.

4.3 Basic System Requirements

Following is a list of the system requirements for installation and running of the iXp components. Some of the pre-requisite software is already present on the machines when the operating system is loaded.

4.3.1 iXp Daemon

1) Tomcat 6.x

2) JRE 6.x

3) PERL v5.1 or higher.

4) 2Gb of disk space.

5) 2Gb of available RAM.

6) CA Workload Automation AE Full Client and configuration files for all instances of CA Workload Automation AE that will be accessed.

4.3.2 iXp Client

1) Java v1.5.x and v1.6.x

2) 1Gb of available RAM.

3) 5Mb of disk space required if installing the iXp Client CLI.

4.3.3 iXp Agent

1) Java 1.5.x and v1.6.x (Optionally, PERL v5.1 or higher for the deprecated agent program)

2) The CA Workload Automation AE `sendevent`, `zql`, `xql` binaries are optional. They are needed if it is desired to install and start the PERL-based iXp Agent automatically by the iXp Daemon.

4.4 Server Pre-Installation Checklist

Following is a checklist of information required for installation. Please fill it in with the values that you will use during the installation process. Beneath is a checklist with filled-in sample values.

Hostname of the iXp Web Server	
IP Address of the iXp Web Server	
Port number of the Web Server process	
Secure or Standard Web Server process (`http` or `https`)	
Total Number of CA Workload Automation AE Instances that will be monitored	
Web Server Type and Version	
Location of Web Server `webapps` directory	
Directory for JRE 1.5.x binaries	
CA Workload Automation AE RDBMS Vendor	
Location and name of PERL v5.x executable	
CA Workload Automation AE Root Directory (`AUTOSYS`) on the Web Server	
Directory for CA Workload Automation AE Configuration Files (`AUTOUSER`) on the Web Server	
Directory for Sybase interfaces file (`SYBASE`) on the Web Server	
Directory for Oracle `tnsnames` file (`TNS_ADMIN`) on the Web Server	
Oracle Home Directory (`ORACLE_HOME`) on the Web Server	
Root Directory for iXp files (`IXP_HOME`) on the iXp Server	
Root Directory for iXp files (IXP_HOME) on the iXp Clients and iXp Agents (for CLI and Agent program)	

4.4.1 Example Server Pre-Install Checklist Form

Following is a checklist with sample values filled in.

Hostname of the Web Server	venus
IP Address of the Web Server	192.168.200.19
Port number of the Web Server process	8080
Secure or Standard Web Server process (`http` or `https`)	Standard (http)
Total Number of CA Workload Automation AE Instances that will be monitored	3 (PRD, DEV, TST)
Web Server Type and Version	Tomcat v6.0.35
Location of Web Server `webapps` directory	/opt/tomcat6/webapps
Directory for JRE 1.5 binaries	/opt/jre1.6/bin
CA Workload Automation AE RDBMS Vendor	Sybase
Location and name of PERL v5.x executable	/opt/perl5/bin/perl
CA Workload Automation AE Root Directory (`AUTOSYS`) on the Web Server	/opt/CA/UnicenterAutoSysJM/autosys
Directory for CA Workload Automation AE Configuration Files (`AUTOUSER`) on the Web Server	/opt/CA/UnicenterAutoSysJM/autouser.PRD
Directory for Sybase interfaces file (`SYBASE`) on the Web Server	/opt/sybase
Directory for Oracle tnsnames file (`TNS_ADMIN`) on the Web Server	N/A
Oracle Home Directory on the Web Server (`ORACLE_HOME`)	N/A
Root Directory where iXp files will be installed (`IXP_HOME`)	/opt/ca/ixp
Root Directory for iXp files (IXP_HOME) on the iXp Clients and iXp Agents (for CLI)	C:\ca\ixp

4.5 Installing iXp

4.5.1 Configuring/Installing JRE v1.6

Before installing the iXp Server, the Java environment on the machine needs to be present. The iXp Server needs the Java Runtime Environment (JRE) v1.5.x.

1. To check the version of Java on the server machine, run the following command from a command prompt.

   ```
   % java –version

   java version "1.6.0_34"
   Java(TM) SE Runtime Environment (build 1.6.0_34-b04)
   Java HotSpot(TM) 64-Bit Server VM (build 20.9-b04,
   mixed mode)
   ```

   If the output of the command looks like the sample above, then the Java environment installed is set-up correctly, and is of the required version.

2. If the version of java is not 1.6.x, or if the Java environment is not present, then it will be necessary to install the appropriate version. Typically, the `java` executable that was run above is in the `/usr/bin` directory on UNIX/LINUX systems and `C:\Windows\System32` on Windows systems. On UNIX/LINUX, it is a soft link to the real `java` executable in the directory where the Java environment resides. There is a possibility that the required version of Java is present on the server machine, but it is not present in the `PATH` of the user. Before installing the new version of Java, it may be helpful to search the server machine for any other versions of Java.

   If the JRE has to be downloaded, the java.oracle.com website (or *http://www.oracle.com/technetwork/java/javase/downloads/index.html*) has download instructions for most platforms. If your platform is not listed, please contact CA Technologies, Inc. for appropriate download directions.

   Once the JRE files have been located, put the directory in the `PATH` environment variable of the user that will install and run iXp. For example,

   `PATH=/opt/jre1.6/bin:$PATH; export $PATH` (On UNIX/LINUX)

4.5.2 Installing the iXp Server

The iXp software is delivered in two DVD image files:

The ISO file for CA Workload Automation iXp CA Third Party Requirements
The ISO file for CA Workload Automation iXp 7.1 SP1

Create the directory where all the iXp files will be stored (the iXp Root Directory), as defined in the *Pre-Installation checklist*. Make sure the directory can store up to 50Mb of data. Before proceeding with the installation, please contact CA Customer Care (*http://www.ca.com/us/customer-care.aspx*) and provide the hostname, fully qualified hostname, and IP address of the webserver for creation of your license file.

1) Log on to the Web Server machine as the user that runs the web application server (Tomcat). This user must have the following privileges:
   a. Full privileges on the iXp home directory and also all the subsequent sub-directories
   b. Edit Superuser capability on all CA Workload Automation AE instances (Only if users will be performing job updates through iXp).

2) Mount the ISO file for CA Workload Automation iXp CA Third Party Requirements.

3) From the ISO image mount point, run the "installer.bat" (if Windows) or "installer.sh" (if UNIX).

4) You must agree to the EULA (End User License Agreement) to continue.

5) On UNIX/Linux, extract the tar file “ixp-thirdPartyLib-7.1.1.1.tar” from the ISO image to the "lib" directory under Tomcat's location. On Windows, extract the zip file “ixp-thirdPartyLib-7.1.1.1.zip” to the “lib” directory under Tomcat’s location.
6) Mount the ISO image for CA Workload Automation iXp 7.1 SP1
7) From the ISO image mount point, run the "installer.bat" (if Windows) or "installer.sh" (if UNIX).

8) You must agree to the EULA (End User License Agreement) to continue.

9) Set the following environment variables.

```
IXP_HOME=<Root Directory of the iXp files>
JAVA_OPTS=-Dixp.home=$IXP_HOME –Xms500m –Xmx500m
JAVA_HOME=<Root Directory of Java>
```

For example,

```
% IXP_HOME=/export/home/ixphome; export IXP_HOME
% JAVA_OPTS="-Dixp.home=$IXP_HOME -Xms500m -Xmx500m"
% JAVA_HOME=/opt/jre1.6
% export JAVA_OPTS JAVA_HOME
```

10) On Windows, launch the "Configure Tomcat" menu item from the Windows Start Menu. In this dialog, make the following changes:
    a. Under the "General" tab, change the "Startup type" to "Automatic" if desired.
    b. Under the "Log On" tab, change the service to run as a Windows user that has full Edit and Exec super user privileges to all the CA Workload Automation AE instances.
    c. Under the "Java" tab, add the following three lines to the "Java Options" item:

       -Dixp.home=IXP_HOME
       -Xms500m
       -Xmx500m

       where,

       IXP_HOME=The root directory for iXp software. This value has to be specified in the Windows 8.3 format.

11) On Unix/Linux, add the lines from step#5 to the login profile of the user that runs the web application server.

12) On Unix/Linux, extract the tar file "ixp-distrib-7.1.1.1.tar" from the ISO image to the IXP_HOME directory. On Windows, extract the zip file "ixp-distrib-7.1.1.1.zip" to the IXP_HOME directory.

13) Go to the `IXP_HOME/webapps` sub-directory and copy the iXp WAR file (`ixp.war`) to the appropriate directory of the servlet engine. For the above example, copy the file to `/opt/tomcat6/webapps` directory. If this directory has a previous iXp WAR file, then delete that file before copying this new file. Also, delete the `/opt/tomcat6/webapps/ixp` directory and all of its contents.

14) Restart the Web Application Server and monitor the log files
    `$IXP_HOME/log/ixpDaemon.log`
    —and—
    `$CATALINA_HOME/logs/stdout.log`
to make sure that iXp comes up without error.

4.5.3 Installing the iXp Agents

The iXp Agents run on the Remote Agent machines from which the iXp Clients will retrieve job log files. The iXp Agent can be installed at your convenience. The iXp distribution image includes the software to install the full client (which includes the iXp CLI and Agent), or only the iXp Agent. The software is located in the "`install`" sub-directory under the iXp root directory.

Note: The iXp Agent is not required with CA Workload Automation AE r11.3 and higher.

4.5.3.1 Installing Java Agent on UNIX/LINUX

The software files for the full Client, and Agent only installation are "`ixp-client-RELEASE-PLATFORM.tar`" and "`ixp-agent-RELEASE-PLATFORM.tar`" respectively, where `RELEASE` = the version of iXp, and `PLATFORM` = the OS platform for which the Agent has been compiled.

You need to copy the file appropriate for the Agent platform to the Agent machine.

Once the file has been copied to the Agent machine, the following steps can be performed to install and configure the Agent.

1) Create a directory that will contain the Agent software.
2) Extract the contents of the TAR file to this directory.
3) Modify the Environment profile file for the user running the Agent program to include the following environment variables.

```
IXP_HOME=<Root Directory of Agent Software>; export
IXP_HOME
PATH=JAVABIN:$PATH; export PATH
```

where,

`JAVABIN` = The directory that contains binaries for Java 1.5.x or higher

1) If `$IXP_HOME/etc/ixp.conf` file does not exist, you can copy the sample file provided (`$IXP_HOME/etc/ixp.conf.sample`) to `ixp.conf`.

```
cp $IXP_HOME/etc/ixp.conf.sample $IXP_HOME/etc/ixp.conf
```

2) Edit the `ixp.conf` file to specify the location of the iXp Server.

   `IXP_SERVER_URL=PROTOCOL://IXPSERVERNAME:PORT`

   where,

   `PROTOCOL` = `http` or `https`, depending upon the configuration
   `IXPSERVERNAME` = Name or IP Address of the iXp Server machine
   `PORT` = The Port number of the iXp Server process

   For example,

   `IXP_SERVER_URL=http://venus:8081`

3) Normally, CA Workload Automation AE jobs refer to certain environment variables when specifying the STDOUT/STDERR file names. Also, the default location of Remote Agent log files can be configured to be different than `/tmp`. We recommend creating a file for each CA Workload Automation AE instance that specifies the values for such variables. That way, when users request log files that contain names of environment variables, the Agent will be able to substitute the appropriate values.

   For each instance, create `$IXP_HOME/agent/conf/$AUTOSERV.conf` files, where `$AUTOSERV` = the 3 letter name for the CA Workload Automation AE instance.

   For example,

   `$IXP_HOME/agent/conf/DEV.conf` for the `DEV` instance.
   `$IXP_HOME/agent/conf/TST.conf` for the `TST` instance.

   The following is a sample of environment variables you could specify in each file.

```
AUTOSYS=/opt/CA/UnicenterAutoSysJM/autosys
AUTOUSER=/opt/CA/UnicenterAutoSysJM/autouser
AutoRemoteDir=/opt/CA/UnicenterAutoSysJM/tmp
LOGDIR=/opt/CA/UnicenterAutoSysJM/logs
```

4.5.3.2 Starting Java Agent on UNIX/LINUX

After the iXp Agent has been configured as listed above, it has to be started so that the log files can be retrieved from this machine. You need to start the Agent only once per iXp Server. You can start the Agent in various ways.

- Create a CA Workload Automation AE job to start the Agent. You will need one job per Agent machine.
- Have the standard startup scripts for the machine start the Agent when the machine boots up.
- Start the Agent process by running the startup command in the background of an interactive UNIX command window.

The Agent can be run in one of the following two ways.

- A persistent process that will only stop if it encounters a fatal error, or if the iXp Server is not available for more than 30 minutes.
- A timed process that will gracefully finish when the specified duration is reached. It will also stop if it encounters a fatal error, or if the iXp server is not available for more than 30 minutes.

The syntax to run the iXp Agent is as follows:

```
$IXP_HOME/bin/ixagent SERVER_URL AGENT_MACHINE
DURATION
```

where,

`SERVER_URL` = The URL connection string for the iXp Server

`AGENT_MACHINE` = The name of the local Agent machine that matches the name specified in the Run machine field of the CA Workload Automation AE job.

`DURATION` = The time, in minutes, for which the Agent will keep on running. If you want to run it persistently, specify 0 (zero).

For example,

```
$IXP_HOME/bin/ixagent http://venus:8080 neptune 0
```

This will run the agent as a persistent process, and it will connect to the iXp server running on `venus` over port `8080`. The agent will pick up all log file requests for jobs that ran on machine `neptune`.

```
$IXP_HOME/bin/ixagent http://venus:8080 neptune.ca.com 10080
```

This will run the agent for 7 days, and it will pick up all log file requests for jobs that ran on machine `neptune.ca.com`.

NOTE: You will need to run multiple executions of the Agent concurrently if you have jobs running on the same machine but using different hostnames, DNS aliases, short and fully qualified names.

4.5.3.3 Installing Java Agent on Windows

The software files for the full Client, and Agent only installation are "`ixp-client-RELEASE-windows.zip`" and "`ixp-agent-RELEASE-windows.zip`" respectively, where `RELEASE` = the version of iXp.

You need to copy the desired software image file to the Windows Agent machine.

Once the file has been copied to the Agent machine, the following steps can be performed to install and configure the Agent.

1) Create a directory that will contain the Agent software.
2) Extract the contents of the ZIP file to this directory.
3) Modify the System Environment so that the following environment variables are present when the Agent process is started. You will have to go to the `Control Panel->System->Advanced->Environment Variables` dialog to do this.

```
IXP_HOME=<Root Directory of Agent software>
PATH=JAVABIN;%PATH%
```

1) If `%IXP_HOME%\etc\ixp.conf` file does not exist, you can copy the sample file provided (`%IXP_HOME%\etc\ixp.conf.sample`) to `ixp.conf`.

```
copy %IXP_HOME%\etc\ixp.conf.sample
%IXP_HOME%\etc\ixp.conf
```

2) Edit the `ixp.conf` file to specify the location of the iXp Server.

```
IXP_SERVER_URL=PROTOCOL://IXPSERVERNAME:PORT
```

where,

PROTOCOL = http or https, depending upon the configuration
IXPSERVERNAME = Name or IP Address of the iXp Server machine
PORT = The Port number of the iXp Server process

For example,

```
IXP_SERVER_URL=http://venus:8080
```

The IXP_SERVER_URL environment variable overrides the ixp.conf file. If this variable (optional: IXP_SERVER_URL_2) is set, the value in the ixp.conf file is ignored. This simplifies the use of multiple iXp environments.

3) Normally, CA Workload Automation AE jobs refer to certain environment variables when specifying the STDOUT/STDERR file names. Also, the default value for the Remote Agent logging directory is stored within the CA Workload Automation AE Remote Agent configuration only. We recommend creating a file for each CA Workload Automation AE instance that specifies the values for such variables. That way, when users request log files that contain names of environment variables, the Agent will be able to substitute the appropriate values.

For each instance, create %IXP_HOME\agent\conf\%AUTOSERV%.conf files, where %AUTOSERV% = the 3 letter name for the CA Workload Automation AE instance.

For example,

%IXP_HOME%\agent\conf\DEV.conf for the DEV instance.
%IXP_HOME%\agent\conf\TST.conf for the TST instance.

The following is a sample of environment variables you could specify in each file.

```
AUTOSYS=C:\Progra~1\CA\Unicen~1.DEV\autosys
AUTOUSER=C:\Progra~1\CA\Unicen~1.DEV\autouser
AutoRemoteDir=C:\Progra~1\CA\Unicen~1.DEV\tmp
LOGDIR=C:\Progra~1\CA\Unicen~1.DEV\logs
```

NOTE: You have to specify at least the AutoRemoteDir environment variable, or else the iXp Agent will be unable to retrieve the Remote Agent log files.

4.5.3.4 Starting Java Agent on Windows

After the iXp Agent has been configured as listed above, it has to be started so that the log files can be retrieved from this machine. You need to start the Agent only once per iXp Server. You can start the Agent in the following way.

- Create a CA Workload Automation AE job to start the Agent. You will need one job per Agent machine.

NOTE: Since the Agent will run as a CA Workload Automation AE job, you have to restart the CA Workload Automation AE Remote Agent service so that the new environment variables can be passed to the Remote Agent.

The Agent can be run in one of the following two ways.

- A persistent process that will only stop if it encounters a fatal error, or if the iXp Server is not available for more than 30 minutes.
- A timed process that will gracefully finish when the specified duration is reached. It will also stop if it encounters a fatal error, or if the iXp server is not available for more than 30 minutes. If you choose this method, specify the date/time conditions for the job such that the job runs every day and has start_mins of 5 minutes. That way, when the Agent finishes, the job will restart the Agent within 5 minutes.

The syntax to run the iXp Agent is as follows:

```
%IXP_HOME%\bin\ixagent.exe SERVER_URL AGENT_MACHINE
DURATION
```

where,

`SERVER_URL` = The URL connection string for the iXp Server

`AGENT_MACHINE` = The name of the local Agent machine that matches the name specified in the Run machine field of the CA Workload Automation AE job.

`DURATION` = The time, in minutes, for which the Agent will keep on running. If you want to run it persistently, specify 0 (zero).

For example,

```
%IXP_HOME%\bin\ixagent.exe http://venus:8080 neptune 0
```

This will run the agent as a persistent process, and it will connect to the iXp server running on `venus` over port `8080`. The agent will pick up all log file requests for jobs that ran on machine `neptune`.

```
%IXP_HOME%\bin\ixagent.exe http://venus:8080 neptune.ca.com 10080
```

This will run the agent for 7 days, and it will pick up all log file requests for jobs that ran on machine `neptune.ca.com`.

NOTE: You will need to run multiple executions of the Agent concurrently if you have jobs running on the same machine but using different hostnames, DNS aliases, short and fully qualified names.

4.5.3.5 Installing PERL Agent

If any machine on which you wish to run the iXp Agent does not have Java v1.5.x or higher, but has PERL v5.6.x or higher installed, then you can run the deprecated iXp Agent. The PERL script for the iXp Agent can be installed manually on every system or automatically by the iXp Daemon when needed.

4.5.3.6 Copying the Script

The Agent Script is provided with the iXp distribution. The file "`$IXP_HOME/etc/runlog/ixpagent.pl`" is the Agent Script. You can copy this script to each machine and then run it to start the iXp Agent.

4.5.3.7 Loading the PERL Script

The Agent Script can also be loaded as a database table on any relational database. The advantage of this mechanism is that the script can be extracted to any machine by using a database command, rather than copying the file (as per the previous section). By default, you can use the CA Workload Automation AE database on each instance. **If you copied the PERL script on to the Remote Agent machines, you can skip this section.**

The following steps will load the Agent Script in to the database of choice.

1) cd $IXP_HOME/etc/runlog

2) Load the script using the appropriate command for the database type.

For **Oracle** database, use the following command:

```
zql -u <User> -p <Password> -s <TNS_ALIAS> -f
ixpagent.zql
```

For **Sybase**, use the following command:

```
isql -U <User> -P <Password> -S <Dataserver> -i
ixpagent.sql
```

For **SQL Server**, use the following command:

```
osql -U <User> -P <Password> -S <Dataserver> -i
ixpagent.sql
```

where,

`<User>`	- The Database logon userid
`<Password>`	- The Database logon password
`<TNS_ALIAS>`	- Oracle Database name
`<Dataserver>`	- Sybase Dataserver or SQL Server name

The above commands will create a table called "`IXP_AGENT`" on the database provided. The rows inside this table shall contain the actual PERL script of the iXp Agent.

If the iXp Server machine does not have a database client, then copy the `IXP_HOME/etc/runlog/ixpagent.sql` file to any machine that has a database client and is configured to reach the desired database, and run the above command from that machine. For example, you can copy the file to the CA Workload Automation AE Event Processor machine and run the above commands from there.

4.5.3.8 Starting the PERL Agent

We recommend that the iXp Agent should be installed manually. After the script has been loaded in a database, the Agent is ready for installation on the desired Remote Agent machines.

1) Extract the script from the database by running the following command on the machine from which the script was loaded. **If you copied the PERL script on to the Remote Agent machines, you can skip this step.**

For **Oracle** database, use the following command:

```
zql -u <User> -p <Password> -s <TNS_ALIAS> -qh -c
“select text from ixp_agent order by line” >
ixpagent.pl
```

For **Sybase** or **SQL Server**, use the following command:

```
xql -U <User> -P <Password> -S <Dataserver> -qh -c
“select text from ixp_agent order by line” >
ixpagent.pl
```

2) Copy the extracted script to a directory on every Remote Agent machine. In the above case, the file “ixpagent.pl” that was created by extracting the information from the database needs to be copied to each Remote Agent machine.

