Novell
ICEcore

INSTALLATION GUIDE

Novell.

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

ICEcore*

Copyright © 1998 – 2008 SiteScape, Inc., and its licensors. All rights reserved. ICEcore software is governed by the Common Public Attribution License Version 1.0 (the "CPAL"); you may not use ICEcore software except in compliance with the CPAL. You may obtain a copy of the CPAL at www.opensource.org (http://www.opensource.org/licenses/cpal_1.0).

Contents

About This Guide

This guide covers the installation of ICEcore. In this guide, the term "ICEcore" applies to all versions of the ICEcore software unless otherwise noted.

This guide provides information about Installing ICEcore.

Audience

This guide is intended for ICEcore administrators.

Software and Documentation Version

This guide describes features in ICEcore Version 1.0.3. This is the first version of this guide.

Feedback

We want to hear your comments and suggestions about this guide and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com (http://www.novell.com/documentation/feedback.html) and enter your comments there.

Additional Documentation

You can find more information in the ICEcore documentation, which is accessible from links within ICEcore:

- ICEcore Help system
- *ICEcore Quick Start Guide*
- *ICEcore User Guide*
- *ICEcore Administration Guide*

The ICEcore online documents can be found within the ICEcore Help system. To access the ICEcore Help system, log in, then click the *Help* link.

In the ICEcore Help system, click the *Getting Started Manuals* link to access copies of the online documents listed above.

Change History

This table documents changes for recent revisions to this guide:

Guide Revision	Changes
0 (First Version)	

Documentation Updates

For the most recent version of the *ICEcore Installation Guide* and other documentation, visit the Novell Teaming Web site (http://www.novell.com/documentation/team_plus_conf).

Conventions

This guide uses the following conventions:

A greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux* or UNIX*, should use forward slashes as required by your software.

Installing ICEcore

<div align="right">1</div>

This section describes how to initially install and configure ICEcore:

1.1 Prerequisites

You need a few things before you install ICEcore:

1. Computer:

   - Minimum 2 GHz processor
   - Multi-CPU systems preferred
   - Minimum 2 GB RAM

   > **NOTE:** You can potentially run with less RAM for specific development and testing configurations without simultaneous users, lots of database traffic, etc.

   - Linux systems need to have a minimum open file limit of 4096.

     For SUSE® Linux Enterprise Sever (SLES), check `/etc/security/limits.conf`:
     ```
     * hard nofile 65535
     * soft nofile 4096
     ```

   See "Memory Guidelines" on page 17 for details.

2. Sun* JDK 1.5.0_011 or higher or the IBM JDK 1.5 installed (NOTE: Java* 1.6 is not yet supported).

   > **NOTE:** Sun* has changed the name of JDK 1.5 to 5.0. In this manual, any reference to the JDK 1.5 refers to JDK 5.0, see http://java.sun.com (http://java.sun.com/j2se/1.5.0/docs/relnotes/version-5.0.html) for more information.

3. A database server:

- MySQL* 5.0.26 (or higher) Server and Client for Linux or MySQL 5.0.26 (or higher) Server and Client for Windows. MySql 5.0.26 ships with SUSE® Linux Enterprise Sever (SLES) 10 SP1.

  NOTE: MySQL 5.1 is not yet supported.

- SQL Server* for Windows* (2000 or 2005)

- Oracle* 9, 10

- Enough disk space for the software and for the data files:

  This depends on how much data you plan to put into the system. See sections "Database Planning" on page 11 and "File System Planning" on page 14.

  The software takes about 250 MB.

See "Database Planning" on page 11 for details.

1.2 Steps for Installing ICEcore

1 Install the JDK.

- The Sun JDK 1.5.0_011 (or higher) or the IBM JDK version 1.5 is supported.

- Java 1.6 is currently not supported.

You can download the Sun JDK from java.sun.com (http://java.sun.com/products/archive). Select the "JDK/JRE - 5.0" product line and download the JDK.

The IBM JDK 1.5 is available on the SLES 10 distribution. Use YAST to select and install the IBM JDK. Larger deployments with 64-bit hardware should seriously consider using the 64-bit JDK, because it allows increased memory allocation and better performance.

On the Sun site, you can ignore the notes in the 64-bit JDK about "Java Web Start" and "applets," because ICEcore is neither of those.)

2 Set the `JAVA_HOME` environment variable on your computer to the correct JDK path and name on your system.

For example:

- On Linux: `export JAVA_HOME=/usr/java/jdk_1.5.0_11`

- On Windows: `JAVA_HOME=C:\Program Files\Java\jdk1.5.0_11`

WARNING: In Windows, you must set the JAVA_HOME environment variable. Also, close and re-open any command prompt windows after setting this variable.

---

3 Install and Configure the Database Server. See "Database Planning" on page 11.

4 Download the appropriate ICEcore kit and unzip it.

5 If you want to run the indexing service in ICEcore on a different machine from ICEcore, install the Lucene Index Server on the other machine. See "Installing a Standalone Lucene Index Server" on page 29.

6 Run the installer from where you unzipped the ICEcore kit. See "Running the Installer" on page 23.

The installer is named one of the following, according to your operating system:

- `installer-liferay.exe`
- `installer-liferay.linux`

If you are doing an upgrade, use your existing `license-key.xml` and `installer.xml` files by placing them in the same directory as the installer program. If you are doing a new installation, use your `license-key.xml` by placing it in the same directory as the installer program. The `license-key.xml` file required to install the product is provided with the software kit (but is not included in the kit).