3) Run the script as follows:

perl <FILE> http://<iXpServer>:<PORT> <Hostname> <Duration>

where,

<FILE>	- The full path and name of ixpagent.pl
<iXpServer>	- The name of the iXp Server machine
<PORT>	- The Port number for Tomcat on iXp Server
<Hostname>	- The name of the iXp Agent machine
<Duration>	- The amount of time (in minutes) that the iXp Agent script will stay running.

For example,

```
perl C:\temp\ixpagent.pl http://venus:8081 neptune 600
```

The above command will have the iXp Agent running for 600 minutes before exiting. While the Agent is running, iXp Clients can retrieve log files from the Agent machine.

Tip: We recommend that you create two (2) jobs per Remote Agent machine in your environment. The first job performs the SQL extract and the second job runs the extracted PERL script. You can setup the second job to run after

the SUCCESS of the first job and after its own completion. By doing this, you do not need to copy the Agent file to each machine, as the first job for each machine will extract the code from the database.

Tip: If you end up creating a job that performs the SQL extract, we recommend that you create a limited database userid that has permissions to read the IXP_AGENT table. That way, exposing the database userid and password will not compromise your current security model.

4.5.4 Installing the iXp Client CLI

The iXp Client GUI is a Java application, and does not require any installation. The client machine needs to have JRE v1.5.x installed.

The iXp Client CLI is a set of commands that can be installed on any machine, including the iXp GUI clients. These commands enable authorized users to run CA Workload Automation AE commands such as autocal_asc, autorep, autostatus, jil, job_depends, sendevent from their client machines. The client machines do not need to install the CA Workload Automation AE software.

Administrators need to setup authorizations for users running these commands. This can be done from the Admin Tool.

The iXp Client CLI software is in a single file named `ixp-client-RELEASE-PLATFORM.tar` (on UNIX/LINUX) or `ixp-client-RELEASE-windows.zip` (On Windows). The software is located in the "`install`" sub-directory under the iXp root directory, on the iXp Server machine.

Create the directory where the iXp files will be stored (the iXp Root Directory on iXp Client), as defined in the *Pre-Installation checklist*. Make sure the directory can store up to 5Mb of data.

1) Log on to the Client machine and extract the files from the software image to the iXp Root Directory.

   For example, as per the sample checklist, this would be the `C:\ca\ixp` directory.

2) Edit the `IXP_HOME\etc\ixp.conf` file and add a line that provides the connection information for the iXp Server.

   The line to add is as follows:

   ```
   IXP_SERVER_URL=http://IXP_SERVER:PORT
   ```

where,

IXP_SERVER = Name or IP address of iXp Server machine
PORT = The HTTP port of the iXp Web Server

For example, as per the sample checklist,

```
IXP_SERVER_URL=0
```

If there is an alternate iXp server running for high availability, then an additional line can be added in the file to provide its connection information to the CLI.

```
IXP_SERVER_URL_2=http://IXP_SERVER_2:PORT
```

where,

IXP_SERVER_2 = Name or IP address of the alternate iXp Server machine
PORT = The HTTP port of the alternate iXp Web Server

For example,

```
IXP_SERVER_URL_2=http://mercury:8081
```

Alternatively, the above two lines can be set as environment variables `IXP_SERVER_URL` and `IXP_SERVER_URL_2`, rather than in the file. If the values are set as environment variables, then they take precedence over the lines in the file.

3) Have the user set two environment variables before executing the commands: `IXP_HOME` and `AUTOSERV`. `IXP_HOME` gets set to the Root Directory on the Client machine where the iXp CLI is installed, and `AUTOSERV` gets set to the CA Workload Automation AE Instance name for which the command is being executed. These can be set beforehand or at runtime.

For example, as per the sample checklist,

```
IXP_HOME=C:\ca\ixp
AUTOSERV=DEV
```

4) Now, to run the commands, the users can go to IXP_HOME/bin directory. More information about the commands is provided in the **iXp User Guide**.

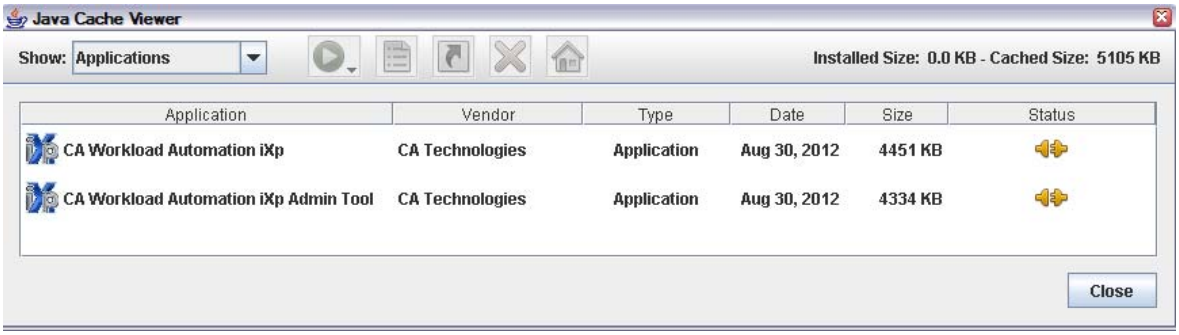
4.6 Launching the Admin Tool

You can launch the iXp Admin Tool using Java Web Start on your iXp Client machine. Pass in the appropriate URL as specified by your Pre-Install Checklist as an argument to Java Web Start. Using the working example, run the following command on your Client machine:

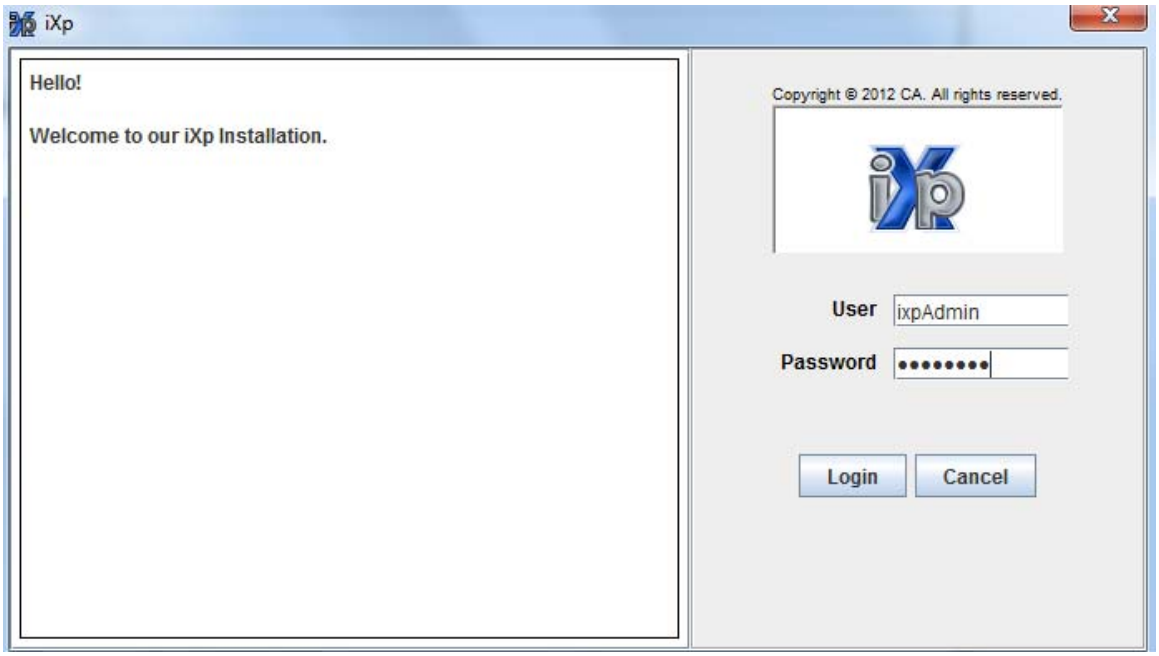
```
javaws http://venus:8080/ixp/AdminTool.jnlp
```

You will be prompted with a Security Warning. Click on `Always` and `Run` to continue launching the Admin Tool. The first time you launch the application, Java Web Start has to upload the application image, so you may experience some delay if you have a busy network. This initial upload only occurs the first time you launch an application through Java Web Start or when software upgrades occur. Once loaded you should see the login prompt.

Creating a shortcut on your desktop is recommended for launching Java Web Start. When you use Java Web Start for the first time, it prompts you to do that. After doing so, subsequent launches of iXp can be done by using the desktop shortcut or by launching the Java Application Viewer using the command "`javaws -viewer`".



At the iXp logon prompt, enter the user "`ixpAdmin`" and the default password "`ixpAdmin`". You will be prompted to accept the License Agreement and change the password for this user. Select a new password that is at least 8 characters in length, and has a special character and a numeric character.



1) To provide information about the CA Workload Automation AE Instances, to configure additional users, and setup the iXp Authentication policy, please refer to **Chapter 6: iXp ADMIN TOOL**.

2) If there are any errors during the installation, please refer to the *Installation Trouble Shooting* section for solutions.

3) Launch the iXp Application using Java Web Start. Since this is the first time you are launching the application, on the iXp Client machine run a command similar to the one for the Admin Tool.

```
javaws http://venus:8081/ixp/iXp.jnlp
```

4) Provide the login information and, once authenticated, the iXp monitoring and control application will be launched.

4.7 Post-Installation Checklist

After installation according to the above steps, the following items need to be done or verified to assure full functionality of iXp clients.

A Web browser must be in the PATH variable on the client or the location of the browser must be provided in the IXP_HOME/dat/`user/<username>/props` file	√
Need a `C:\temp` directory on the client. (It is necessary for each new iXp client.)	√
The ID that the Web Application Server runs under needs to have JIL update privilege in CA Workload Automation AE, if users are going to be performing JIL updates through iXp. Typically, that id is the same as the CA Workload Automation AE Edit SuperUser ID.	√

4.8 Installation Trouble Shooting

4.8.1 Common Installation Problems

1) `CLASSPATH` variable is incorrect in `IXP_HOME/etc/ixp.env` file.

2) `IXP_HOME` variable is incorrect.

3) Database URLs are incorrect.

4) `JAVA_OPTS` environment variable is not set correctly.

5) Database user password is incorrect.

6) Database user permissions are incorrect.

7) The Tomcat options for Java are not set correctly.

4.8.2 Pathology: Symptoms and Solutions

SYMPTOM/PROBLEM	CAUSE/SOLUTION
"`Invalid User`" error message.	Possible typo in login screen. Retype user name.
"`Invalid Password`" error message.	Possible typo in login screen. Retype password.
Web Application Server does not start iXp Daemon	`JAVA_OPTS` variable may be incorrect. Re-set the `JAVA_OPTS` variable and restart the web server.
No jobs visible within iXp application and SQL exceptions in `ixpDaemon.log` file.	Database URLs are incorrect. Correct the database URLs, as described in this chapter.
"`permission denied`" errors in `ixpDaemon.log` file.	User permissions are incorrect. Solution: correct them.
"`unauthorized access`" errors in `ixpDaemon.log` file.	Database user password is incorrect. Solution: correct it.
Java Web Start window does not show the icons for iXp or Admin Tool	There is a problem with the URL specification or the web server is not running.
English version of iXp appears on your French or Italian operating system.	Cause may be the `locale` variable is not set on your system. Solution: set the `local` variable on your system.

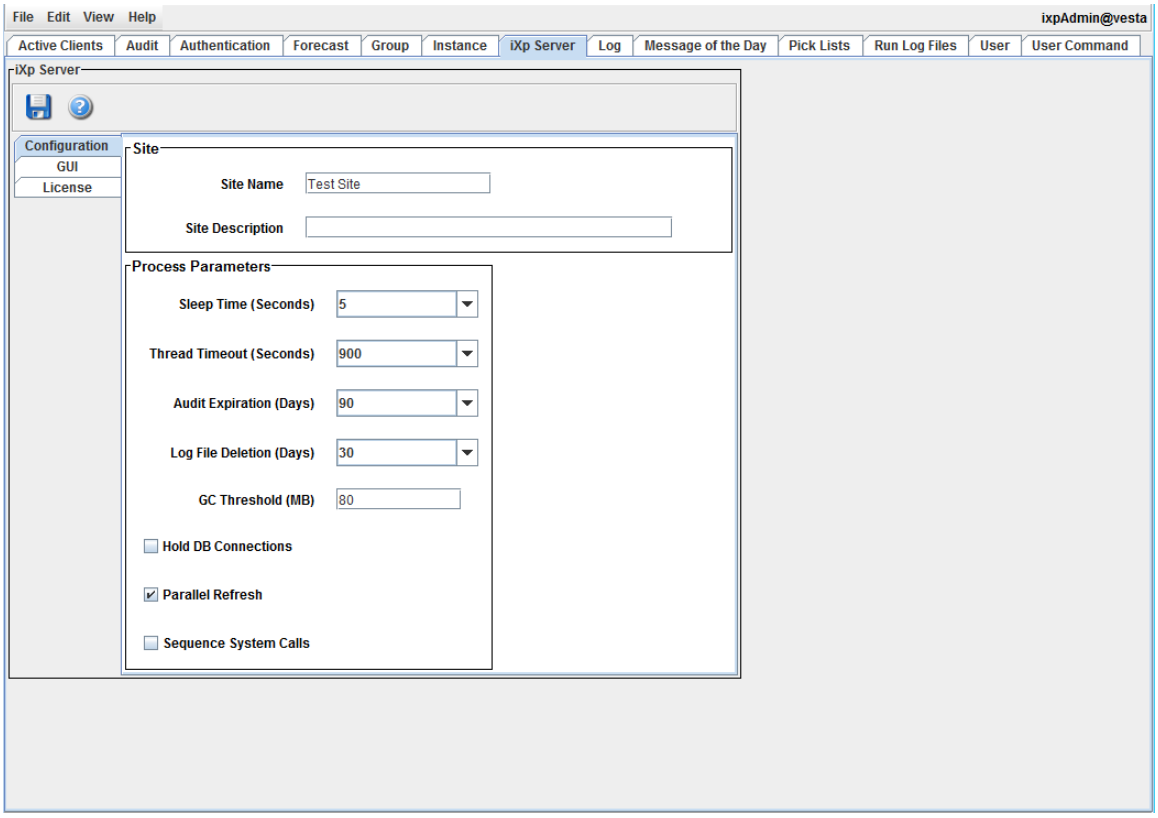
5 iXp ADMIN TOOL

The iXp Administration Tool (“Admin Tool”) allows the iXp Administrator to control and monitor security, user permissions, user commands, and connections to CA Workload Automation AE. iXp Admin Tool functions manage authentication and authorization policies for end users.

The Admin Tool can be accessed only by designated iXp Administrators.

5.1 iXp Server Settings

The *Installation* chapter should get you to the point where you can see the Administration Tool interface, looking something like this.



The tabs at the top of the Admin Tool allow the administrator to view, enter or modify information pertaining to a particular area. The data within each tab has to be saved individually. For example, clicking on the Save button on the above screen would save only the iXp Server information.

The following table explains the fields listed in the **Configuration** section. This section enables you to specify some general information, and server-level configuration parameters.

iXp Server Configuration Fields	Description
Site Name	Name of the site running iXp software. The Site name will be shown in the Tree view of the main GUI.
Site Description	Description of the iXp Server site.
Sleep Time	The time, in seconds, that the iXp Daemon will sleep after each series of data retrieval from the CA Workload Automation AE instances. The recommended value is between 1 and 10 seconds, depending upon the processing power of the iXp Server machine. The default value for this setting is five (5) seconds.
Thread Timeout	A central thread monitors all the threads of the iXp Daemon. If any of the threads has not finished its work within the value specified here, iXp will start reporting timeout errors. Setting a low value here may be too restrictive on the daemon, and a high value would not have any use. On average, no thread should take more than 5 minutes to finish its work.
Audit Expiration	iXp will maintain an online audit for the number of days specified here. After that, the audit files will be archived. The default value is 30 days.
Log File Deletion	iXp Daemon logs will be kept for the number of days specified here. After that they will be deleted. The default value is 30 days.
GC Threshold	Available memory is constantly checked and if below the Threshold, GC (Garbage Collection) is requested from the Java VM immediately. This means a high Threshold causes GC to happen more often which could degrade performance. Suggested setting for GC Threshold is about 33% or less of the Tomcat -Xmx setting.
Hold DB Connections	If this option has been selected, each of the daemon threads that reads data from its CA Workload Automation AE instance will keep the database connection open. This reduces the processing overhead on the server and the database. By default, this option is turned on. **NOTE:** We recommend turning this option OFF.

Parallel Refresh	If this option has been selected, the daemon will retrieve data from multiple CA Workload Automation AE instances concurrently. This speeds up the data refresh cycle on the server. By default, this option is selected.
Sequence System Calls	Enable this option to limit the iXp Server to perform user actions sequentially. This setting can be useful for servers with limited RAM.

The **GUI** tab controls instance monitoring and GUI idle timeout.

iXp Server GUI Attributes	**Description**
Enable instance monitor…	The instance monitor is an icon, at the bottom left corner of the GUI, which signifies the status of CA Workload Automation AE instances. The check boxes enable this feature for all users or for Administrators only.
Pause client refresh…	If a user does not perform any action in iXp, the refresh will automatically pause after the specified number of minutes. A pop-up window informs the user, and when the user clicks “OK” the automatic refresh resumes.

The **License** tab shows the iXp license attributes. The license key information received from CA Technologies, Inc. would be entered here.

iXp Server License Attributes	**Description**
License Key	The unique iXp license code generated for the following fields.
Host Name	iXp Server machine name.
IP Address	Internet Protocol address for the iXp Server.
Expiration	License expiration date.
Instances	The number of CA Workload Automation AE instances that the iXp Server will access.

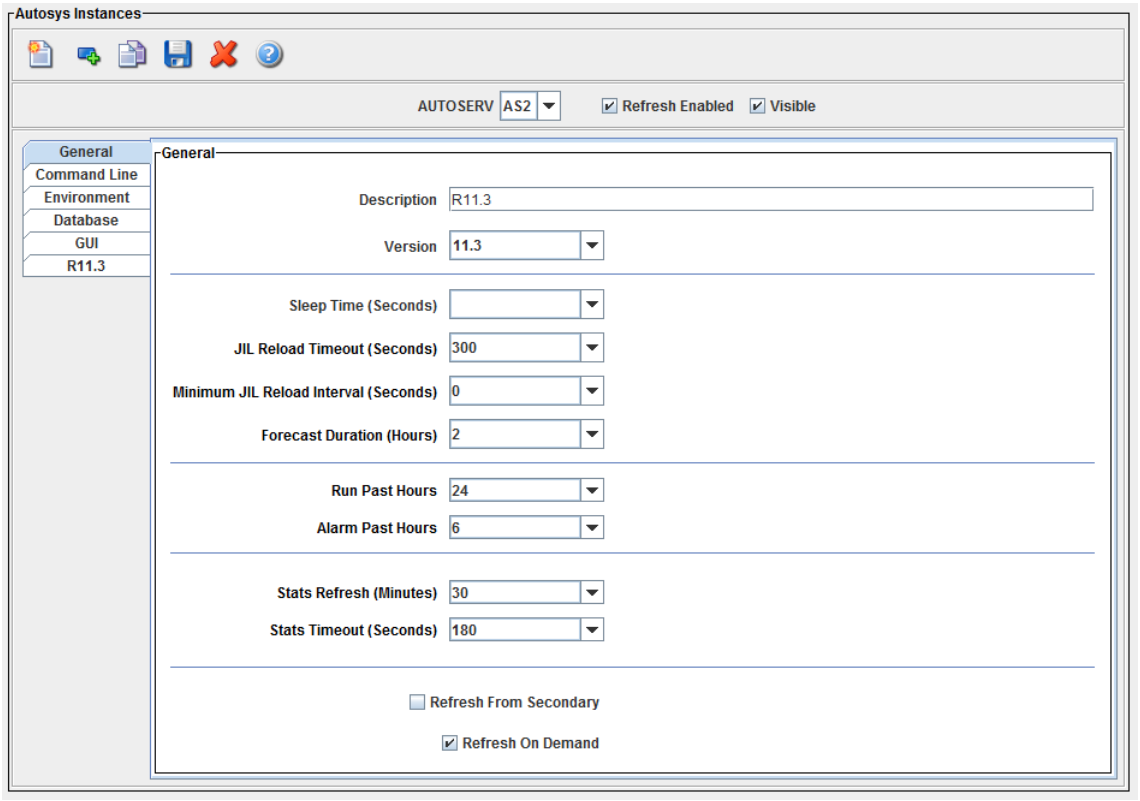
If the iXp Server is a part of a hardware cluster, or if it has the capability to roll-over to another system, in case of a system crash, then the license for the second system can be entered in the second set of license fields.

NOTE: You have to enter a valid license key in order for users other than "`ixpAdmin`" to launch the iXp GUI or CLI. Please contact CA Technologies, Inc. to obtain a valid license key. You will need to provide the iXp Server hostname, IP Address, and the number of CA Workload Automation AE instances that will be configured in iXp.