7 Start and stop ICEcore. See .
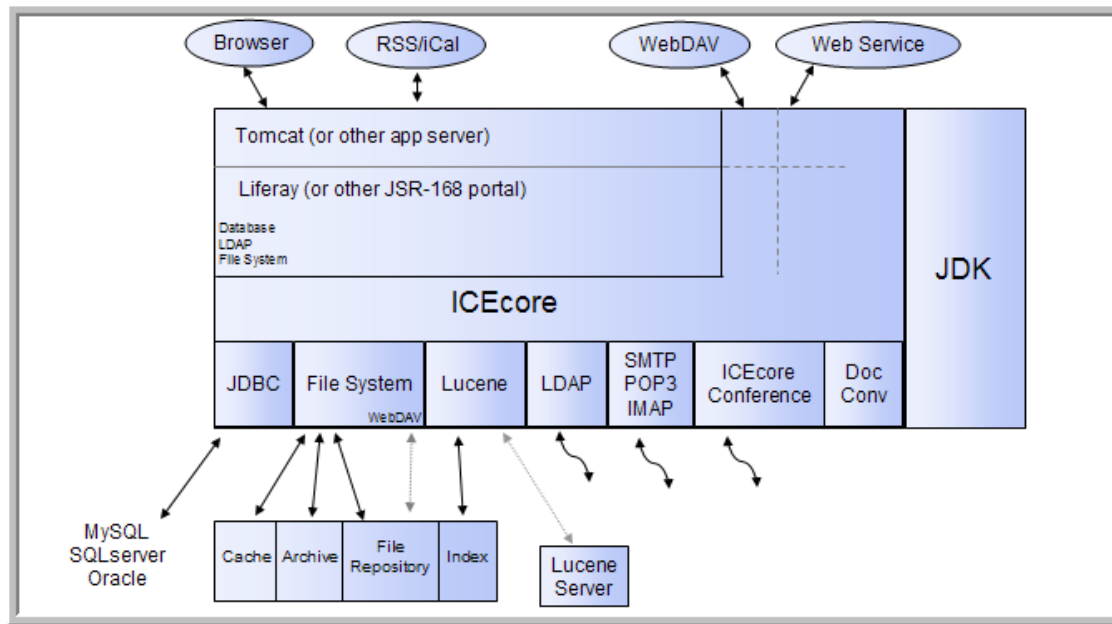
1.3 Database Planning

ICEcore and Liferay* use separate dedicated databases within your database server. A set of SQL configuration scripts are used to initialize the databases (creating the necessary tables, etc.).

ICEcore's default database for Linux and Windows is MySQL. ICEcore also supports SQL Server on Windows, and Oracle on Linux and Windows.

ICEcore's database requirements are relatively modest. The bulk of the data uploaded to ICEcore is stored in a file repository (see). The database is primarily used for storing metadata and descriptive text.

Because the amount of data stored in the database is highly sensitive to the usage patterns of ICEcore (which are highly variable) there is no reliable formula for determining disk space usage, but the following can be used as a guideline:

- numberAttachments x averageAttachmentSize = totalAttachmentSpace
- totalAttachmentSpace * .04 = sqlDataSpace
- sqlDataSpace x 5 = sqlStorageSpace

NOTE: The installer generates your database scripts for you, but if your DBA wants to create the databases by hand, the necessary scripts are provided in the kit. See "Non-Local Database Setup" on page 13.

See the following subsections:

1.3.1 MySQL

- MySQL 5.0.37 (or higher) Server and Client for Linux or MySQL 5.0.26 (or higher) Server and Client for Windows are required with `innodb support` enabled.

  NOTE: MySQL 5.1 is not yet supported. You can get the MySQL Community Server at: http://www.mysql.com/ (http://www.mysql.com/).

- Specify `root` for the administrator password when you run the installer.

  NOTE: The installer generates your database scripts for you, but if your DBA wants to create the databases by hand, the necessary scripts are provided in the kit. See "Non-Local Database Setup" on page 13.

- Set the default character set to UTF-8 by selecting *Best support for Multilingualism* in the Windows Configuration window, or edit the `my.cnf` configuration file to the following:
  ```
  [mysqld]
  character_set_server = utf8
  [client]
  default_character_set = utf8
  ```

  NOTE: This file is usually located in `/etc/my.cnf` for Linux and in `c:\my.cnf` for Windows. See the MySQL documentation for details.

1.3.2 Microsoft SQL Server

- You can use SQL Server 2000 or SQL Server 2005
- Make sure to select SQL Server and Windows for authentication (the default is Windows only)
- Set the administrator password to `sa` when you run the installer.

  NOTE: The installer generates your database scripts for you, but if your DBA wants to create the databases by hand, the necessary scripts are provided in the kit. See "Non-Local Database Setup" on page 13.

1.3.3 Oracle Database

- Supported versions: Oracle 9, Oracle 10.

Oracle's default database name is `orcl`. We use this name in our examples. If you have chosen a different name, replace `orcl` with your database name.

- ICEcore is enabled for Unicode and requires you to set up the Oracle Database Character Set to support Unicode character encodings. ICEcore requires either the UTF8 or AL32UTF8 character set for proper operation. Oracle recommends the use of AL32UTF8, because it has increased support for certain Asian languages. Consult your Oracle documentation for the choice that makes sense for your database.