Click on the Save icon to save the configuration. The Help icon launches the Help dialog that provides similar information as above.

5.2 CA Workload Automation AE Instance Configuration

Select the **Instance** tab in the Admin Tool to enter the CA Workload Automation AE Instance configuration area.



Each CA Workload Automation AE instance to be accessed from iXp has to be set up here. The Copy Instance button can be used to create an identical AUTOSERV instance connection with a new instance name. Once copied, just the unique information should be changed.

The toolbar at the top provides the buttons to manage the data in this tab.

Tool Bar Icon	Field	Description
	New Instance	Specify the information for a CA Workload Automation AE instance not previously configured.
	New External Instance	Specify information for an MVS-based instance not previously configured.
	Copy Instance	Copy the information from a configured Instance to a new Instance.
	Save	Save the information.
	Delete	Delete the Instance configuration.
	Help	Launch the Help Dialog.

The Instance tab has 5 (five) subsections. Each section can be accessed from the menu on the left-hand side of the tab.

The following table explains the fields listed in the **General** section. This section enables you to specify how much Run Data and Alarm data will be stored on the iXp Server, and other time-out related settings.

Field	Description
AUTOSERV	The CA Workload Automation AE instance name, as denoted by the AUTOSERV environment variable.
Refresh Enabled	Used to enable or disable the iXp Daemon data retrieval process from the instance. If disabled, the iXp Daemon will not read any data from the instance until it has been enabled.
Visible	Check to make this instance appear in the GUI. If unchecked, this instance is only accessible from Admin Tool. It will not appear anywhere in the GUI. A report that runs against multiple instances will not show information for the invisible instance. A report that only references the invisible instance will give an error.
Version	The Version of CA Workload Automation AE for this

	Instance. The default is r11.3.
Sleep Time	The numbers of seconds that the iXp Daemon will be idle between data refresh cycles. The default value for this setting is Server Sleep Time – 1 (one) second.
JIL Reload Timeout	Number of seconds to wait before reload of JIL based on job time stamp changes.
Minimum JIL Reload Interval	Minimum number of seconds to wait for JIL reloads, irrespective of the amount of changes being made.
Forecast Duration	Specifies if the iXp Forecasting Engine should generate a real-time forecast of upcoming events and runs, based on the current state of the CA Workload Automation AE instance. If this value is set to non-zero, then on every Instance refresh cycle, the iXp Server will generate a forecast for the next X hours, where X is the value specified here. The default is 0.
Run Past Hours	The iXp Daemon will retrieve all the runs within the CA Workload Automation AE instance that either started, or ended in the time frame set by this field. If set to 0 (zero), the iXp Daemon will not retrieve any run data for the instance, until the value is changed. The lower value here, the faster the performance of the server.
Alarm Past Hours	If the value for this field is set to 0 (zero), the iXp Daemon will not retrieve any CA Workload Automation AE Alarms from the instance. If the value is set to any other value greater than 0 (zero), then the iXp Daemon will retrieve all the Alarms from the instance. This setting is “loosely enforced” meaning that the current setting plus 24 hours worth of alarms are retrieved.
Stats Refresh	Dictates how often the iXp Daemon retrieves and calculates the CA Workload Automation AE performance statistics for the particular instance. Typically, it is 30 to 45 minutes.
Stats Timeout	Number of seconds after which server statistics collector process will timeout.
Refresh from Secondary Connection	If the CA Workload Automation AE instance has Dual Servers, the iXp Daemon would retrieve the data from the Primary Server. If this option is selected, the iXp Daemon will retrieve data from the Secondary database. This may be desired in case of network problems or database issues.
Refresh On	If this option is selected, the iXp Daemon will try to retrieve

Demand	CA Workload Automation AE data at the most optimal point in its refresh cycle. This ensures that the iXp Cache features more recent job status information. If you have a busy iXp Server and fast-performing database servers, we recommend turning this option ON.

The following table explains the fields listed in the **Command Line** section. You can specify the Date Format that is used for this CA Workload Automation AE instance, and alternate "`jil`" and/or "`sendevent`" commands.

Field	Description
Sendevent Shell Command	By default, iXp uses "`/bin/sh`" or "`cmd /c`" when executing the `sendevent` command on the iXp server. You can specify an alternate shell command here (e.g. ssh).
JIL Shell Command	By default, iXp uses "`/bin/sh`" or "`cmd /c`" when executing the `jil` command on the iXp server. You can specify an alternate shell command here (e.g. ssh).
Sendevent Command	iXp uses "`sendevent`" as the default command to use when performing certain actions. You can specify a new command here, and iXp will pass all the sendevent parameters to the command specified here, instead of passing them to the "`sendevent`" command.
JIL Command	iXp uses "`jil`" as the default command to use when modifying, creating job and override definitions. You can specify an alternate command here, and iXp will pass the JIL file to this command instead.
Date Format	The Date Format, as specified within CA Workload Automation AE, for this instance. This field is needed for proper functioning of "`sendevent`". The value for this field must be identical to what has been specified within the CA Workload Automation AE configuration file/registry.

The following table explains the fields listed in the Environment section. You can specify the values for all the CA Workload Automation AE related environment variables here.

Field	Description
AUTOSYS	Path to AUTOSYS directory on the iXp Server. If the CA Workload Automation AE Remote Agent has not been installed on the iXp Server, leave this field blank.
AUTOUSER	Path to AUTOUSER directory on the iXp Server, if the CA Workload Automation AE Remote Agent has been installed on the machine.
SYBASE	Value of the SYBASE environment variable on the iXp Server. This is used by the CA Workload Automation AE Remote Agent to locate the database connection information.
ORACLE_HOME	Value of the ORACLE_HOME environment variable on the iXp Server.
TNS_ADMIN	Value of the TNS_ADMIN environment variable on the iXp Server. This is needed by the CA Workload Automation AE Remote Agent to locate the database connection information.

The information provided in the checklist for AUTOSYS, AUTOUSER, SYBASE, TNS_ADMIN, ORACLE_HOME, TZ environment variables would be applied to each instance of CA Workload Automation AE that is configured.

The following table explains the fields listed in the **Database** section. The fields in this section specify the connection information to the CA Workload Automation AE database.

The Primary Connection and Secondary Connection values mirror the CA Workload Automation AE convention for Primary and Secondary Event Server.

Field	Description
Database Vendor	Name of the supplier of the CA Workload Automation AE database (Oracle, SQL Server, Sybase). Oracle is the default.
JDBC Driver	Java™ DataBase Connection driver. The pull-down lists the options for Sybase, Oracle, and SQL Server drivers.
Schema Owner	The owner of the DB schema being used by CA Workload Automation AE— needed only for Oracle (typically "aedbadmin")
URL	The database connection parameters used by the JDBC driver.
User	The database user for the connection.
Password	The password for the database user.
Re-enter	Re-enter the password for verification.
DSQUERY	The name of the Sybase Dataserver or the ORACLE TNS alias.

In addition to these values, the following information about each database would be needed.

JDBC Driver

```
oracle.jdbc.driver.OracleDriver          (Oracle)
com.sybase.jdbc.SybDriver                (Sybase)
net.sourceforge.jtds.jdbc.Driver         (MS SQL Server)
```

URL

```
jdbc:oracle:thin:@HOST:PORT:SID           (Oracle)
jdbc:jTDS:sybase:HOST:PORT/DATABASE       (Sybase)
jdbc:jtds:sqlserver://HOST:PORT/DATABASE  (MS SQL Server)
```

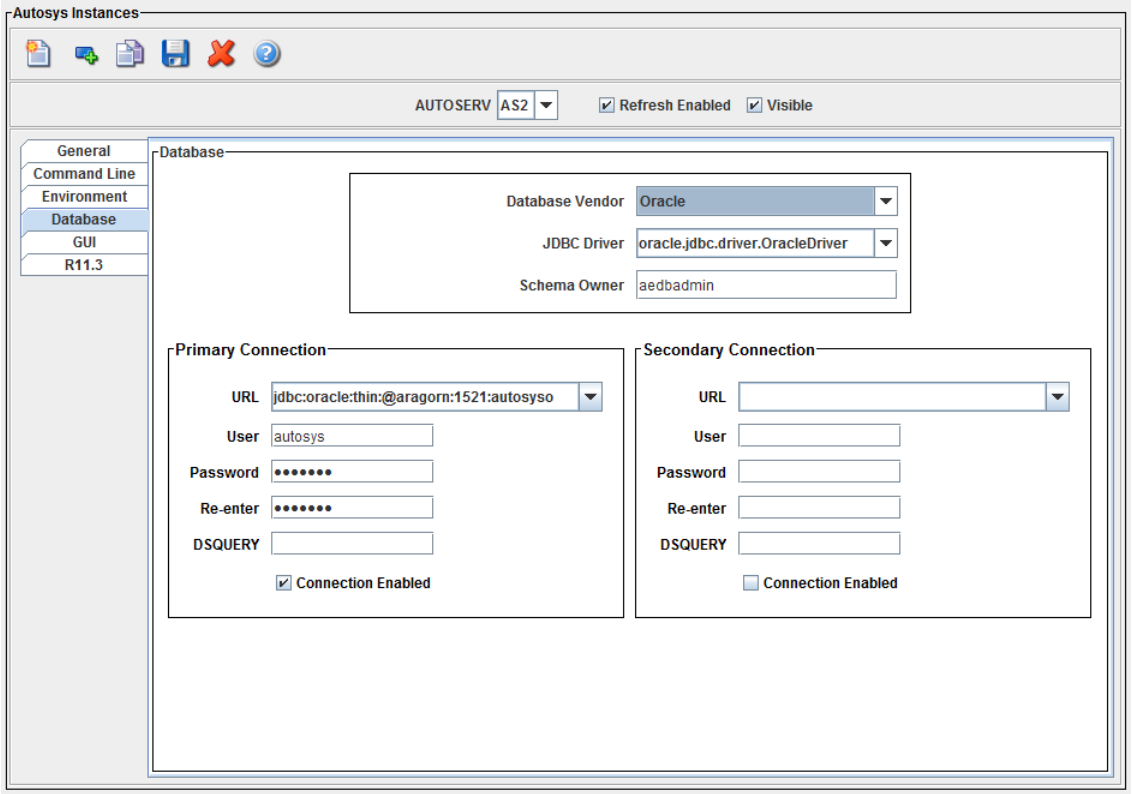
User/Password

```
DBUSER                                  (All)
DBPASS                                  (All)
```

where,

HOST =	Hostname or IP Address of the machine where the autosys RDBMS resides
PORT =	Port number for the database
SID =	The Oracle SID or Global Service Name for the Oracle instance that the database is on
DATABASE =	The name of the Sybase or MS SQL Server database where the CA Workload Automation AE data for the instance is stored. This parameter needs to be used only when the name of the database is not the same as the default database for the DBUSER.
DBUSER =	The database user that has read permissions on all CA Workload Automation AE tables and update permissions on the alarm table. Typically, this is the database user autosys.
DBPASS =	The password of the database user DBUSER.

The **Connection Enabled** check-box allows the administrator to disable a DB connection (un-check the box) when a database is down for maintenance or a rollover has occurred. If all the DBs are disabled (Connection Enabled unchecked), then the GUI Instance Status Monitor will show the instance as “Disabled”.



The following table explains the fields listed in the **GUI** section. You can use this section to customize how the CA Workload Automation AE job data for this instance is shown in the iXp GUIs.

Field	Description
Console Background	You can specify the color that will be used to highlight the jobs of this CA Workload Automation AE Instance, in the iXp Console View. The default is “No Highlight”.
Timezone	The timezone value for this instance. This value can be displayed whenever start times or end times for jobs belonging to this instance are shown.
Time Format	The format that will be used to display the dates, times, and timezone for the run data for jobs of this instance.
Current Time	Shows a sample display of date/time, for the chosen value in the “Time Format” field.
Enable Server Statistics for All Users	If this option is selected, all iXp Users will be able to view the iXp Server Statistics Graph for their instances.
Enable Server Statistics for iXp Administrators Only	If this option is selected, then only iXp Administrators will be able to view the iXp Server Statistics Graph for their instances.

NOTE: We recommend that you create all the needed CA Workload Automation AE instance information first, before creating user definitions. You have to restart the iXp Web Application Server (Tomcat or SunONE) after all the CA Workload Automation AE instances have been created.

NOTE: You also have to restart the iXp Web Application Server if you configure a new CA Workload Automation AE instance.

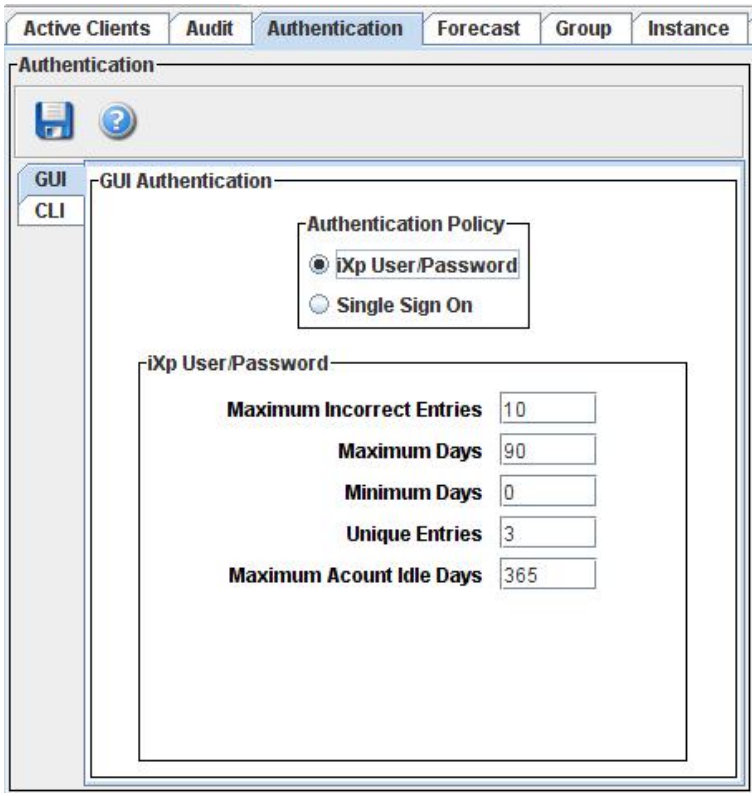
The following table explains the fields listed on the **r11.3** tab.

Field	Description
Only Permit…	When checked, users are only allowed to define Unicenter AutoSys JM 4.5 job types through ixjil (from the command line) and File►CLI►jil.

5.3 User Authentication

Select the **Authentication** tab at the top of the iXp Admin Tool window to configure the authentication policy for iXp. You have to specify the user authentication model for users launching the Client GUI.

By default, iXp is configured to authenticate users based on internal iXp user and password storage. The other option is to use Single Sign-On based authentication.



From this tab, you can also enable the iXp Client CLI and specify the authentication model for users executing the commands.

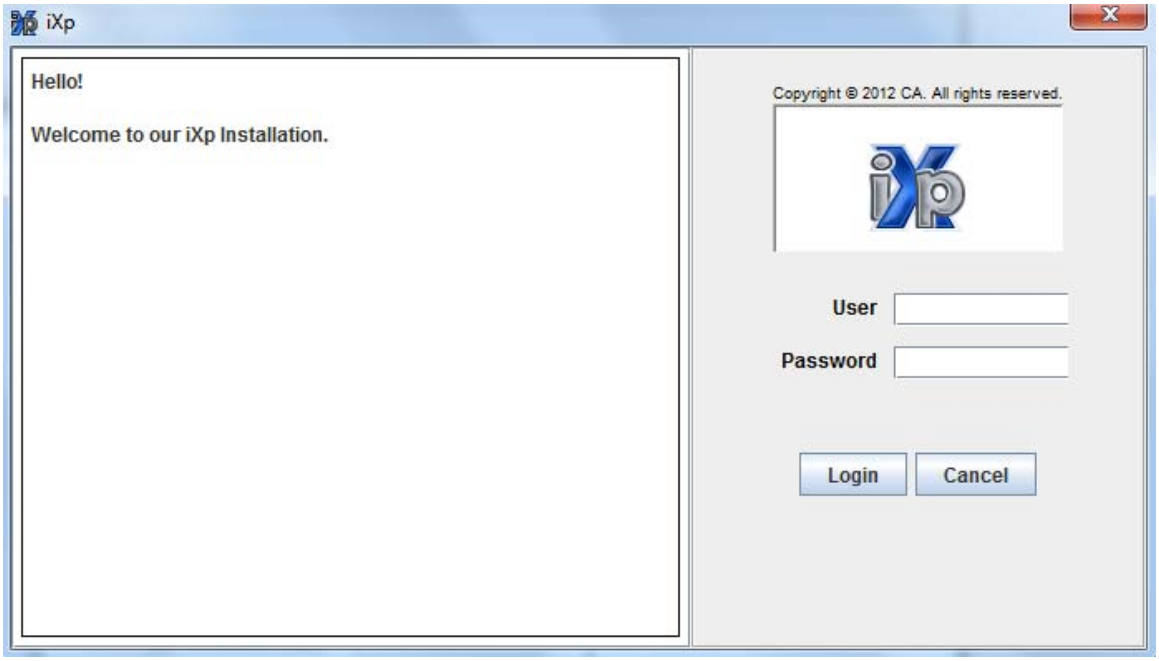
5.3.1 iXp User/Password

This is the default configuration for users launching the iXp GUI. Each user has to be setup with a password within iXp. The user name is of your choice, though we recommend using names that already exist for users within your domain. You can specify the password policies and other account settings here if you select this option.

The following table provides details on the fields for which information has to be provided.

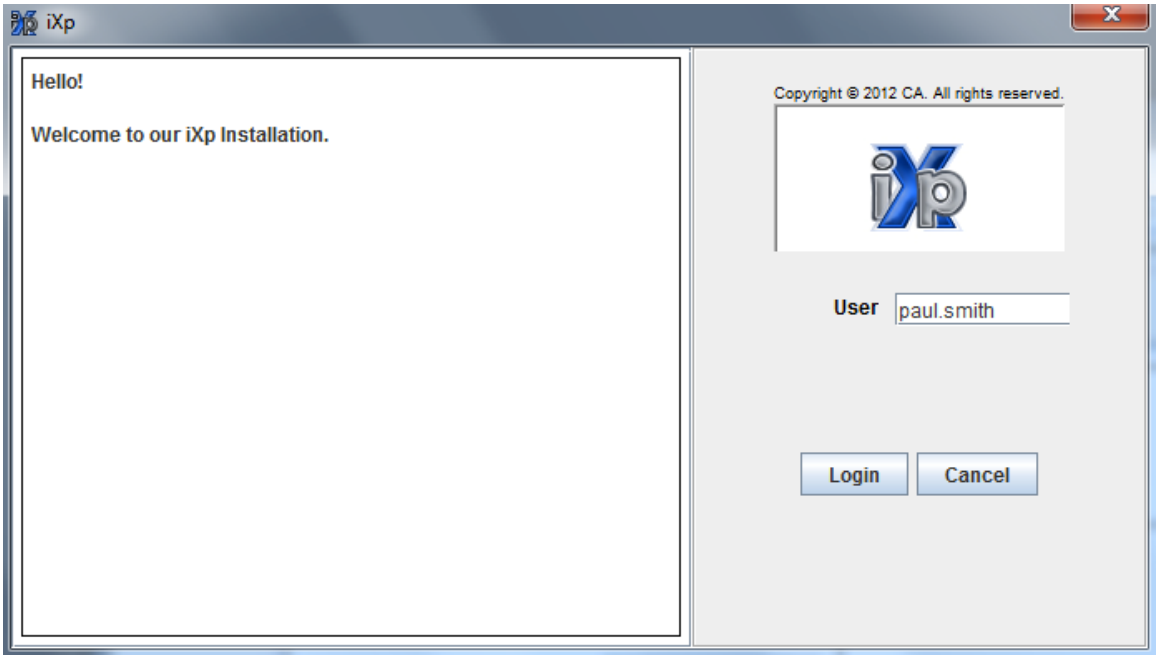
Field	Value
Maximum Incorrect Entries	The maximum number of incorrect password entries allowed before the account is automatically Suspended. The default value is 1000 tries.
Maximum Days	The number of days after which the password for any user has to be reset.
Minimum Days	After setting a password, users cannot change that password for the number of days specified here. Admins can still change the password from the Admin Tool. The default value is 0 days.
Unique Entries	A new password cannot match the last X number of passwords. The default value is 3.
Maximum Account Idle Days	If an iXp User account does not have any user or Admin activity for the number of days specified here, the account will be automatically Suspended by the iXp Daemon. The default value is 365 days.

When you choose this authentication model, users have to enter a user name and password every time they launch the GUI.



5.3.2 Single Sign-On

If you choose this method of authentication, users do not have to enter a username and password when they launch the GUI. The iXp Client GUI will recognize users based on their Domain or OS authentication.



Under this method, you can provide a list of valid IP Addresses, Host names, and Windows Domain names. Users that are not authenticated to the valid

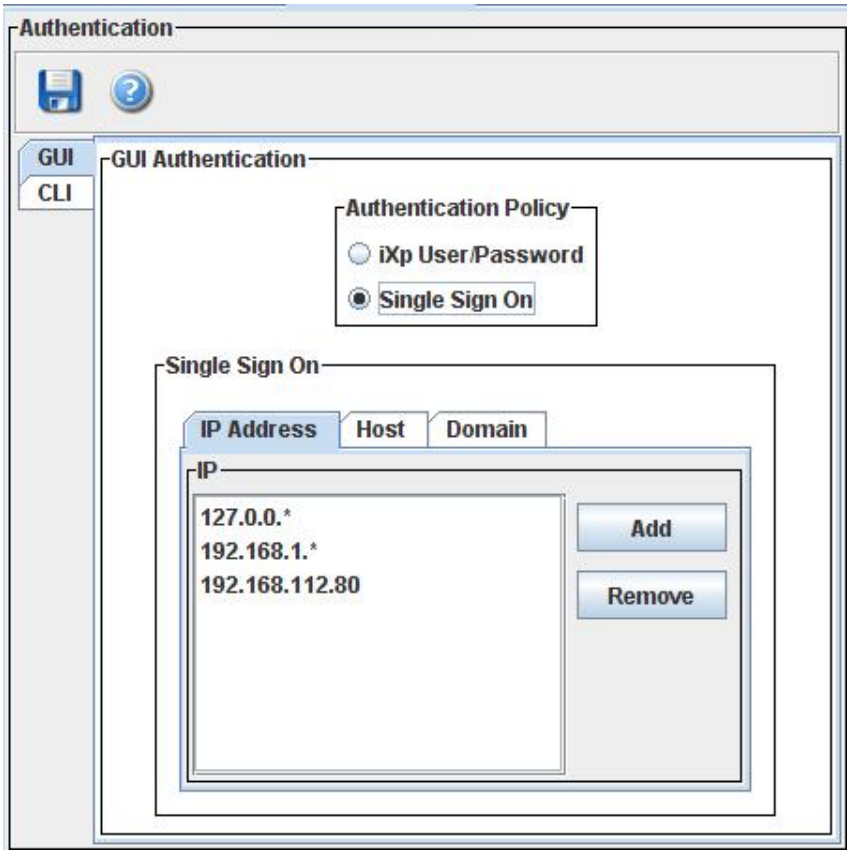
Domain, or whose iXp Client machines are not within the list of valid IP Addresses **or** host names, will be unable to launch the GUI.

5.3.2.1 IP Address

You have to provide a list of IP Address strings that will cover all valid iXp Client machines.

You can provide multiple lists of valid IP Addresses, and you can use wildcards or ranges for any of the values. For example,

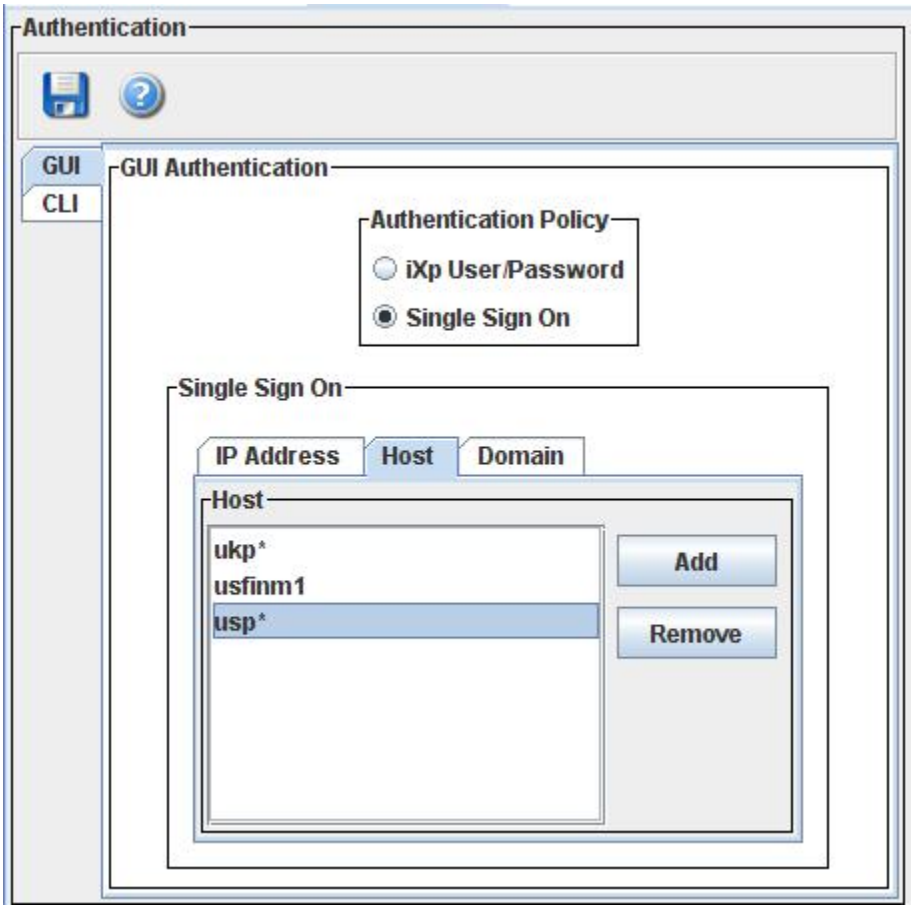
```
192.*.*.*
19.68.5.10-200
10.1.86.*
*
```



If an iXp Client machine matches any one of the specified IP Address lists, then only it will pass the IP Address check.

5.3.2.2 Host Names

You have to provide a list of host name strings that will cover all valid iXp Client machines. The list of host names is needed along with the list of IP Addresses. A valid iXp Client machine has to pass both the checks.



You can provide multiple lists of valid host names and you can use wildcards for any values. For example,

```
ukp*
usp*
usfinm1
*
```

If the host name of an iXp Client machine matches any of the specified host name lists, it will pass the host name check.

5.3.2.3 Domain

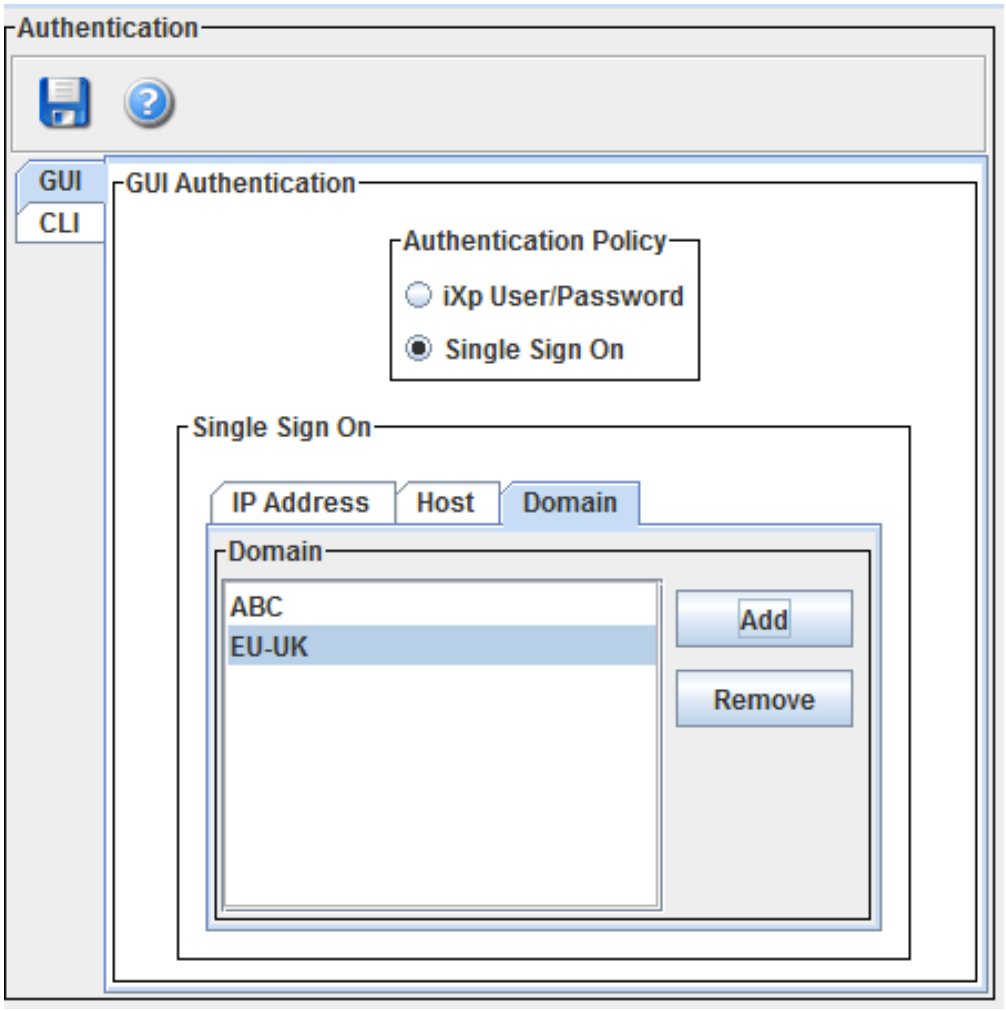
This field is used to validate the Domain or OS authentication of a user. You have to provide a list of Domains or host names that will cover all valid user authentications.

If a user is logging on to a Windows Active Directory, NT Domain, or local Windows machine, you can provide the names of valid Domains and local machine names. If a user logs on to a Domain not listed, or logs on to the local machine not listed, then the user will not be authorized to launch the iXp GUI.

If a user is logging on to a UNIX/LINUX machine, then you have to provide the names of valid machines. In this case, the list of machines could match the list of Host names provided earlier.

For example,

```
CA
us*
*
```



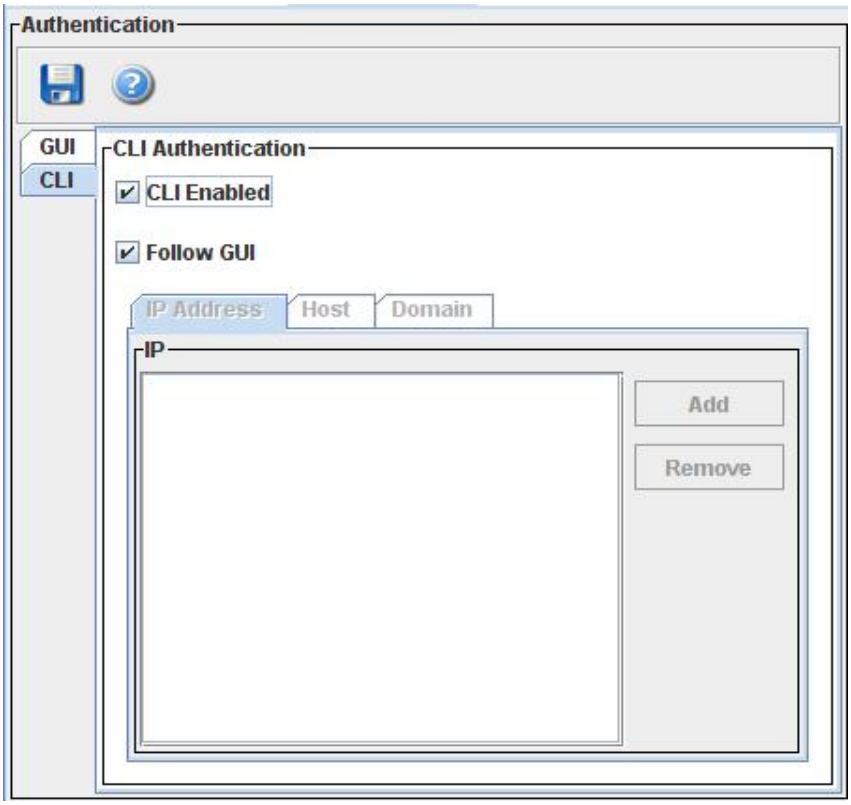
Once a user has logged on from a valid client, and has authenticated to a valid Domain or machine, the user will be able to launch the iXp GUI. The user will not be asked to provide a password, since the authentication has passed all checks.

5.3.3 iXp CLI Authentication

By default, the iXp Client CLI is not enabled. Users can install the CLI, but will not be able to use the commands until it has been enabled here in the Admin Tool. Also, the CLI must be enabled to allow users to execute command directly from the iXp GUI (**File►CLI**).

When you enable the CLI, you have to specify the same authentication fields as you did for the GUI, if you chose the Single Sign-On option for the GUI. For the CLI, there is no option to choose iXp User/Password based authentication.

If you chose the Single Sign-On authentication option for the GUI, you can elect to follow the values specified for the GUI authentication. In that case, the CLI has to be launched from machines that are valid for the GUI and has to be launched by users authenticated to the same Domains and hosts specified for the GUI.



If you do not want to follow the same settings as the GUI, then you have to specify the list of valid IP Addresses, Host names, and Domains. The same options exist here as they do for the GUI.

Please refer to **Section 6.3 User Authentication** for more information.

Once the user authentication has been setup, you now have to setup authorizations for each valid user.

Without valid authorization policies, users will not be able to launch the GUI or CLI, even after successful authentication.

5.4 Users and Groups

Once the authentication policy has been defined, privileges have to be assigned to users. Users that do not have any privileges will not be able to launch the iXp GUI or the CLI.

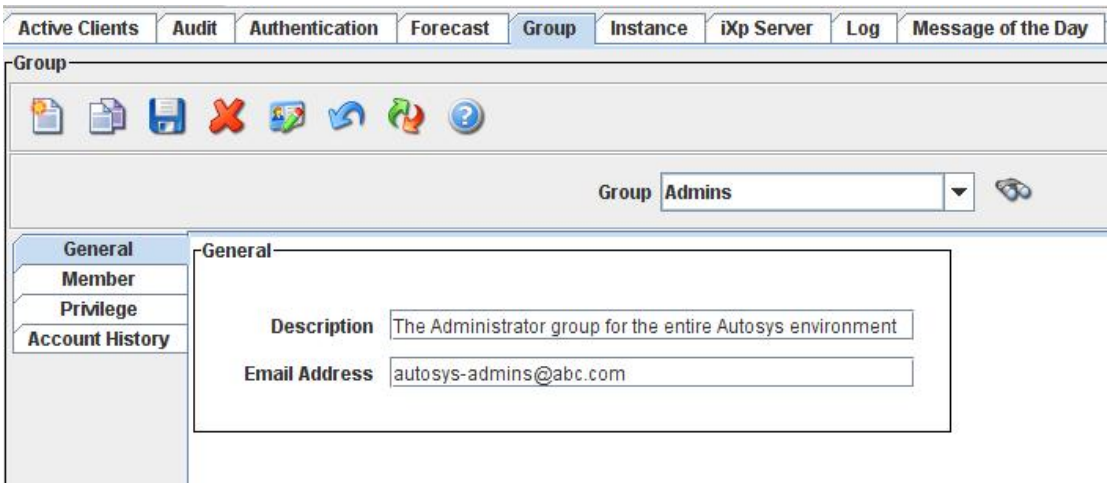
You can assign privileges to users in two ways:

1. Assign privileges to Groups and assign Group memberships to each user.
2. Assign privileges directly to users, with or without Group memberships. These privileges would be in addition to any other privileges the user inherits from its group membership.

5.4.1 Groups

You can create Groups and assign privileges to each group. Users can be members of multiple groups. The privileges for each user will be the combined privileges of all of their groups.

Select the **Group** tab at the top of the iXp Admin Tool window to maintain group definitions.



To find an existing Group, you can select it by using the pull-down list in the Group name area, or click on the Search button to launch a wild-card based Search dialog.

 To create a new Group, click on the **New Group** icon on the toolbar.

 Click on the Copy Group icon to copy an existing group, including its memberships and privileges.

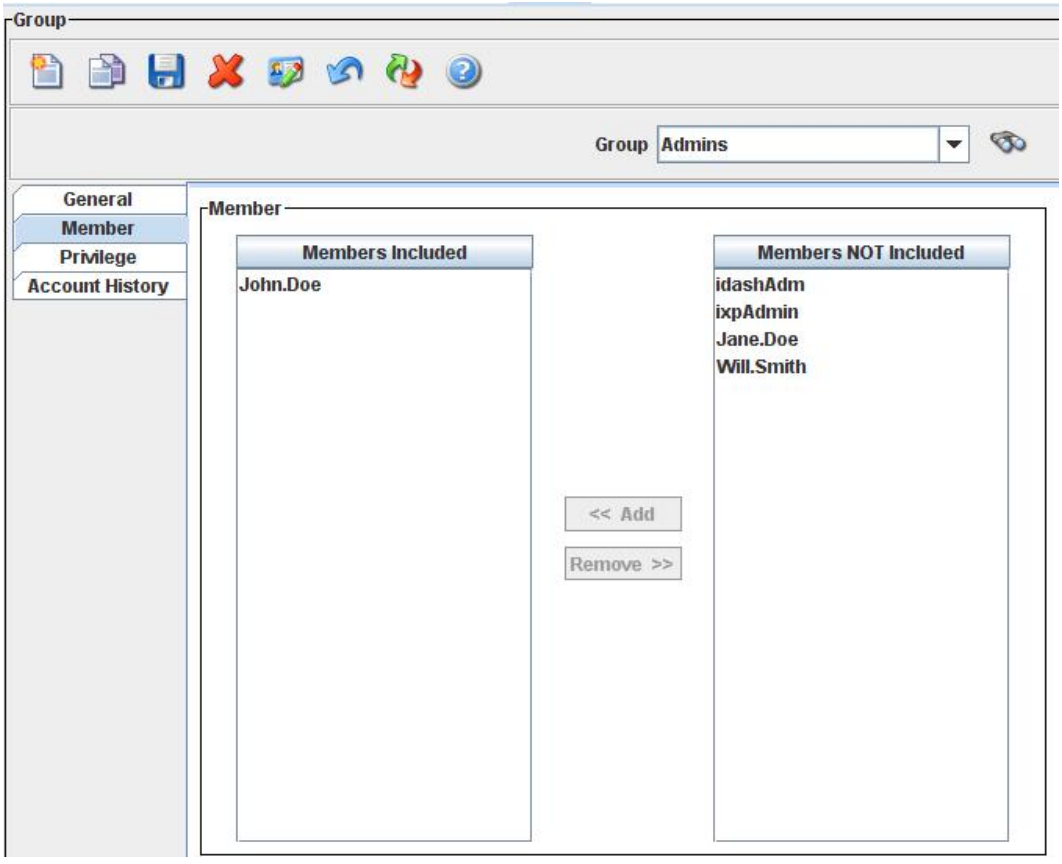
 Click on the Rename Group icon to rename an existing group.

Enter the desired name for the group. The names cannot include any spaces.

The left-hand side shows multiple tabs. The first tab “General” provides an area to enter a description and an email address for the group. This information is strictly for viewing only.

5.4.1.1 Member

This tab enables you to assign membership to this group.



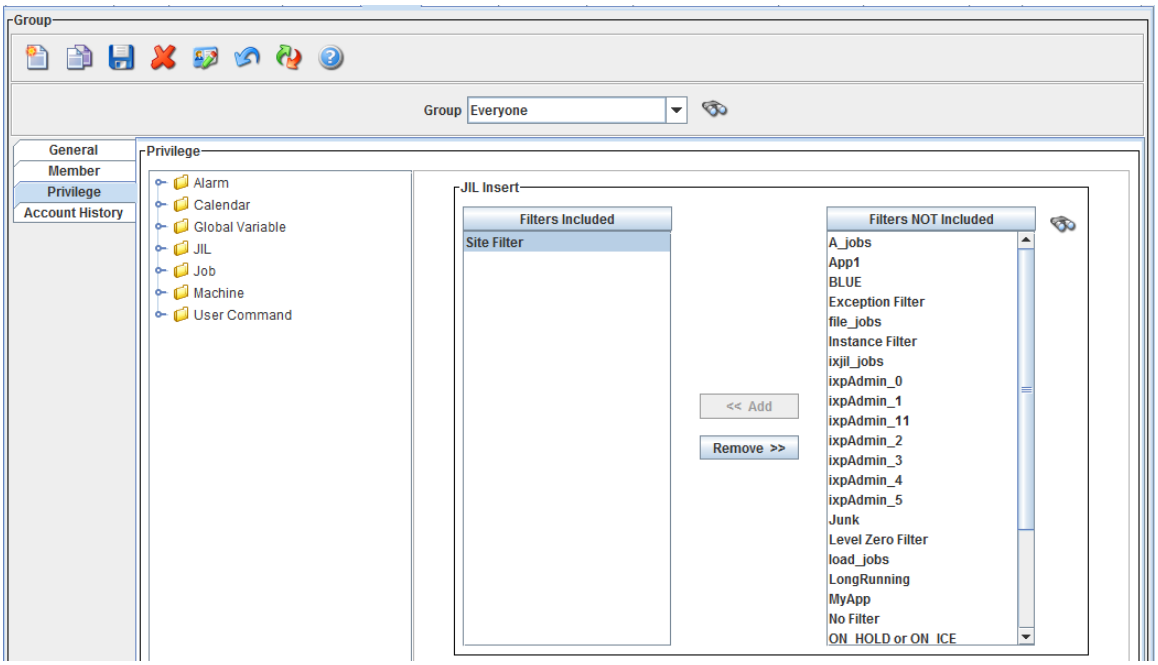
To add members, select the users from the “Members NOT Included” area and click on the “Add” button.

To remove members, select the users from the “Members Included” area and click on the “Remove” button.


You can click on the “Save” button at any time to save the current Group definition.