- To check your database character set, execute the following command with SQL*Plus:
  ```
  select value from SYS.NLS_DATABASE_PARAMETERS where PARAMETER =
  'NLS_CHARACTERSET';
  ```

- Before you install the ICEcore application software, run the two Oracle database scripts. These are provided in the kit:
  ```
  \database\oracle\create-database-oracle.sql
  \database\oracle\create-oracle.sql
  ```

  1. These two scripts need to be accessible to the SQL*Plus utility and your database, so copy them to a location on your system path.

  2. Edit the supplied database scripts to set the passwords for the users (these are defined in the top few lines of each script).

     For example, in `create-database-oracle.sql`, you see:
     ```
     drop user sitescape cascade;
     create user sitescape identified by sitescape;
     grant connect, resource to sitescape;
     connect sitescape/sitescape;
     ```
     - `user sitescape` indicates that `sitescape` is the name of the database created. Do not change this.

     - `identified by sitescape` declares that `sitescape` is the password for this database. Change this to the password you intend to use, such as `identified by yourpassword` and change the `connect sitescape/sitescape` to `connect sitescape/yourpassword`.

  3. Use the SQL*Plus utility with the SYSTEM account to run the scripts:
     ```
     sqlplus SYSTEM/systemPassword @create-oracle.sql
     sqlplus SYSTEM/systemPassword @create-database-oracle.sql
     sqlplus> quit
     ```

1.3.4 Non-Local Database Setup

When installing the ICEcore software for the first time, one of the steps involves creating two databases: one for the portal and one for the ICEcore application. If your DBA wants to create the databases by hand, the necessary scripts are provided in the kit. This is required for Oracle. For more information, see .

The database scripts are in a `database` directory next to the installer:

```
\database\mysql\create-database-mysql.sql
\database\mysql\create-mysql.sql

\database\sqlserver\create-database-sqlserver.sql
```

```
\database\sqlserver\create-sql-server.sql

\database\oracle\create-database-oracle.sql
\database\oracle\create-oracle.sql
```

When you run the installer, select the *My DBA has already created the databases* option in the *Database selection* window.

1.4 File System Planning

ICEcore software and configuration files are stored in a tree shared with Liferay, Tomcat, etc. There are some temporary files also located here, but it is used mainly for locks and similar files.

ICEcore data is stored in the database (see) and on the file system. The file system usage is divided up into several functional areas:

- `filerepository`: This is where all attachment files are located, so it is a large consumer of disk space. The tree is roughly organized by site, binder (folder/workspace), and entry.
- `archiveStore`: Only activated in the Enterprise version of ICEcore. This is where previous file versions are stored to meet compliance and archival goals.
- `cachefilestore`: This tree holds information derived from the attachments, such as thumbnails, scaled images, text, and HTML renderings. Depending on the nature of the attachments this tree consumes somewhat less space than the file repository (but it can, conceivably, store more).
- `lucene`: This tree holds the search index for the data. It tends to be a fraction of the space consumed by the file repository, but it is also sensitive to the type of information stored.
- Other trees: These are other trees that you cannot configure, which typically only consume a small amount of space.
  - `rss`: Caches of RSS feeds for folders
  - `temp`: Temporary files

1.5 Network Planning

1.5.1 Planning Firewalls, Ports, and Session Timeouts

ICEcore is a complete Java/J2EE Web application and includes a Web server (Tomcat). Because the Web server listens on ports and processes network connections, you should consider the following as part of the installation process:

Firewall Setup: The default installation uses port 8080/8443 for the Web server. Some local firewalls (such as the Windows Security Center's firewall) might restrict traffic to port 8080/8443 and thus require proper configuration for access by other systems.

If you can access ICEcore on the local server, but cannot access it from another system, examine your firewall settings.

NOTE: When utilizing an Oracle database, you may need to reconfigure ICEcore to run on port 8081.

Port Conflicts: The Tomcat software uses a number of TCP/IP ports for both Web traffic and management messages. If you have other Web servers (Apache, IIS, or Tomcat) or other software installed that uses the same port numbers, your ICEcore software does not function properly.

NOTE: Novell Open Enterprise Server 2 (OES 2) systems have another Tomcat application running by default.

The `netstat -anp tcp` command on Windows and the `netstat -tan` command on Linux can help find which ports are currently being used. For more information on network ports, consult your operating system networking guide.

When you are in this situation, make sure that in addition to the http/https ports (8080/8443) you also assign free port numbers to the *AJP Port* and *Shutdown Port* settings in the *Network information* window when you run the installer.

Port Mapping: For instances where the port number on which Tomcat listens for connections is different than what the end user sees in his or her browser, change the *Listen Port* and *Secure listen Port* settings to the local server ports and set the *Port* and *Secure Port* settings to the port numbers that the browsers are using in the *Network information* window when you run the installer. See "Port 80/443 configuration on Linux" on page 15 and "Integrating with iChain" on page 16 for details.

The host, port, and securePort settings are used to generate URLs within the product and so must match what the user's browser sees. Unless they are overridden by the *Listen Port* and *Secure listen Port* settings, the *Port* and *Secure Port* settings are used as the server's listen ports.

Liferay Session Timeout: You can configure ICEcore to log out of a session after a certain number of minutes. The default is 2 hours (240 minutes). You can change this by altering the *Liferay session timeout* setting in the *Network information* window when you run the installer.

1.5.2 Port 80/443 configuration on Linux

The default installation of ICEcore uses the Tomcat Web server. Although the Tomcat server is part of the Apache project, it is not the same thing as the Apache Web server. Tomcat is written in Java and can not perform some of the special privilege behaviors that Apache can.