5.4.1.2 Privilege

This is the tab where you assign the privileges for this Group. All members of this group will then inherit these privileges. When you change the privileges for any group, all the members will immediately get the new set of privileges.



Note the search icon  beside the list of Filters NOT Included. This can be used to find a particular filter by name.

Each Group can be assigned different levels of authorizations, depending upon its function. By default, each Group gets created with “read-only” privileges on every CA Workload Automation AE job.

The privileges are classified in to multiple categories: Alarm, Calendar, Global Variables, JIL, Jobs, and User Command. You can expand each category and see the individual privileges.

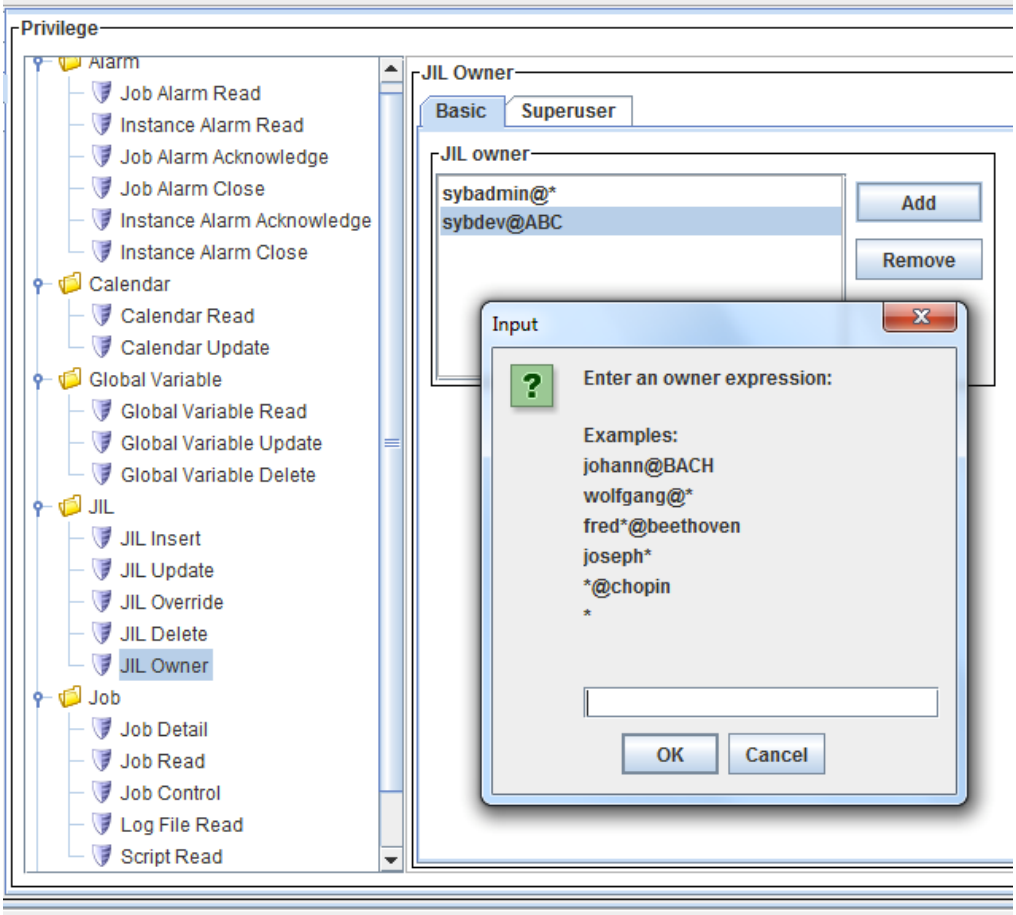
The following table explains all of the categories and their individual privileges.

Category	Privilege	For the Jobs matching the filters, Users can
Alarm	Job Alarm Read	✓ See the Alarm Dialog ✓ View the job-related CA Workload Automation AE Alarms subject to Job Detail privilege
	Instance Alarm Read	✓ See the Alarm Dialog ✓ View the non-job related CA Workload Automation AE Alarms for instances covered by the Job Read privilege.
	Job Alarm Acknowledge	✓ Acknowledge the CA Workload Automation AE Alarms in the iXp Alarm Dialog.
	Job Alarm Close	✓ Close the CA Workload Automation AE Alarms in the iXp Alarm Dialog.
	Instance Alarm Acknowledge	✓ Acknowledge non-job related CA Workload Automation AE Alarms.
	Instance Alarm Close	✓ Close non-job related CA Workload Automation AE Alarms.
Calendar	Calendar Read	✓ View all the Calendars for the Instance(s) included which match the given expressions.
	Calendar Update	✓ Create, or modify any of the Calendars for the Instance(s) included which match the given expressions.
Global Variable	Global Variable Read	✓ Change any Global Variables for the Instance(s) included which match the given expression(s).
	Global Variable Update	✓ Change any Global Variables for the Instance(s) included which match the given expression(s).
	Global Variable Delete	✓ Delete any Global Variables for the Instance(s) included which match the given expression(s).

JIL	JIL Insert	✓ Create new jobs from the iXp GUI using Job Editor, or from the CLI using “ixjil”.
	JIL Update	✓ Update jobs using the iXp GUI or CLI.
	JIL Override	✓ Issue one-time overrides using the iXp GUI or the CLI. ✓ Delete one-time overrides using the iXp GUI or CLI.
	JIL Delete	✓ Delete jobs using the iXp CLI.
	JIL Owner	✓ Assign a Job Owner when creating or updating jobs using the iXp GUI or CLI.
Job	Job Detail	✓ View and search job attributes such as command, owner, description. ✓ Also controls Log read and Alarm read access.
	Job Read	✓ View Jobs and their statuses, details, run history, alarms ✓ View One-time Overrides ✓ Generate Historical and Forecast Run Reports.
	Job Control	✓ Issue any `sendevent`.
	Log File Read	✓ View Job Log, Job Profile, Remote Agent log files subject to Job Detail privilege
	Script Read	✓ View Command Script file specified in the selected job’s `command` attribute.
	Machine Edit	✓ Change or Delete any Machine for the Instance(s) included which match the given expression(s).
Machine	Machine Use	✓ Use any Machine for the Instance(s) included and which match the given expression(s) in a job definition “machine: field.
	Job Command	✓ Execute pre-defined Job User Commands.

User Command	Alarm Command	✓ Execute pre-defined Alarm User Commands.
	Context Free Command	✓ Execute pre-defined Context-Free User Commands.
		✓

NOTE: In the Job Owner area, you can provide a list of owners that are allowed for each user. You can use wildcards (*) in the owner names. If you specify just a wildcard, then the user will be allowed to specify any owner, other than "`root`". If you want to allow the user to specify "`root`" as a job owner, go to the "Superuser" tab and enable that privilege.



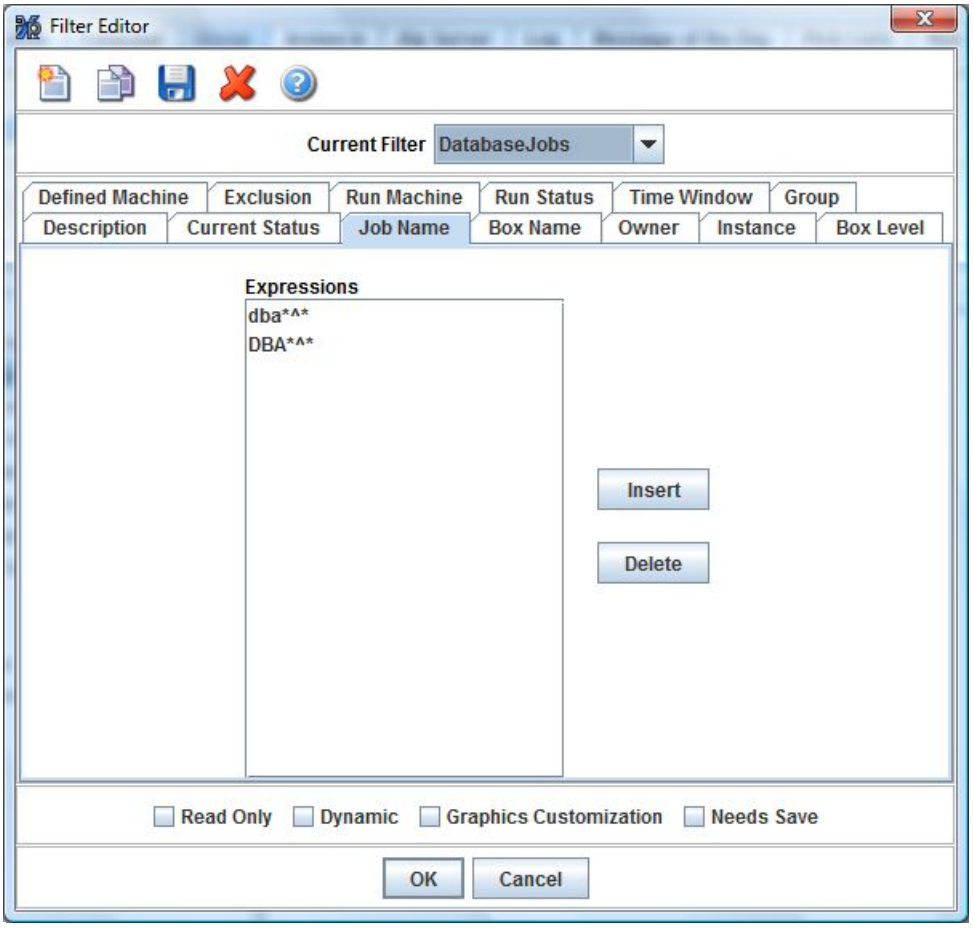
Most of the privileges must be allocated a list of jobs. These lists are called "Filters" in iXp. The iXp Administrator can assign any out-of-the-box filters provided with iXp, or create new filters and assign them. Users will be able to leverage their privileges only on those jobs that pass through the filters assigned for that privilege.

5.4.1.3 Using Filters for Security

Filters are used to create authorizations. Only the filters created by users with "Admin" privileges and the filters provided standard with iXp can be used for this purpose.

Each filter can include wildcards for any of the job attributes. The iXp Daemon dynamically generates a list of jobs based upon the values assigned to each field in the filter definition.

To create new filters, go to **Edit►Edit Filters** and the Filter Editor will be launched.



 Click on the New Filter icon to create a new filter. When you type in the desired name for the filter, please remember that the name cannot include any spaces.

You can create a filter that includes jobs based on values for one or more attributes listed in this dialog. By default, each attribute will be ignored unless you explicitly provide a list of values.

The following table describes all the fields that can be used for a filter definition.

Filter Attribute	Description
Description	A text description for the filter being created. This is not the same as the description attribute of jobs.
Current Status	Only jobs with Statuses that are selected by a check mark will satisfy the filter. If no statuses are checked, no jobs will pass the filter. By default, all current statuses are selected.
Job Name	Each list entry can be a `job_name` like “`Job_A`” or a wildcard expression like “`Job_A*`”. Note that "^" is a special character for delimiting instances. “`Job_A`” is equivalent to “`Job_A^*`”. This list will be applied to all types of jobs (Box, Command, and File Watcher).
Box Name	Each list entry can be the name of a Box Job. The name supports wildcard expressions. The filter will return all Box Jobs that match the name expressions AND all the jobs within them, including other box jobs.
Owner	Only jobs with `owner` that match these expressions will pass the filter. Entries may be a string like “`root@pluto`” or a wildcard expression like “`root@*`”.
Instance	CA Workload Automation AE instance names, like “`ACE`”, or wildcard expressions like “`A*`”.
Box Level	Box Level is the maximum number of box levels that will satisfy the filter. All jobs at a higher box level will be excluded.
Defined Machine	Each list entry can be a string like “`pluto`” or a wildcard expression like “`jup*`".A job satisfies the filter if it has at least one machine in its JIL "`machine:`" definition that matches any of the filter's wildcard expressions.
Exclusion	Each list entry can be a `job_name` like “`Job_A`” or a wildcard expression like “`Job_A*`”. Jobs that match the names listed here or are within a Box job listed here will not pass through the filter.
Run Machine	Each list entry can be a string like “`pluto`” or a wildcard expression like “`pl*`”.Only those jobs that have at least one execution on the machines listed here will pass through the filter. Only jobs having one or more historical run records on a run machine (or virtual machine) will pass the filter.

Run Status	Works in conjunction with the **Time Window** tab. Only those jobs that have historical or forecasted runs with a status selected here AND within the time frame selected in the Time Window tab will pass through the filter. By default, all run statuses are selected.
Time Window	"Future Hours" and "Past Hours" set bounds on the Run Status filter tab. Choosing "Ignore" for Past Hours means no recent run history is required to pass the filter. Choosing "Ignore" for Future Hours means no forecasted execution is required to pass the filter.
Group	A filter can be assigned to one or more groups. When a filter gets assigned to groups, only the members of that group (and Admins) will be able to see the filter in their Shared Filter list. If a group has no members and the filter gets assigned to that group, only the Admins will be able to see that filter. Group members can use the filter, but not edit it. If a user is removed, all of that user's filters will be deleted, including group filters.

 You can also copy an existing filter; change some of its properties, and save it under a new name.

If any of the fields in the filter definition are left blank, those fields will be ignored when passing jobs through the filter. So, for example, if the Job Name field is blank, all job names will pass through the filter.

NOTE: When you assign filters to the "**JIL Insert**" or "**JIL Delete**" privilege, only the Job Name and Instance attributes are used. All other attributes are ignored. If the Job Name field is left blank, these privileges will be available on ALL jobs. You have to specify a list of values in the Job Name field in order to restrict this privilege.

Each filter also has some "informational" check boxes shown at the bottom of the Filter Editor dialog. These are for information only and cannot be changed by the user creating the filter.

Check Box	Description
Read Only	Indicates this filter is not owned by the current user and/or cannot be modified.
Dynamic	Indicates that jobs that will pass through this filter will depend upon their job run information (e.g. Current Status)
Graphics Configuration	Indicates additional drill-down or drag-n-drop actions are associated with this filter.

Needs Save	Indicates that filter configuration changes have not yet been saved to the iXp Daemon.

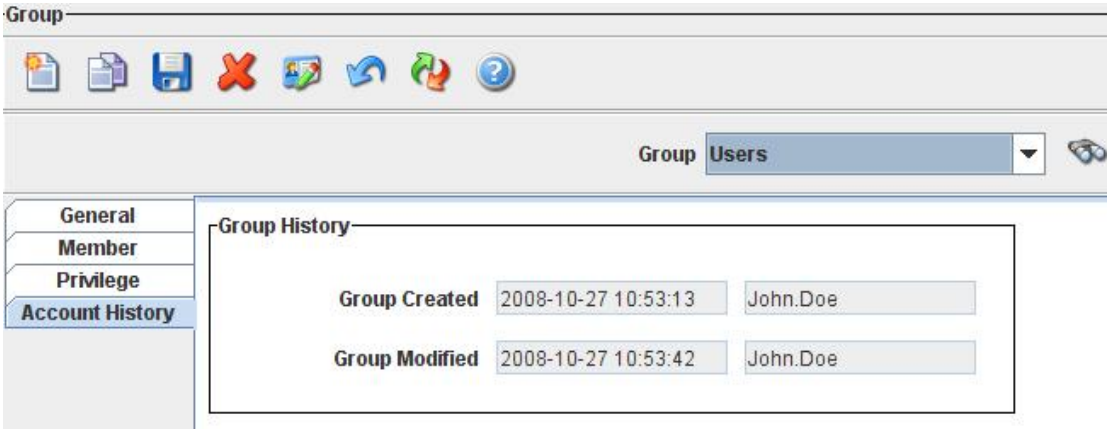
 Once you have created a filter as per your specifications, click on the icon to save the filter definition. Only after saving the filter, can it be used for privileges.

 After saving a new filter, click on the Reload Filters icon on the Group toolbar. This will import the newly saved filter in the list of filters available for assignment to privileges.

 You can delete any of the existing filters by selecting the filter in the Filter Editor and clicking on the delete icon.

5.4.1.4 Account History

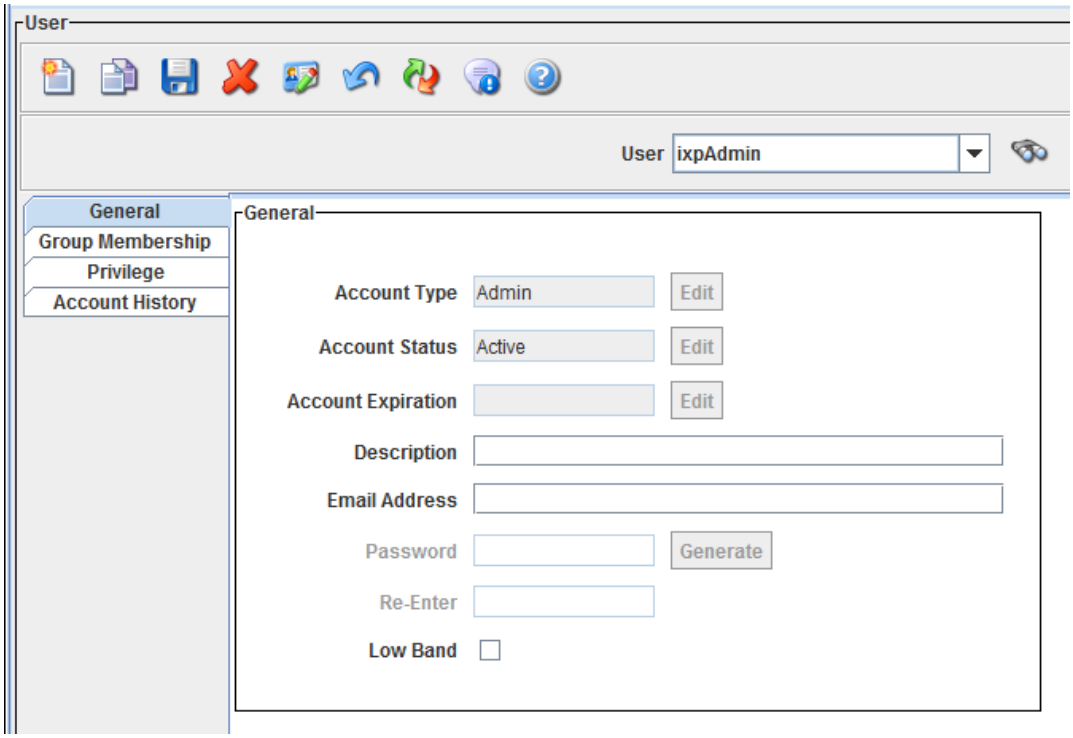
This tab shows creation and modification information about the Group.



Once you have created one or more Groups, you can now create users and assign group memberships to them.

5.4.2 Users

Select the **User** tab to create users and their group memberships and/or privileges. Once users pass the Authentication tests, they can launch the iXp Client GUI or the CLI **only** if they have been defined here.



If you have chosen the iXp User and Password Authentication model, you have to create user names and passwords in this tab, define the user type, and assign group memberships and/or privileges to them.

If you have chosen the Single Sign-On Authentication model, you just have to define the user type, and assign group memberships and/or privileges. **The user name must match the Active Directory or OS user name. The user names are not case-sensitive if the iXp Server is running on Windows. If the iXp server is running on UNIX/Linux, then we recommend using lowercase for all usernames in iXp.**

To find an existing User, you can select it by using the pull-down list in the User name area, or click on the Search button to launch a wild-card based Search dialog.


To create a new User, click on the **New User** button on the toolbar. Enter the name for the user as per the specifications. The names cannot include any spaces. The left-hand side shows multiple tabs.

5.4.2.1 General

This tab enables you to assign user type and some general information. The following table provides details about the fields in this tab.

User Field	Description
Account Type	An iXp user can be of the following types: Regular, Admin, or Temporary. Click on the Edit button to set the Account type. An Admin user will be able to access the Admin Tool and perform all the functions available here. A Temporary user will be like a Regular user, except that the userid will Expire after the account expiration date has been reached.
Account Status	Indicates whether the account is Active or Suspended. If an account is Suspended, it cannot be used for launching the iXp Client or the Admin Tool. To Suspend an account, click on Edit and set the status to Suspended. To re-activate a Suspended account, click on Edit and set the status to Active.
Account Expiration	The number of days after which the selected Temporary userid will Expire. Admins can re-activate Expired accounts by changing the Expiration Date. This field is available only for Temporary accounts.
Description	A text description for the username.
Email Address	You can use this to store any characteristics of the user, for example the email address.
Password / Re-enter	The password for the iXp username. This is required only if the GUI authentication is set to iXp User/Password. Passwords have to be at-least 8 characters in length and need to contain at-least one numerical character and one special character. A password supplied here is temporary, and users have to change their password at the next login time.
Generate	Use this button if you would like iXp to create a temporary password. The temporary password will be shown in a pop-up dialog. You can highlight the text and copy it using standard copy-and-paste operation.

User Field	Description
Low Band	You can assign this setting to users that will connect to the iXp Server over a slow network connection. When you enable this setting, the iXp Client receives lesser amount of data upon startup and on every refresh.

 To set the above fields, click on the Save icon after creating/modifying each user account.