In particular, Linux does not allow non-root processes from allocating TCP/IP ports less than 1024. For this reason, the default configuration for Tomcat (and ICEcore) uses ports 8080 for http and 8443 for https (SSL). Unfortunately, this requires specifying the port number in the browser's URL for ICEcore (such as http://icecore.mycompany.com:8080).

WARNING: Although running Tomcat as `root` solves this problem, it creates many (far worse) problems. Do not run Tomcat as `root`.

If you want ICEcore to be available on the default http/https ports, use an operating system feature called "kernel space port forwarding." With this, the `iptables` command is used to map requests from port 80 to port 8080 (or whatever port you specify). For example:

```
iptables -t nat -A OUTPUT    -d localhost    -p tcp --dport 80 -j
REDIRECT --to-ports 8080
```

```
iptables -t nat -A OUTPUT     -d yourHostname -p tcp --dport 80 -j
REDIRECT --to-ports 8080
iptables -t nat -A PREROUTING -d yourHostname -p tcp --dport 80 -j
REDIRECT --to-ports 8080
iptables -t nat -A OUTPUT     -d localhost    -p tcp --dport 443 -j
REDIRECT --to-ports 8443
iptables -t nat -A OUTPUT     -d yourHostname -p tcp --dport 443 -j
REDIRECT --to-ports 8443
iptables -t nat -A PREROUTING -d yourHostname -p tcp --dport 443 -j
REDIRECT --to-ports 8443
```

These changes only affect the running system. You need to add them to your startup sequence. See the `*iptables*` and `*iptables_save*` man pages for more information about setting up port forwarding.

When you use this type of configuration, you must specify the ports the Tomcat server listens to and the ports that the browsers see in the *Network information* window when you run the installer.

For example:

- ◆ *Port* = 80
- ◆ *Listen Port* = 8080
- ◆ *Secure Port* = 443
- ◆ *Secure listen Port* = 8443

1.5.3 Integrating with iChain

NOTE: This feature is only available in the Enterprise version.

Liferay/ICEcore contains support for single sign-on (SSO) based on iChain®. When ICEcore is running in this mode, the iChain server performs user authentication tasks and passes a token representing the authenticated user's login name to Liferay/ICEcore in each request to indicate who made the request.

You must configure Liferay/ICEcore with the same LDAP that iChain is using so that it can obtain and copy the user information into the portal. However, this configuration differs from the regular portal/LDAP integration in that no password checking is performed against LDAP when logging into the portal (because credential checking is only performed by the iChain server). This technique is safe only if all access to the portal is routed through the iChain proxy.

The SSO configuration sets up a valve that only permits access from the local server and the iChain proxy server. For more information about Tomcat's valve configuration, see tomcat.apache.org (http://tomcat.apache.org/tomcat-5.5-doc/config/valve.html).

To set up an iChain proxy:

1 Assure that the iChain proxy's host name and port number match the ICEcore/Liferay port number. If you have configured ICEcore to run on port 8080, the iChain proxy should also use port 8080. If the port numbers differ, you must set the *Listen Port* setting in the *Network information* window when you run the installer.

For example, if iChain is set to provide access on the default ports (80/443) while Tomcat is listening on ports 8080/8443, use the following setting values in the *Network information* window when you run the installer:

- ◆ *Host* = your.server.host.name
- ◆ *Port* = 80
- ◆ *Listen Port* = 8080
- ◆ *Secure Port* = 443
- ◆ *Secure listen Port* = 8443

IMPORTANT: If the iChain proxy and ICEcore host names/ports are not set up identically, you might experience page layout issues and other ICEcore features might not function properly.

2 In *iChain configuration* window of the installer, select the *Enable iChain* option, set the *Logoff URL* setting to the log-off URL provided by the iChain proxy server, and set the *iChain Proxy* setting to the IP address of the iChain proxy server.

3 Finish running the installer's New Installation or Reconfiguration. See .

4 After enabling the iChain SSO, log into Liferay by using a direct login URL: `http://localhost:8080/c/portal/login` and log in as `admin`.

Because the SSO configuration removes the *Sign In* link, you must type the URL manually.

5 If you have not already done so, use Liferay's *Admin* portlet to enable LDAP-based authentication.

This setup is necessary because Liferay has to copy user data from the LDAP into the portal database upon user login.

6 To prevent users from logging into the portal directly through the portal's login form, remove or rename the following file from ICEcore's `liferay-portal-tomcat` directory:

`webapps/ROOT/html/portal/login.jsp`

You might also need to remove:

`work/Catalina/localhost/_/org/apache/jsp/html/portal/login_jsp.*`

7 Liferay/ICEcore is now ready to be put behind the proxy. Consult your iChain documentation on how to do this.

8 To enable iChain integration-related debug log messages, modify Liferay's `portal-log4j-ext.xml` to add the following category:

```
<category name="com.sitescape.team.liferay.security.auth">
 <priority value="DEBUG" />
</category>
```

Use this only for testing or troubleshooting purposes. It is not recommended to have debug logging enabled on a production system.

1.6 Memory Guidelines

ICEcore is a Java/J2EE application and therefore executes within a Java Virtual Machine (JVM*). The JVM exists as a process on your system and all of the ICEcore and portal software runs within

it. Consequently, the amount of memory available to the JVM process is a critical aspect of overall ICEcore performance.

The key JVM memory setting is the Java heap size, which roughly corresponds to the amount of memory available to the Java applications (such as ICEcore).

The *JVN heap size* setting in the *Java JDK location* window of the installer allows you to set the amount of physical memory you want to devote to the Java Virtual Machine.

The default configuration assumes that the JVM uses 1 GB (one gigabyte) of memory. It is possible to run ICEcore with less than 1 GB of assigned memory, but this is only applicable to very small test configurations, and is not suitable for production use. (In such a test configuration, 512 MB is the minimum amount of memory required to produce a functioning ICEcore application.)

A general rule is that no more than 75% of the available physical memory should be allocated to the JVM. This means that you need to account for other processes (including the operating system itself) that use memory. Databases, in particular, tend to be memory-intensive, so take special care when co-locating a database server with the ICEcore application server.

Java and ICEcore make heavy use of available memory to provide good performance. Larger deployments (large numbers of users and/or large numbers of documents) often need memory settings in excess of 2 GB to provide adequate performance. For these conditions, 64-bit hardware coupled with a 64-bit JVM are required.

WARNING: A JVM on a 32-bit system should never be configured to take more than 1.5 GB (1500 MB) of memory. Larger memory allocations require 64-bit servers.

ICEcore memory usage is driven by a number of factors:

1. **Number of sessions (users logged in):** Does not represent a significant amount of memory.
2. **Number of active/concurrent sessions:** Does not represent a significant amount of memory.
3. **Novell Teaming internal database caches:** Can be a significant fraction of memory use. These caches are separate from any caching or memory use by the database server itself, which might or might not be on the same server). Internal tuning settings have been set to handle a wide range of deployment sizes.
4. **Lucene index cache:** Can also be a significant consumer of memory. Large data stores (particularly with large files or a high number of files) can create a very large index. Installations with this type of demand should consider separating the Lucene Index Server to a dedicated standalone server for optimum performance (see "Installing a Standalone Lucene Index Server" on page 29).

1.7 Security Guidelines

- **Liferay Session Timeout:** The default session timeout is 2 hours (240 minutes). You can lengthen or shorten this by changing the *Liferay session timeout* setting in the *Network information* window of the installer.
- **Role-Based Access Control:** ICEcore controls all access to folders and entries by using role-based access controls. See "Adjusting Access Control for the Site" in the ICEcore Administration Guide to learn more about the default roles and access settings. Keep in mind that ICEcore is intended to be used primarily for the sharing of information, so many default access rights lean toward allowing at least universal read access.

- **Inbound E-mail:** You can configure ICEcore to read e-mail and post those e-mail messages as entries in a folder. Because e-mail is inherently insecure, there is no way to be sure that the sender is who they claim to be. ICEcore marks the entries posted by e-mail to alert users about their origin.

- **Web Services are enabled by default:** The default ICEcore installation allows authenticated access via Web services. If you are not using Web services, you can disable them by deselecting the Enable Web Services option in the *Network information* window of the installer.

- **WebDAV Authentication:** Authenticated access via WebDAV is via the Basic Authentication mechanism. Because of the weaknesses of Basic Authentication, secure (SSL) connections are recommended.

- **RSS feed URLs:** Because RSS readers are outside of the authentication system, the URL provided by ICEcore for an RSS feed embeds some authentication information about the user. This means that the RSS URL must be protected and not shared between users.

  For this reason, RSS is not recommended for use on highly sensitive data.

  To disable RSS feeds, deselect the *Enable RSS* option in the *RSS configuration* window of the installer during initial installation, or run a reconfiguration with the installer to disable RSS feeds after installation.

- **LDAP (directory service) Proxy User:** You can configure ICEcore (and the portal) to utilize information in the LDAP directory service to provide basic user account information (and group memberships). Access to the LDAP server is done via a configuration page that requires you to specify a username and password for access to the LDAP directory. This user should be created in the LDAP directory service with the minimum number of privileges needed to perform the job. In particular, all LDAP synchronization activities are one-way, so the proxy user only requires read access to the directory. We highly advise that you configure LDAP with a secure SSL connection.

- **LDAP Directory access is unencrypted by default:** See "Secure LDAP/eDirectory Setup" in the ICEcore Administration Guide for information about configuring ICEcore to use SSL when communicating with the LDAP directory.

- **File System Repositories contains unencrypted data:** See "File System Planning" on page 14 for details about how ICEcore uses the local file system for data storage. These directories contain uploaded information in various formats (both native file formats and potentially a number of rendered formats (such as cached HTML versions of files, thumbnails, and RSS feeds) as well as archived data.

  These files are managed exclusively by the ICEcore application software and the file system protections should be set to protect those directories from unauthorized access.

- **Database access is unencrypted by default:** Depending on your local security guidelines, you might want to encrypt the database connections between the ICEcore and Portal software and their respective databases. SSL encrypted data between the applications and database servers imposes a performance penalty because of the increased overhead of encrypting and decrypting the retrieved data.

  Support for this is highly dependent on the database client drivers and JDBC connector support and how you are configuring your client and server certificates. You should check with the database vendors on how to set up SSL connections on both the client and server sides of the connection. You need to at least update the *JDBC URL* selections in the Database selection window of the installer. For example, for MySQL you might add `useSSL=true&requireSSL=true` to the options part of the URL.

- **Mirrored Folder Proxy User:** See "Configuring Mirrored Folders" in the ICEcore Administration Guide for more information about the mirrored folder feature of the Enterprise version of ICEcore.

  You can configure ICEcore to use server directories (either in the local file system or via file sharing) as repositories for ICEcore folders. Because the ICEcore application server is accessing those directories, the user ID that the application server runs as acts as a proxy user for all file system access (that is, the file system only sees one user accessing the files on behalf of all ICEcore users who have access to the ICEcore folder). This proxy user should be used to configure any local file system access (or shared file access) appropriately.