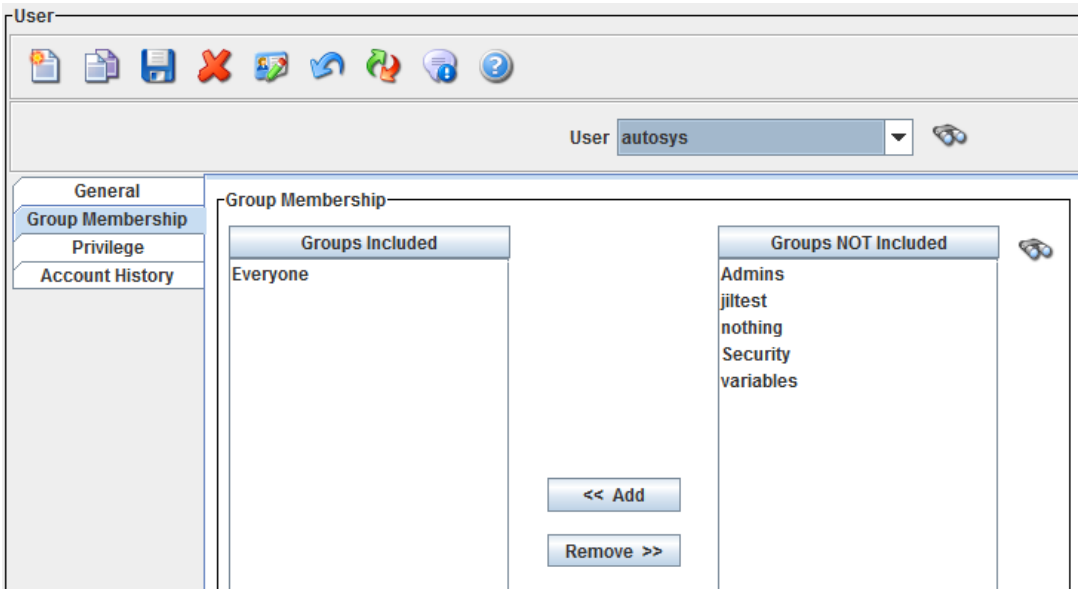
5.4.2.2 Group Membership

This tab enables you to assign group memberships to the selected user.

To add this user to a Group, select the Group from the “Groups NOT Included” area and click on the “Add” button.

To remove this user from a Group, select the Group from the “Groups Included” area and click on the “Remove” button.

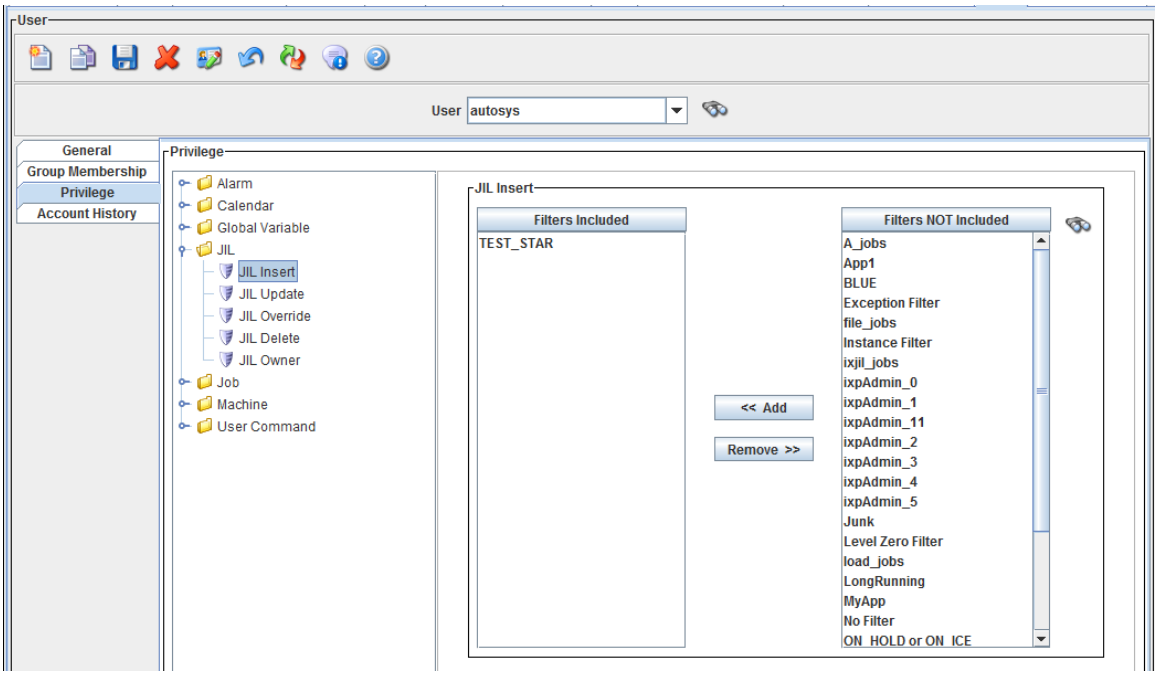
You can click on the “Save” icon at any time to save the current User definition.



When a user is a member of multiple groups, the user inherits the combined privileges of all those groups.

5.4.2.3 Privilege

This tab enables you to assign specific privileges to users. These privileges will be given to the user, in addition to the privileges that the user inherits from its group membership. This is recommended if the user needs to be assigned some privileges temporarily, or if the user is fulfilling some unique responsibilities not assigned to the rest of the group members.



Note the search icon  beside the list of Filters NOT Included. This can be used to find a particular filter by name.

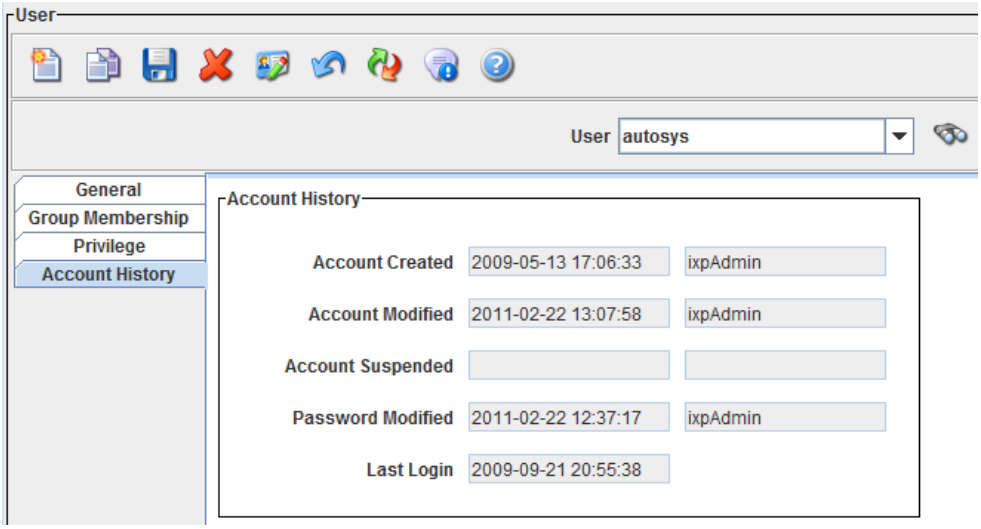
The privileges are identical to those at the group level. To get more details about this tab, please refer to the **Section 6.4.1.2 Privilege** under the **Groups** area.

5.4.2.4 Account History

The following table lists the informational fields shown in this tab for the selected user. This tab shows limited data when in SSO (Single Sign-On) mode.

Informational Field	Description
Account Created	The Date and time when an account was created.

Informational Field	Description
Account Modified	The Date and time when an account was modified, either by an Admin or by the iXp Daemon.
Account Suspended	The Date and time when an account was Suspended, either by an Admin or by the iXp Daemon.
Password Modified	The Date and time when the password was last modified, either by an Admin or by the user.
Last Login	The Date and time of the last login by the user.



5.4.2.5 Create a New User

1) Click on New User icon on the toolbar.

2) Type in the new **User Name** in the input box, and click **OK**.

3) If you are using iXp User/Password authentication, enter the new user **Password**, and re-enter for verification. This password will be temporary only. The user will be forced to change the password at the time of initial login. If you would like iXp to create a temporary password, click on "**Auto PWD**" button next to the password field. The temporary password will be shown in a pop-up dialog. You can select the text of the password and copy it using standard OS text copy-and-paste buttons (e.g. Ctrl-C on Windows).

4) Enter a **Description** of the user, if desired.

5) Enter an **Email Address** for the user, if desired.

6) By default, the user type is set to "**Regular**". If you wish to assign iXp Administrator privileges to this user, click on the Edit button and select "**Admin**" as account type. If the user id is a temporary id, click on the Edit button and select "**Temporary**" as account type.

7) If the User is of the type "**Temporary**", then click on the Edit button for **Account Expiration** and assign the number of days for which the user will be valid.

8) Assign the Group memberships for this user. Optionally, assign specific privileges to the user.

9) Click on the **Save** icon to save the new user definition. Now, the saved user can log on to the iXp GUI.

5.4.2.6 Copy User

 Use the Copy User icon to create a clone of the selected user's Group memberships and the properties defined in the ID box. Adjust the password and other identity and security properties as needed.

5.4.2.7 Edit User

1) Select the user from the User Name drill-down list.

2) Follow steps 3-9 of *Create a New User*.

5.4.2.8 Remove/Rename User

 To remove a user, select the user from the User Name drill-down list and click on the **Remove** icon.

 To rename any user, except the “ixpAdmin” user, select the user from the User Name pull-down list and click on the **Rename** icon.

After renaming a user, it is possible to edit the user definition as needed.

5.4.2.9 Undo Changes

 While working with a user information, you can undo any changes you have made prior to a Save or Delete by clicking on this icon.

5.4.2.10 User Message

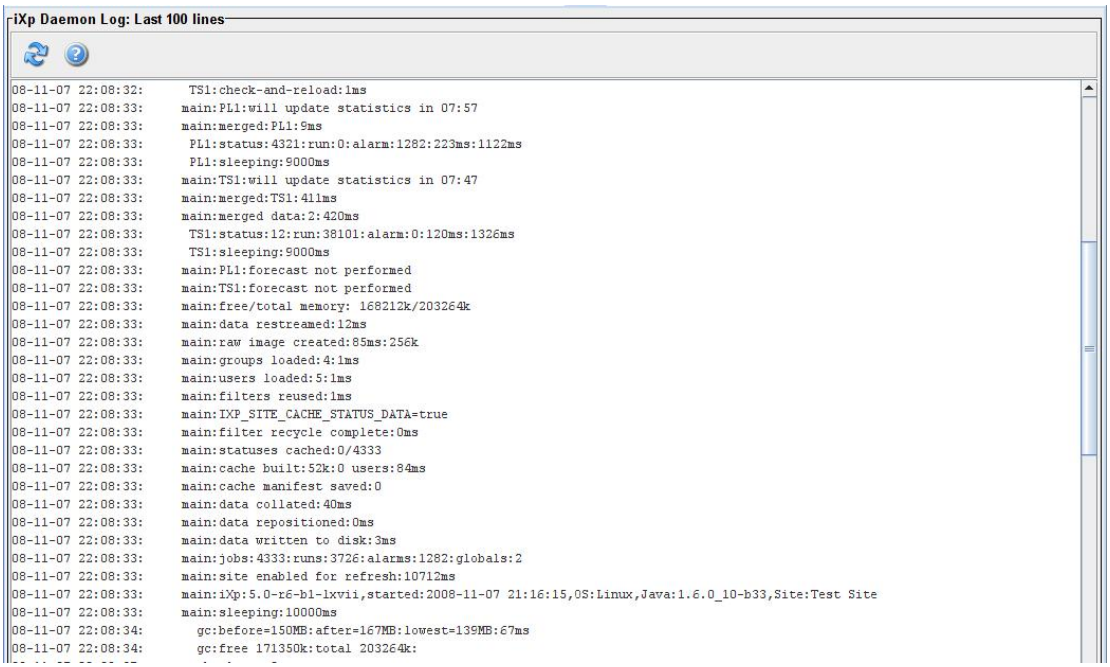
 This icon allows the administrator to send a message to the currently selected user. The message will display in a pop-up box and the user’s refresh will be paused until the message is acknowledged. A User Message expires after 20 minutes.

5.4.2.11 ixpAdmin User

The security settings for the `ixpAdmin` user cannot be changed. This user id represents the iXp “Administrator”. The password for this user can be changed from this screen.

5.5 Administration Log

Select the **Log** tab in the Admin Tool to view the iXp Daemon log.



 Click on Refresh icon to fetch the latest log, and select the number of lines, counted from the end of the log, that you want to view.

The iXp Daemon log stores all the messages, warnings and errors generated by the daemon. This log is very useful in tracking the status of the daemon and its threads, and also in troubleshooting the problems. The log is particularly useful for:

1) Changing or adding an Instance configuration. The effect of instance modifications on the daemon can be seen.

2) Checking for License errors.

3) Monitoring client access.

5.6 User Commands

User Commands are a way for iXp to pass job data to an existing UNIX, LINUX or Windows script, command, or program. If a user selects one job and executes a User Command, it is processed as follows:

1) The command is authorized for the user.
2) The proper environment variables are set.
3) The command is executed.
4) The results are returned.

User commands can be executed for a single or multiple jobs / alarms. If the command fails, or is unauthorized, the server captures the resulting text, and returns the results to the client.

There are three types of User Commands in iXp – Job, Alarm and Context-Free commands.

Job and Alarm commands are available only when the user has selected a job or an alarm in the iXp client. Context-free commands can be executed without selecting any job or alarm. A job command may be setup to return Operator Instructions for any job selected by the user on the client. An alarm command may be setup to page the on-call person with details about the alarm generated. A context-free command may be used to get system details of the iXp Server.

5.6.1 Create or Edit User Commands

 To create a new User Command:

1) Click on New User Command icon and assign a name.

2) Type or edit the command in the **Command** text box. Please enter the full path to any command or executable that is not included in the PATH set on the iXp Server for the user running the iXp Daemon.

3) Enter the **Input Prompt** text if applicable. (See the email address example in the *Interactive User Commands Section* of this document.)

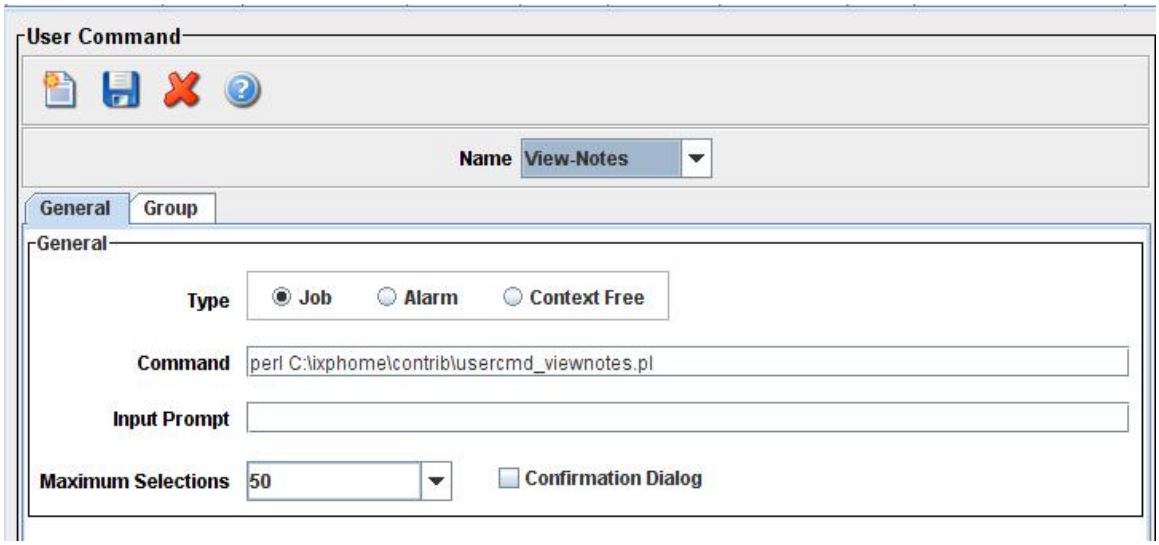
4) Select the **Type**: Job, Alarm, or Context Free.

5) Click on Save icon to save the new User Command to the iXp Daemon. This will make the User Command available to iXp users with the proper permissions. After defining a new User command, users with active sessions will have to exit and log back in to the GUI to access the new commands.

6) By default, when users execute any user command, they need to confirm the execution. If desired, the additional step of confirmation can be eliminated for certain user commands by un-checking the **Confirmation Dialog** checkbox.

7) If you are creating a Job or Alarm User Command, you can run the User Command on multiple jobs. In order to do so, you have to specify the maximum number of jobs/alarms that a user can select when executing this User Command. Use the **Maximum Selections** field to specify that limit. The default value for this setting is 50 (fifty). If the user command is returning a URL, then we strongly recommend setting the value for this field to 1 (one).

8) Click on Remove icon to remove the User Command from the iXp Daemon.



For User Commands of **Type Job**, the following environment variables are set on the iXp Server before the command is executed.

User Command Variable	Description
IXP_USERCOMMAND_JOB_NAME	The selected job's name.
IXP_USERCOMMAND_JOB_LAST_START	The selected job's last start time.

IXP_USERCOMMAND_JOB_LAST_END	The selected job's last end time.
IXP_USERCOMMAND_JOB_LAST_RUN_MACHINE	The selected job's last run machine.
IXP_USERCOMMAND_JOB_JOID	The selected job's joid value.
IXP_USERCOMMAND_JOB_STATUS	The selected job's current status, e.g. "SUCCESS".
IXP_USERCOMMAND_USER_NAME	The name of the iXp user executing the command.
IXP_USERCOMMAND_INPUT	The text input by a user during execution of an *Interactive User Command*.
IXP_USERCOMMAND_INPUT_PROMPT	The text prompted to a user during execution of an *Interactive User Command*.

For User Commands of **Type Alarm**, the following variables are set on the iXp Server before the command is executed.

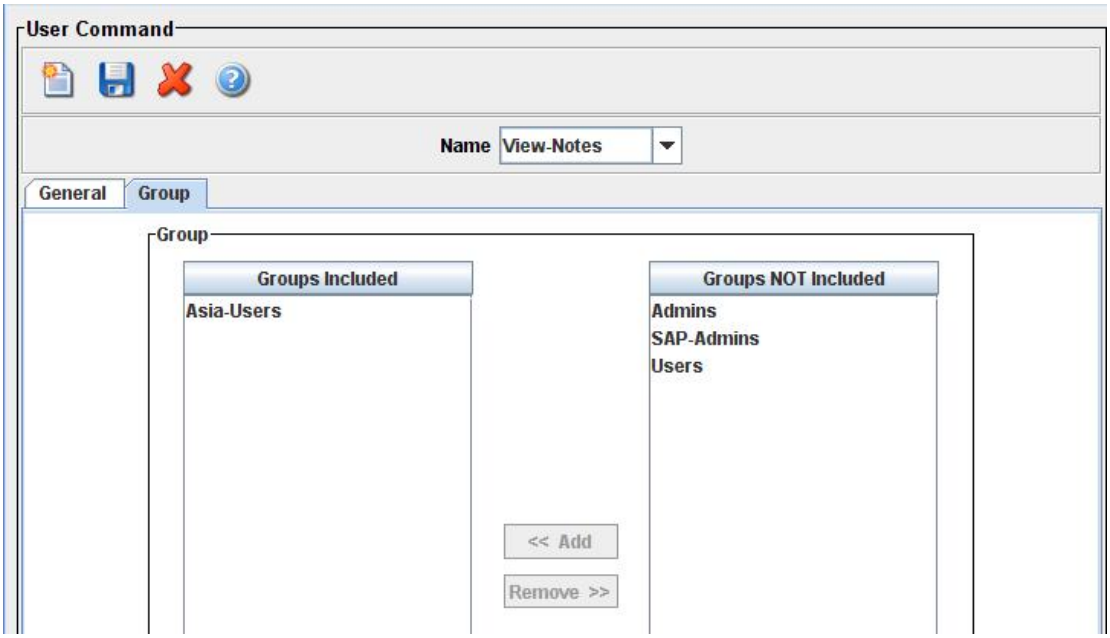
IXP_USERCOMMAND_ALARM_NAME, (e.g. `JOB_FAILURE`)
IXP_USERCOMMAND_ALARM_TIME
IXP_USERCOMMAND_ALARM_COMMENT
IXP_USERCOMMAND_USER_NAME
IXP_USERCOMMAND_ALARM_RESPONSE
IXP_USERCOMMAND_ALARM_JOB_NAME
IXP_USERCOMMAND_INPUT
IXP_USERCOMMAND_INPUT_PROMPT

The following CA Workload Automation AE environment variables are set for both Job and Alarm User Commands.

AUTOSERV, AUTOUSER, AUTOSYS, DSQUERY, IXP_SERVER_URL, PATH, prepended with the proper `$autosys\bin` directory.

For **Context Free** User Commands, no variables are set, except for IXP_USERCOMMAND_INPUT, IXP_SERVER_URL, and the PATH variable of the Web Server.