  If you configure a mirrored folder to a WebDAV or Microsoft SharePoint* directory, the resource driver is configured to use a proxy username and password. The same access control practices should be applied to these resources as are used with the file system resource driver.

- **Conferencing account password is stored in configuration file:** If you are using Conferencing, the password to a Conferencing account is stored in:

  `WEB-INF/classes/config/ssf.properties`

- **Password Storage in the Server File System:** A number of application accounts and passwords are stored in the file system. These files should be protected against unauthorized access on the server.

  The `installer.xml` file contains a majority of the account and password information. You should protect this file accordingly. The installer's *Reconfigure* option uses this information to create and update a number of configuration files within the application software directory tree. These files are outlined below:

  - Database user IDs and passwords for the portal and ICEcore software are stored in XML files in the following directories:

    `conf/Catalina/localhost/ssf.xml (ICEcore)`

    `conf/Catalina/localhost/ROOT.xml (Liferay Portal)`

  - Mirrored folder resource drivers to WebDAV/Sharepoint shares store the proxy user and passwords in:

    `WEB-INF/classes/config/ssf.properties`

  - The e-mail access (both inbound and outbound) might contain usernames and passwords for authentication (such as, authenticated SMTP). These are stored in:

    `conf/Catalina/localhost/ssf.xml (ICEcore)`

    `conf/Catalina/localhost/ROOT.xml (Liferay Portal)`

  Some application accounts and passwords are stored in the database. These are protected by application access controls, but are available if access to the database is obtained through other means, such as an LDAP proxy user and password.

1.8 Load-Balancing and Clustering ICEcore

NOTE: In this release, ICEcore does not support session replication across cluster nodes.

IMPORTANT: If you have more than one clustered installation of ICEcore in your network, each cluster group needs a unique multicast group IP address to prevent them from interfering with each other.

To set up a scalable, clustered ICEcore configuration, use the following steps as a guideline:

1 Set up a shared file storage that is accessible to all nodes.

ICEcore's Simple File Repository is safe for use in a clustered environment with shared storage that is accessible to all nodes. Setting up shared storage is a platform-specific task.

2 Install and configure the Lucene Index Server.

See "Installing a Standalone Lucene Index Server" on page 29 for how to place the Lucene Index Server on a dedicated server system. All cluster nodes share this index server.

3 Make sure that the time is the same on all nodes in the cluster (using synchronized time services for your cluster is highly advised).

4 Install the ICEcore/portal bundle kit on each cluster node.

IMPORTANT: Because all the nodes share the same database server, it is important not to execute the database-initialization SQL scripts more than once (that is, use the installer's *Reconfigure* option rather than the *New installation* option on all but the first cluster node).

Each node in the cluster can use the same `installer.xml` file (generated by the installer during initial installation). This assures that cluster nodes are configured uniformly.

The following settings must be uniform across the cluster:

- Configure the database connection settings on each node to use the same database.
- Set the file system settings to all point to the same shared file storage server.
- Set the Network settings (name, port, etc.) to the name of the LOAD BALANCER system.
- In the *Lucene configuration* window of the installer, select *server* in the *Lucene configuration type* drop-down list and set the *Host* selection to the hostname of the machine on which you installed the Lucene Index Server. See "Installing a Standalone Lucene Index Server" on page 29.

5 Set the portal up to work in a clustered environment.

If you need to set more advanced configurations, use `cache.cluster.properties` instead of `cache.cluster.multicast.ip` in the steps below. For more information, refer to Liferay's documentation at wiki.liferay.com (http://wiki.liferay.com/index.php/Clustering) and http://wiki.liferay.com (http://wiki.liferay.com/index.php/Liferay_FAQ). Liferay provides additional high availability information at wiki.liferay.com (http://wiki.liferay.com/index.php/High_Availability_Guide).

5a Edit the `/webapps/ROOT/WEB-INF/classes/portal-ext.properties` file by uncommenting the following two lines:

```
cache.event.listeners=com.opensymphony.oscache.plugins.clusters
upport.JavaGroupsBroadcastingListener
cache.cluster.multicast.ip=231.12.21.100
```

5b Edit the /webapps/ROOT/WEB-INF/classes/cache-multi-vm-ext.properties file by uncommenting the following two lines:

```
cache.event.listeners=com.opensymphony.oscache.plugins.clusters
upport.JavaGroupsBroadcastingListener
cache.cluster.multicast.ip=231.12.21.101
```

6 Configure the load balancer.

There are a variety of load balancing solutions that work with J2EE deployments. The following example configuration uses the balancer module built into the newer Apache

(version 2.2.4), and is based on a widely used sticky session technique. ICEcore does not support session sharing/replication among Tomcat instances.

6a Edit Tomcat's `server.xml`.

Add `jvmRoute="jvm<n>"` to the `<Engine name="Catalina" ...>` element, where `<n>` should be an integer unique to each Tomcat instance. For example, assuming you have two Tomcat instances in a cluster, the Catalina engine element should look like the following in each `server.xml` respectively (`jvm<n>` is the name of the worker as declared in the load balancer):

```
<Engine name="Catalina" defaultHost="localhost"
jvmRoute="jvm1">
<Engine name="Catalina" defaultHost="localhost"
jvmRoute="jvm2">
```

6b Edit Apache's `httpd.conf file` (in the `<apache installation>/conf` directory).

1. Uncomment the following three lines:
```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
LoadModule proxy_balancer_module modules/
mod_proxy_balancer.so
```

2. Append the following section to the end of the file:
```
<Location /balancer-manager>
SetHandler balancer-manager
Order deny,allow
Deny from all
Allow from 127.0.0.1
</Location>
<Proxy balancer://aspenCluster>
BalancerMember ajp://<tomcat host name 1>:8009 route=jvm1
BalancerMember ajp://<tomcat host name 2>:8009 route=jvm2
</Proxy>
<Location />
ProxyPass balancer://aspenCluster/ stickysession=JSESSIONID
</Location>
```

Substitute the real Tomcat hostnames for `<tomcat host name 1>` and `<tomcat host name 2>`.
If you have more than two Tomcat instances in the cluster, make an additional line for each.

7 Configure Hibernate second-level cache to use cluster-safe distributed cache. To do this, rename the `ehcache-hibernate.xml` file in the `webapps/ssf/WEB-INF/classes/config` directory to something else, say, `ehcache-hibernate-non-clustered.xml`. Then, rename the `ehcache-hibernate-clustered.xml` file in the same directory to `ehcache-hibernate.xml`.

8 Start the Lucene Index Server, and then start the application cluster nodes.

1.9 Running the Installer

If you are doing an upgrade, use your existing `license-key.xml` and `installer.xml` files by placing them in the same directory as the installer program (the unzipped ICEcore kit). If you are doing a new installation, use your `license-key.xml` by placing it in the same directory as the installer program. The `license-key.xml` file required to install the product is provided with the software kit (but is not included in the kit).

NOTE: You can edit the `installer.xml` file directly and run the installer's Reconfigure option to reconfigure ICEcore. See Appendix A, "A Sample installer.xml File," on page 35.

---

1 Run the installer (on Linux, do a `chmod +x installer-liferay.linux` to make the installer executable).

The installer is named one of the following according to your operating system:

- `installer-liferay.exe`
- `installer-liferay.linux`

NOTE: On Linux, you can run a console-based version of the installer (has all the same options as the GUI-based installer) using the following command:

`installer-liferay.linux -console`

---

2 In the *Welcome* window, select the procedure you want to perform, then click *Next*:
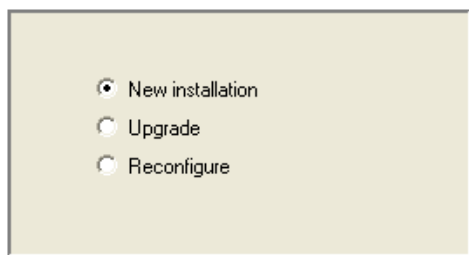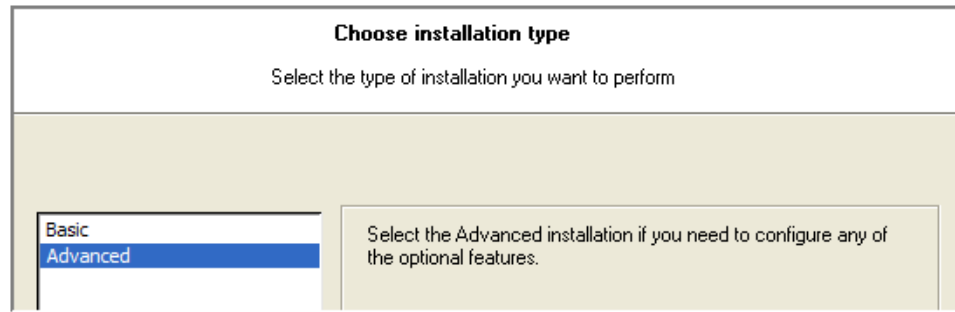
- *New installation*

  Use this option to perform an initial installation of ICEcore. The rest of this procedure walks you through an advanced initial installation to show you all the available options.

- *Upgrade*

  Use this option to perform an upgrade of an existing ICEcore application.

- *Reconfigure*

  Use this option to reconfigure an existing ICEcore application.

3 Read the license agreement, select the *I accept the license agreement* option to indicate your acceptance of this agreement, then click *Next*.
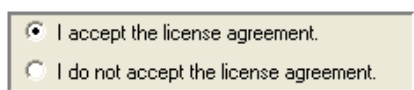
4 In the *Choose installation type* window, select the type of installation you want to perform, then click *Next*:

Choose installation type
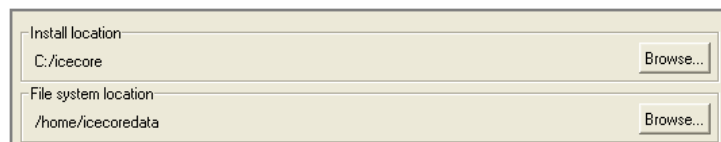
Select the type of installation you want to perform

| Basic | Select the Advanced installation if you need to configure any of |
| Advanced | the optional features. |

- ◆ *Basic*: Use this option if you are do not need to configure any optional features.