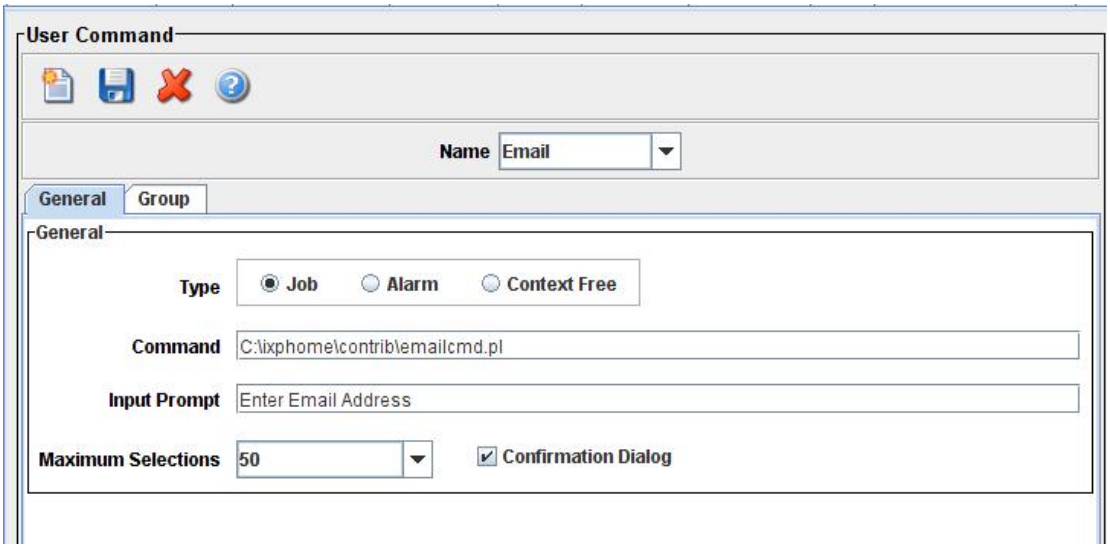
Just like filters, you can assign groups to User Commands. When you do that, only the members of the included groups, and iXp Administrators, will be able to execute the user commands.



5.6.2 Interactive User Commands

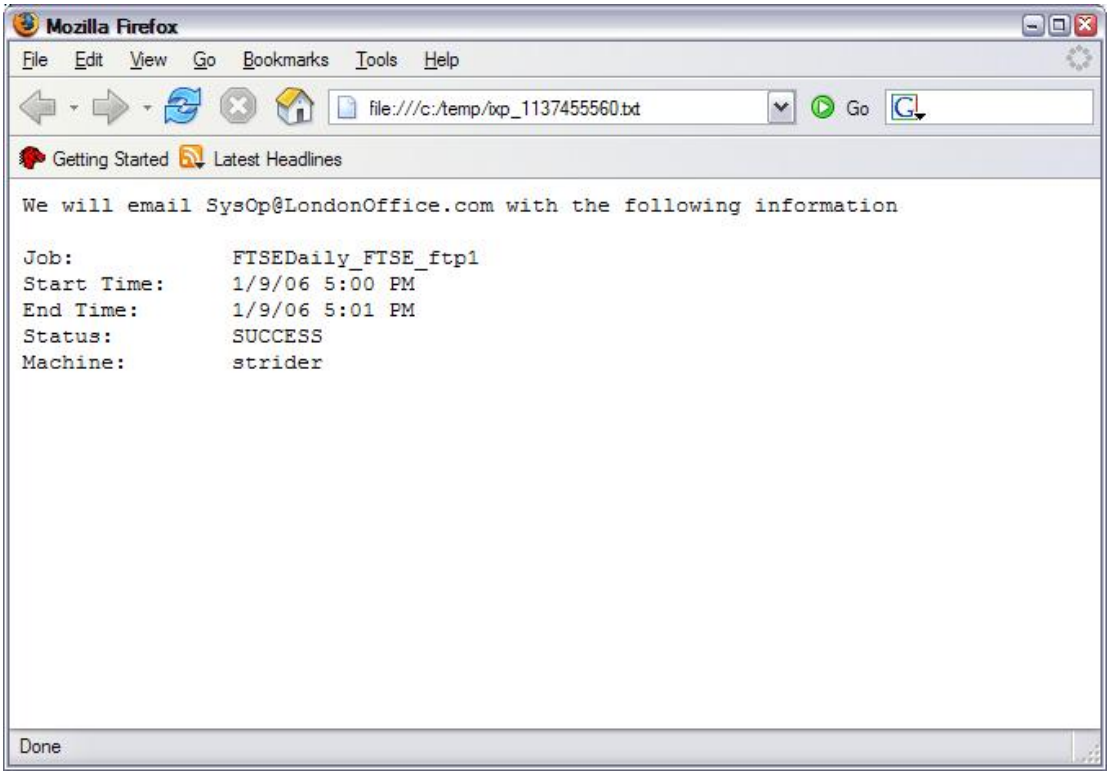
A User Command may prompt for input during user execution. The following example User Command prompts for an email address to send job run data.

The following defines the Email command with the prompt.



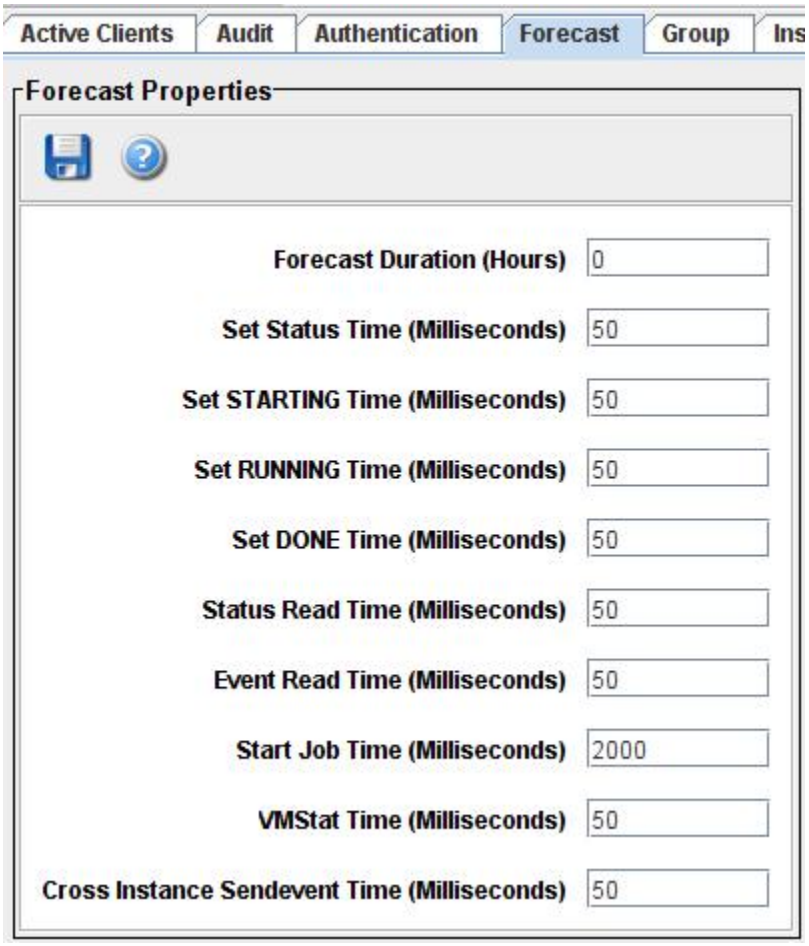
In the iXp GUI, select a job, and go to **File►User Command►Job►Email**. Since the User Command was setup with a non-empty Input Prompt, the iXp GUI will launch a dialog with an input prompt. Put in the desired information, and click on OK.

iXp shows any output generated by the User Command in a web browser window. Since iXp sets the above listed environment variables before executing the script associated with the User Command, the script can leverage the same variables and perform the required action.



5.7 Forecast

Select the **Forecast** tab to access iXp Forecast Properties.



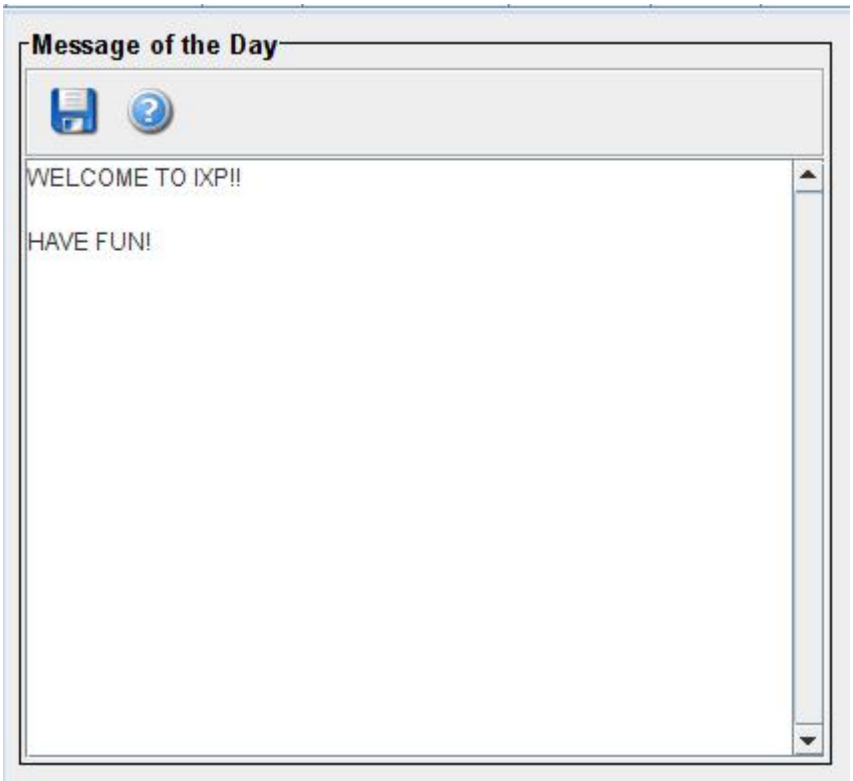
Edit and save these values to change the forecast properties, held in the `IXP_HOME/etc/forecast.props` file. These parameters delineate the output of the command line *forecast utility*, and the job run forecasts that will be viewable in the *Time View Chart*. These parameters also affect the iXp application *Interactive Forecasting* tool.

Forecast Property	**Description**
Forecast Duration	This setting is deprecated. Only the iXp forecasting command line interface uses it. This field specifies a time window for the forecasting engine, in hours. Example: setting this to "4" will generate a 4 hour forecast when processed via the command line.

Set Status Time	The estimated time in milliseconds that the Event Processor requires to update a status in the Event Server database.
Set STARTING Time	The time in milliseconds the Event Processor requires to update a status with the value of STARTING in the Event Server database. Its value may differ from the default (Set Status Time).
Set RUNNING Time	The time in milliseconds the Event Processor requires to update a status with the value of RUNNING in the Event Server database. Its value may differ from the default (Set Status Time) because the value should also capture the contention that may arise from a remote agent simultaneously performing a roughly identical task.
Set DONE Time	The time in milliseconds the Event Processor requires to update a status with the value of SUCCESS, TERMINATED, or FAILURE in the Event Server database. Its value may differ from the default (Set Status Time) because the value should also capture the contention that may arise from a remote agent simultaneously performing a roughly identical task.
Status Read Time	The time in milliseconds the Event Processor requires to read a job's status (such as when evaluating a job's "condition:" field) from the Event Server database.
Event Read Time	The time in milliseconds the Event Processor requires to read an event from the event server database. This value may reflect the presence of a backup server and the attendant event synchronization required.
Start Job Time	The time in milliseconds the Event Processor requires to perform the calculations to determine if a job can be started.
VMStat Time	The time in milliseconds the Event Processor requires to perform vmstat on each machine in the "machine:" field of a command job or file watcher.
Cross Instance Sendevent Time	The time in milliseconds the Event Processor requires performing a sendevent to an external instance when processing the job completion of a job that has an external successor.

5.8 Message of the Day

Select the Message of the Day tab to view or change the Message of the Day.

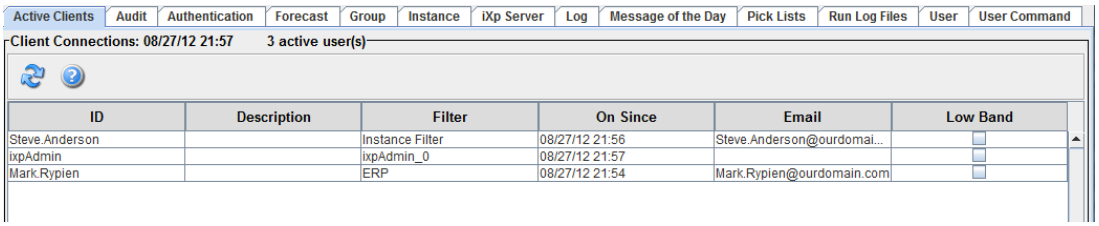


Click on the Save icon to save the message.

The Message of the Day appears in the iXp Logon screen. If the Message of the Day is changed, then all active iXp Clients will display the new message. Users have to acknowledge the message before continuing with the client.

5.9 Active Clients

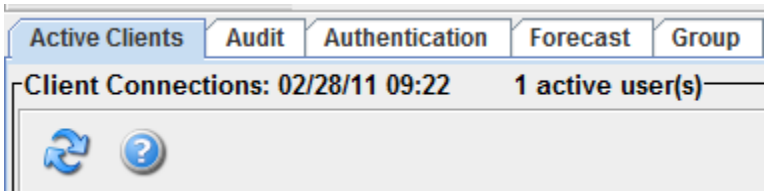
Select the Active Clients tab to access the active Client Connections list.



Client Attribute	Description
ID	User Name.
Description	Description as entered in the *User* tab.
Filter	Current active user filter.
On Since	Time elapsed since user logged on.
Email	Email address of user, as specified in the user definition.
Low Band	If a logged on user has been assigned the “Low Band” setting, then this will be checked.

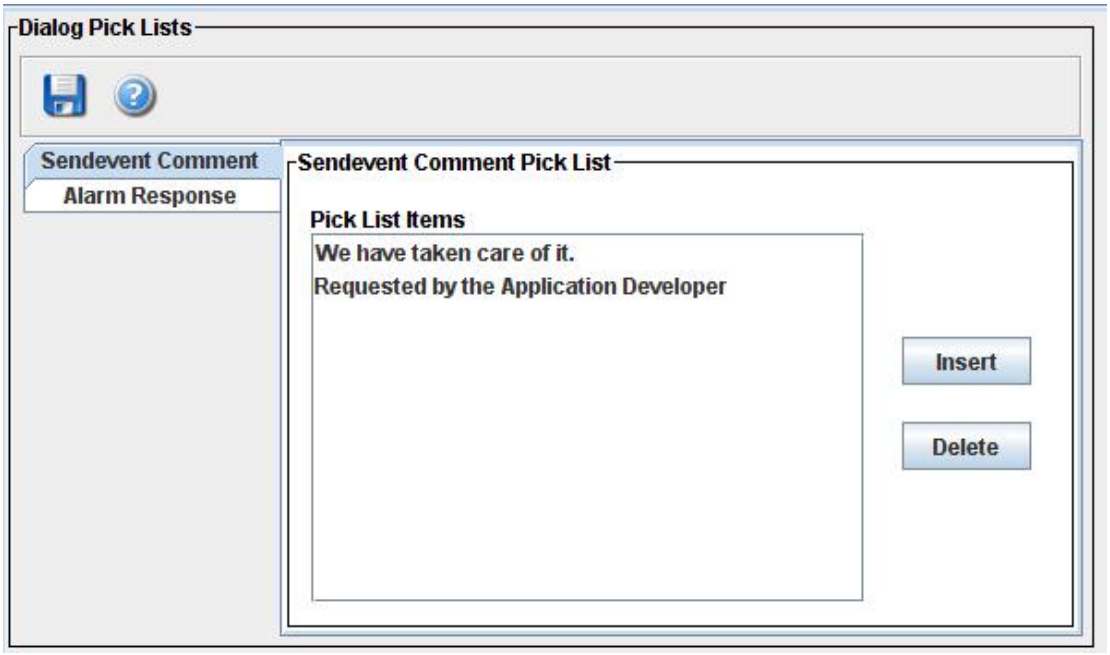
Click on the Refresh icon to update the list.

The number of active client sessions (users) is displayed on the top line:



5.10 Pick Lists

Select the **Pick Lists** tab to access the Dialog Pick Lists functions.



Sendevent Comment and **Alarm Response** pick lists may be populated to be available in the iXp application. The pick lists can be used to pre-define standard comments and responses that can be selected by users when executing sendevents or responding to alarms.

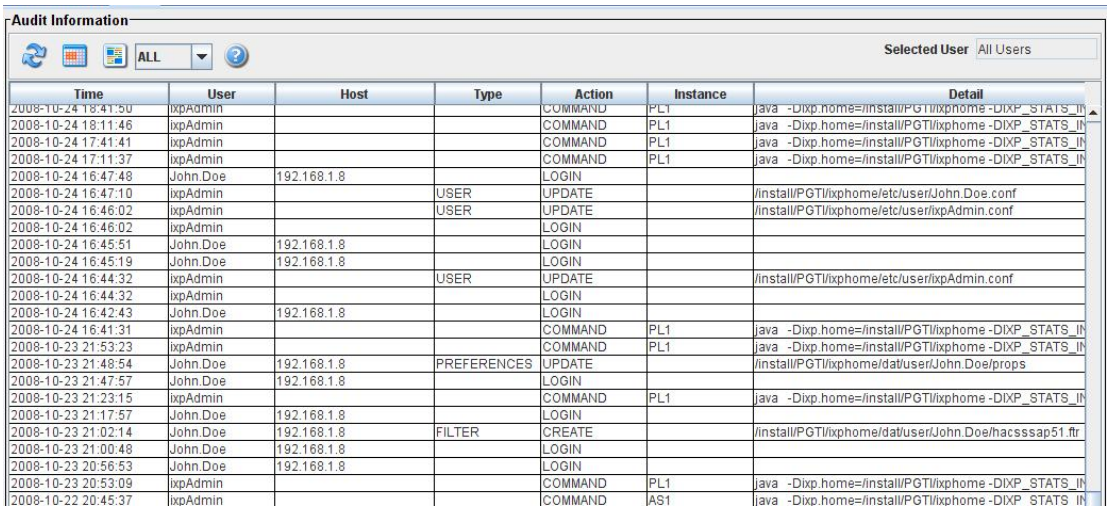
The Sendevent Comment pick lists appear in the *Job Control Popup* functionality sendevent comment input boxes. The Alarm Response pick lists appear in the *Alarm Manager* functions Acknowledge and Close alarm response input boxes.

To insert pick list items, click on **Insert**, enter the pick list text, click OK, then click on the **Save** icon.

To delete pick list items, highlight them, click on **Delete** to remove them from the list, then click on the **Save** icon.

5.11 Audit Trail

iXp keeps track of certain activities performed within the iXp GUI or the iXp Admin Tool. The audit trail of such activities can be seen in this tab of the Admin Tool.



Toolbar Field	Description
Duration	Choose to display all the audited activities for the last X days
	Refresh the displayed audit trail to show the latest information.
	Click on this button to create a CSV file of the displayed audit trail.
	Click on this button to create an HTML file of the displayed audit trail.

If a user performs any of the following activities in the **Admin Tool**, the audit trail will include them.

- Logs on to the Admin Tool
- Creates, Updates, or Deletes an iXp user definition.
- Updates the iXp Server information
- Creates, Updates, or Deletes an CA Workload Automation AE Instance information
- Creates, Updates, or Deletes an iXp User Command
- Creates, or Updates the Message of the Day.

- Creates, Updates, or Deletes any of the Pick list lines
- Creates, Updates, or Deletes a Filter definition.

If a user performs any of the following activities in the **iXp GUI**, the audit trail will include them.

- Logs on to the iXp GUI.
- Creates, Updates, or Deletes a Filter definition.
- Issues any sendevent successfully.
- Updates a job definition successfully.
- Creates, Updates, or Deletes a job override successfully.
- Updates the user preferences
- Saves the Session settings.
- Executes a User command successfully.
- Acknowledges or Closes an Alarm successfully.
- Changes the Password.
- Creates, Updates, or Deletes a Report definition.

If a user performs any of the following activities in the **iXp CLI**, the audit trail will include them.

- Views job summary, detail, or definition using *ixautorep*
- Creates, Updates Job definitions using *ixjil*
- Issues sendevents using *ixsendevent*
- Creates one-time job overrides using *ixjil*

Audit Attribute	Description
Time	The time of the activity.
User	The User that performed the activity.
Host	The name of the client machine from which the user performed the audited action. This field is used only in Single Sign-On mode.
Type	The Type of activity performed. This is an internal iXp classification mechanism.
Action	The performed activity.
Instance	The name of the CA Workload Automation AE Instance on which the action was performed.
Detail	The details of the actual activity performed.

You can sort by any of the above columns by clicking on the column header.

You can also view the Audit Trails for a single user by selecting the user in the menu provided in **View►View By User.**

5.12 Run Log Files

Select the **Run Log Files** tab to configure the Log File Retrieval from iXp Client.

Note: This tab is applicable for environments running CA Workload Automation AE r11 or older only.

5.12.1 Functional Overview

From the iXp Client, users can request access to the log files created by any job when it runs on its Remote Agent machine. These files include the STDOUT, STDERR, STDIN, Remote Agent log file, and Job Profile file. Since these files reside on the Remote Agent machine, the iXp Server communicates with the iXp Agent, which must be running on the Remote Agent machine, to retrieve the files.

For more information about the iXp Agent, refer to *Chapter 4 Architecture* and *Section 5.5.3 Installing iXp Agents*.

5.12.2 Configuration Settings

If “Run Log Retrieval Enabled” is checked, (and the iXp Agent is running) users will be able to view log files (std_in, std_out, std_err, profile, auto_remote log, script) depending on security permissions. If it is NOT checked, no access to logs is available through iXp.

5.12.3 Resolving File Names

The file name for each of the log files is resolved by the iXp Daemon and the iXp Agent by extracting the name of the file from the job definitions and then resolving Environment variables and Global Variables. If the file name includes standard CA Workload Automation AE variables such as `AUTO_JOB_NAME`, `AUTORUN`, `AUTOROOT`, `AUTOUSER`, `AUTOSERV`, `AUTOSYS`, then the applicable substitution is made depending upon the name of the selected job, and the values for the variables in either the iXp Agent's execution environment, or in the Instance's configuration file in the `$IXP_HOME/agent/conf` sub-directory on the Agent machine. If the file name includes any Global variables, then the appropriate value is substituted in the file name.

If the file name includes any non-standard variable or a variable whose value is dynamic, then you can specify the definition of such variables to iXp.

The iXp Daemon substitutes values for Global variables and substitution parameters defined to iXp. It then passes the resulting file name to the iXp Agent. The iXp Agent will substitute any remaining environment variables with the values found in the current environment. The file name that results from these substitutions will be then retrieved.

If the file name includes the following CA Workload Automation AE variables, then the substitution will happen as given below:

Variable Name	Substituted Value
AUTO_JOB_NAME	The name of the selected job.
AUTORUN	RUN_NUM-NTRY where RUN_NUM and NTRY are the highest run number and ntry values for the selected job.
AUTOSERV	The 3-letter AUTOSERV value of the selected job.
Base_Time	The UTC time of the last start time of the selected job.
__box_joid	The JOID of the box job that contains the selected job.
__box_name	The name of the box job that contains the selected job.
__job_name	The name of the selected job.
__joid	The JOID of the selected job.
__ntry	The highest NTRY for the latest run number of the selected job.
__run_num	The highest RUN_NUM of the selected job.
__run_machine	The name of the Remote Agent machine.

For example, if the selected job has the following JIL definition:

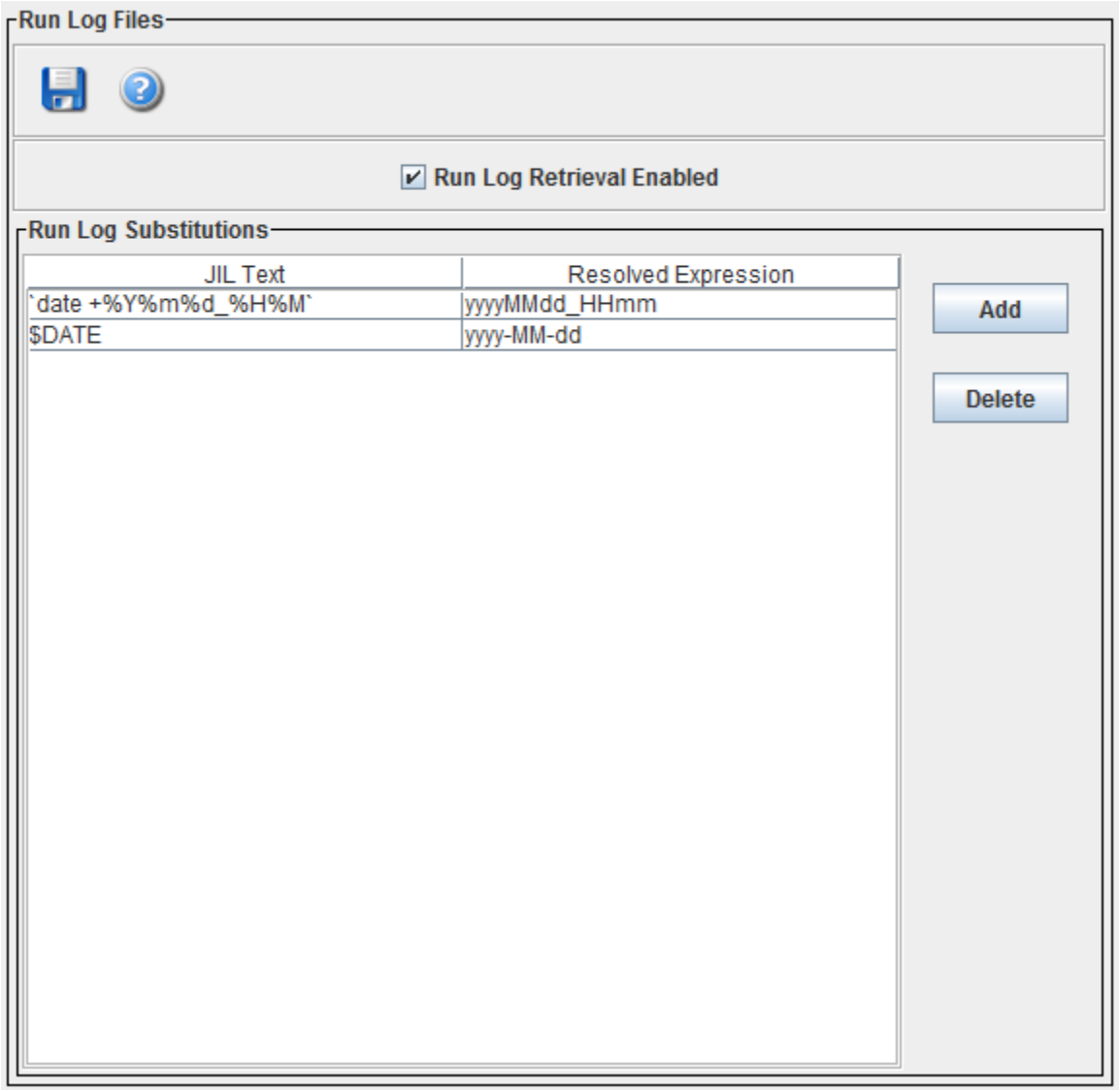
```
insert_job: FTSEDaily_FTSE_upld
job_type: c
box_name: FTSEDaily
machine: venus
command: sleep 5
std_out_file: /tmp/$AUTO_JOB_NAME.$AUTORUN.txt
std_err_file: /tmp/$AUTO_JOB_NAME.$AUTORUN.txt
```

And, if the job last ran with the run number `114` and ntry `1`, then the file name for STDOUT and STDERR will be `/tmp/FTSEDaily_FTSE_upld.114-1.txt`

If there is a global variable called LOGDIR defined to CA Workload Automation AE that has the value set to "*/opt/CA/logs*" and is used as follows in the JIL definition of a job:

```
std_out_file: $$LOGDIR/$AUTO_JOB_NAME.$AUTORUN.txt
std_err_file: $$LOGDIR/$AUTO_JOB_NAME.$AUTORUN.txt
```

Then, the file name for STDOUT and STDERR will be resolved to `/opt/CA/logs/FTSEDaily_FTSE_Daily_upld.114-1.txt`



If the file name uses variables that cannot be determined from the current environment on the Remote Agent, or if it uses variables that have their values set depending upon the date and time of the execution of the job, then you can specify the resolution format of those variables to iXp.

During substitution, the iXp Daemon will calculate the values of these variables as specified and then substitute in the file names. If the variable value includes date or time substitutions, then the iXp Daemon will calculate the value of those variables using the latest start time of the selected job.

As shown above, two (2) variables are defined to iXp. The “$DATE” variable needs date and time values to be substituted in a “yyyy-MM-dd” format. The iXp Daemon will use the latest start time of the selected job to determine the value of this variable.

For example, if the selected job last ran on 12th of January, 2006 and has the following JIL definition:

```
std_out_file: /tmp/$AUTO_JOB_NAME.$DATE.txt
std_err_file: /tmp/$AUTO_JOB_NAME.$DATE.txt
```

Then, the file name for STDOUT and STDERR will be resolved to `/tmp/FTSEDaily_FTSE_Daily_upld.2006-01-12.txt`

For more information on how to specify date and time format substitutions, please refer to the following table.

Letter	Description	Format	Example
yy/yyyy	Year	Numeric	05 / 2005
M/MM	Month	Numeric	2 / 02
MMM/MMMM	Month	Text	Jan / January
d / dd	Day	Numeric	8 / 08
EEE/EEEE	Day	Text	Tue / Tuesday
H/HH	Hour	Numeric	1 / 01
m/mm	Minute	Numeric	9 / 09
s/ss	Second	Numeric	8 / 08

Tip: For substitution variables, remember to put single-quotes around any letters/words in the values that need to be substituted “as-is”.

6 OTHER CLI UTILITIES

Command line utilities may be run at a Windows command line, or UNIX shell prompt, on the iXp Server machine.

6.1 Server statistics

The ixserverstats utility in the `IXP_HOME/contrib` directory allows users to get the iXp generated statistics for any CA Workload Automation AE instance. For information on how to run `ixserverstats.pl`, type:

```
%perl ixserverstats.pl
```

The output should read like this:

```
perl ixserverstats.pl -i instance -k [s|c|e|l] \
    -t [6|12|24|48|72|168] -f outputFile

-i (instance) should be the AUTOSERV variable (e.g. "ACE").
-k (kind) can be s (starts), c (completions), e (events), or l
(latency).
-t (time in hours) can be one of the following: 6, 12, 24, 48,
72, 168.
-f (file) to contain report output.
IXP_HOME must be set.
```

The serverstats script prints to the output file, in three columns. The first column contains sequential UNIX timestamps, the second column has the standard date/time stamp, and the third column has the number of events.

Example: `%perl ixserverstats.pl -i PA1 -ks -t6 -f outputfile` will create the output file with rows showing the number of events over the last 6 hours in the PA1 CA Workload Automation AE instance.

```
1207655707|4/8/08 5:55 AM|190
1207655857|4/8/08 5:57 AM|435
1207656007|4/8/08 6:00 AM|432
1207656157|4/8/08 6:02 AM|389
1207656307|4/8/08 6:05 AM|268
1207656457|4/8/08 6:07 AM|296
1207656607|4/8/08 6:10 AM|301
1207656757|4/8/08 6:12 AM|210
```

6.2 Forecast

For information on how to run **ixforecast.pl** in the IXP_HOME/contrib directory, type

%perl ixforecast.pl.

The output should read like this:

```
perl ixforecast.pl [-p propertyFile] [-r runReportFile] \
    [-e eventReportFile] [-f filter]

Must have either -r or -e (or both).
IXP_HOME must be set.
IXP_SERVER_URL must be set.
The default propertyFile is $IXP_HOME/etc/forecast.props
The default filter is "No Filter"
```

6.2.1 Forecasting Job Runs

For example, **%perl ixforecast.pl -r** prints ten columns to output file. Each row of output corresponds to one job run, similar to the Time View. The rows look something like this one:

```
JU2|job_0_0_0|EveryHour_15|963878909|7/17/00 6:08
PM|963879209000|7/17/00 6:13 PM|05m00s||SUCCESS|Run was
Scheduled based on Job Definition.
```

The columns are as follows:

Instance,
Jobname,
Box name of parent job,
UNIX time stamp in seconds of start of run,
Normal time stamp of start of run,
UNIX time stamp in seconds of end of run of job,
Normal time stamp of end time,
Duration,
Machine,
Forecasted status of run,
Reason for the job run.

If a column entry is not available, it is blank as in ||.

6.2.2 Forecasting Events

For example, **`%perl ixforecast.pl -e`** prints eight columns to the output file. Each row of output corresponds to one event in the CA Workload Automation AE database. The rows look something like these two:

```
964114081955|7/20/00 11:28
AM|JU2|oftenMins||STARTJOB|<eventQualifier>|Event was
Scheduled based on Job Definition.

964114084055|7/20/00 11:28
AM|JU2|oftenMins||CHANGE_STATUS|STARTING|<reason>
```

The columns are as follows:

- UNIX time stamp in milliseconds,
- human readable time,
- instance of the job,
- jobname,
- box name,
- type of event,
- event qualifier,
- reason for the event.

6.3 Currently Active Users

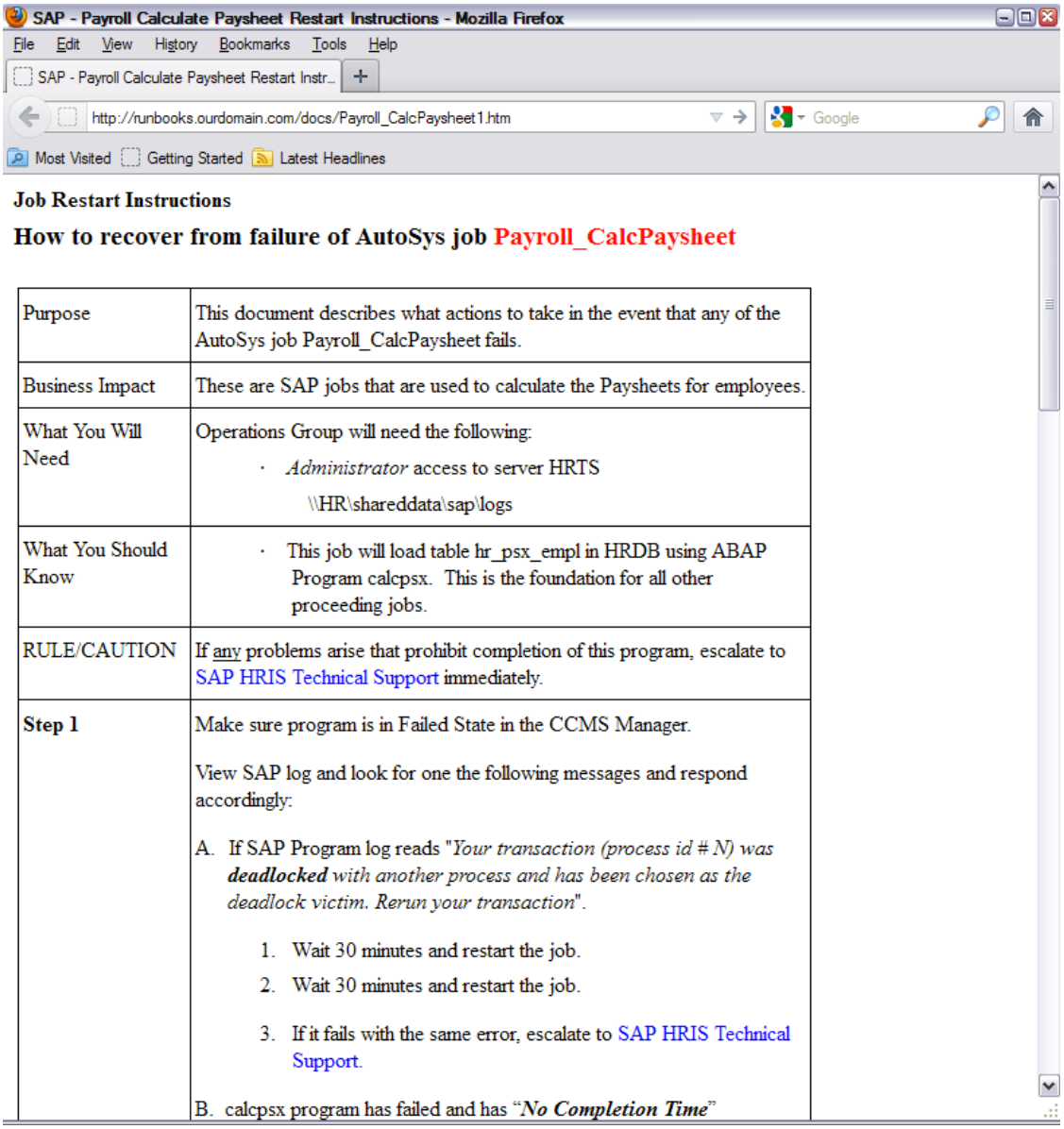
ixusers.pl is an iXp command, located in the `IXP_HOME/contrib` directory, which provides information about currently active users. It returns the user ids, their current filters, the size of the users' caches, and the time each user logged onto iXp.

For example,

```
total users=4
John.Doe, Level Zero Filter,6k,04/01/08 22:49
Jane.Doe,AsiaJobs,3k,03/25/08 19:18
Rob.Smith,EuropeJobs,3k,03/28/08 15:34
Mary.Jackson, Exception Filter,1k,04/03/08 22:01
```

6.4 Job Information and Restart Instructions

A **Job Information and Restart Instructions** browser popup function is available. Select on a job name in iXp (the following example uses “EuropeHourlyBox”) then launch a new browser to view the Job Information and Restart Instructions popup, like this.



6.5 Audit Report

ixauditreport.pl is an iXp command, located in the `IXP_HOME/contrib` directory, which generates an Audit Report in HTML or CSV format. You can generate a report for either a specific user or all the “Admin” users.

For information on how to generate the Audit Report, run the command “**perl ixauditreport.pl**”

```
perl ixauditreport.pl -d auditDays -f outputFileName [-h || -c] [-u
user || -i]
-d is required and should contain the number of days
-f is required and should contain the full Path for the output
fileName.
-h OR -c is required and indicates file extension html OR csv
-u OR -i is required and -u should contain the valid user name. -i
indicates all the users with ixpAdmin Privilege
IXP_HOME and CATALINA_HOME must be set
```

7 GLOSSARY

7.1 iXp Related Terms

Console View: Tabular listing of jobs and their vital data in the upper right portion of the iXp window.

Critical Path Filter: Focuses on a job and then follows predecessor and successor relationships (dependency relationships) from that job.

Exception Filter: A Job Attribute Filter that Displays only jobs that have failed or been terminated.

HTTPD: HyperText Transfer Protocol Daemon.

Instance Filter: A Job Attribute Filter that displays only the Site and its Instances.

Job Attribute Filters: Filters that use each CA Workload Automation AE job attribute to further define or limit the views in iXp.

Job Dependency Filters: Filters that limit the iXp views according to specified job dependencies.

Job Detail View: View in the upper center of the iXp window that shows details of the selected job.

Job Flow View: View in the lower right portion of the iXp window that shows the predecessor and successor job relationships for each job.

Level Zero Filter: A Job Attribute Filter that displays the Site, the Instances, and Level 0 boxes.

Recursive Critical Path Filter: Applies an ancestor-progeny filter that passes all the parents and children of the selected job, all their parents and children, and so forth through the job stream. Then a Critical Path filter is applied to each of those jobs.

Running Filter: A Job Attribute Filter that displays only running jobs.

Server Invocation: A call from a client (web browser) to the server, which in turn makes a call to refresh or request data from a CA Workload Automation AE Instance.

SHTTP: Secure HyperText Transfer Protocol

Site: A group of one or more CA Workload Automation AE instances.

Site Filter: A Job Attribute Filter that displays only the Site icon.

Transitive Closure Filter: Follows all Box and dependency relationships in both directions, throughout the CA Workload Automation AE Job structure. First a Recursive Critical Path filter is applied to the selected job, and then a Recursive Critical Path filter is applied to the resulting jobs, and so forth through the job stream.

Tree View: View on the left of the iXp window that depicts the hierarchical box structure of the jobs.

7.2 CA Workload Automation AE Terms

See also *CA Workload Automation AE User Guide*

Alarm: Alarms are special events that notify operations personnel of situations requiring attention.

Box Job: A CA Workload Automation AE job that spawns other jobs that are "contained" in the Box Job.

Global Variable: Set by the `sendevent SET_GLOBAL` commands, Global Variables are used to create dependency relationships.

JIL: Job Information Language. Refer to the *CA Workload Automation AE User Guide*.

Job: A job is the basic building block upon which an operations cycle is built. A CA Workload Automation AE job is any single command or executable, UNIX shell script, or NT batch file. Each CA Workload Automation AE job definition contains qualifying attributes, including conditions for when and where a job should be run. Command Jobs execute commands, Box Jobs are containers, which hold other jobs, and File Watcher Jobs watch for the arrival of a specified file.

Job Name: The job name is used to identify the job to CA Workload Automation AE, and must be unique within CA Workload Automation AE. It can be from 1 to 30 alphanumeric characters, and is terminated with white space. Embedded blanks and tabs are illegal.

JOID: Job ID (see *CA Workload Automation AE Reference Guide*).

`sendevent:` A CA Workload Automation AE command to activate the Event Processor.

7.3 Symbols in this Guide

Symbol or type style	Represents	Example
Bold	A new term.	The **console** is the upper right window in iXp.
7.3.1.1.1 Alternate color	Hyperlinks to other sections, or to the internet.	See *http://java.sun.com* for further information.
7.3.1.1.1.1 Italic	Words that are emphasized and *blue italic* for hyperlinks.	Dismiss the window *after* finalizing your changes.
	Underlined Italic for titles of other documents.	*CA Workload Automation AE User Guide*
	Syntax variables.	COPY *filename*
`Monospace`	Directories, file names, command names, computer code.	`&HIGHLVL.SRCLIB`
	Computer screen text, system responses, command line commands.	`Copy file? Y/N`
`Monospace bold`	What a user types.	...enter **`RUN APP.EXE`** in the Application field.
`<>`	The name of a key on the Keyboard.	Press <Enter>.
►	Choosing a command from a menu.	Edit ► Preferences.

7.4 Related Documents

CA Workload Automation AE User Guide

CA Workload Automation AE Windows Implementation Guide

CA Workload Automation AE UNIX Implementation Guide

CA Workload Automation AE Reference Guide

iXp Release Notes

iXp User Guide