- ◆ *Advanced*: Use this option if you need to configure any of the optional features.
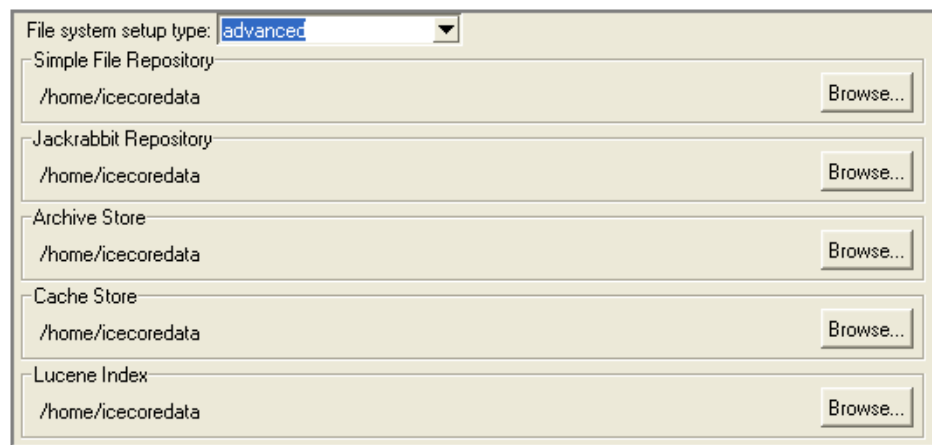
5 In the *Install location* window, use the associated *Browse* buttons to change the default location for the installation or file system:

Install location
C:/icecore   Browse...
File system location
/home/icecoredata   Browse...

6 If you need to change the locations of any of the file system components, select *advanced* from the *File system setup type* drop-down list to activate the *Browse* buttons for these components, make the necessary changes, then click *Next*.

File system setup type: advanced ▼
Simple File Repository
/home/icecoredata   Browse...
Jackrabbit Repository
/home/icecoredata   Browse...
Archive Store
/home/icecoredata   Browse...
Cache Store
/home/icecoredata   Browse...
Lucene Index
/home/icecoredata   Browse...

7 In the *Liferay data location* window, click *Browse* if you need to change the default *Root location* for this data, then click *Next*.

Root location
/home/icecoredata/liferay   Browse...

8 In the *Network information* window, specify the network values for how you want ICEcore to appear on the network. See "Network Planning" on page 14.

9 In the *Database selection* window, configure your databases and select whether or not to create the databases during install. The database server you select must already be installed.

If your DBA created the databases by hand (required for Oracle), select the *My DBA has already created the databases* option. For more information, see "Database Planning" on page 11.

To configure your databases:

  9a Select the database from the *Database type* drop-down list.

  9b Enter your *Username*, *Password*, and *JDBC URL* for the icecore and Liferay databases, then click *Next*.

10 In the *Java JDK location* window, click *Browse* to select the *Java Home* directory location (if not detected automatically), type in the *JVM heap size* value if you require more than the default 1 G (use a small "g" with no spaces), then click *Next*.
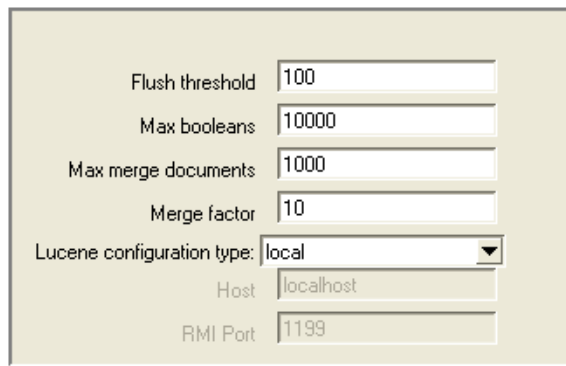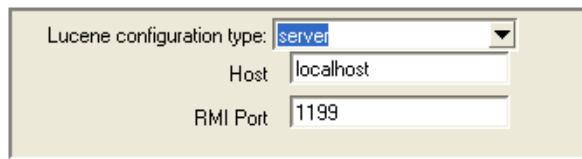
11 In the *Lucene configuration* window, edit the fields to configure the Lucene index.

Flush threshold	100
Max booleans	10000
Max merge documents	1000
Merge factor	10
Lucene configuration type:	local
Host	localhost
RMI Port	1199

To run the Lucene index as a server, select *server* from the *Lucene configuration type* drop-down list, type in the *Host*, and edit *RMI Port* value if the default port cannot be used.
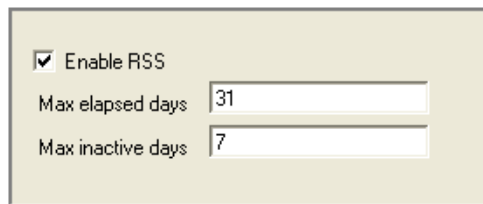
Lucene configuration type:	server
Host	localhost
RMI Port	1199

You can run the Lucene as a server from the same system as ICEcore or from a remote system. You need to install the Standalone Lucene Index Server before running ICEcore. See .

12 In the *RSS configuration* window, edit the fields to configure the ICEcore RSS feeds, or deselect the *Enable RSS* option to disable RSS feeds, then click *Next*.

☑ Enable RSS	
Max elapsed days	31
Max inactive days	7

RSS feed URLs: Because RSS readers are outside of the authentication system, the URL provided by ICEcore for an RSS feed embeds some authentication information about the user. This means that the RSS URL must be protected and not shared between users.

For this reason, RSS is not recommended for use on highly sensitive data.

13 In the *Outbound email configuration* window, edit the fields to configure your outbound e-mail settings. Outbound configuration requires the basic information for generating SMTP mail messages: server, port, and optional authentication information.

The *Username* field is the address used for any e-mail notifications generated by the system. This name is also used for authenticated access to the SMTP server. To support e-mail sent by users, either the ICEcore server or the authenticated user must be authorized to use different *From* addresses.

For more information, see the "Mail Setup" section of the *ICEcore Administration Guide*.

14 In the *Inbound email configuration* window, edit the following fields to configure your inbound e-mail settings, then click *Next*.

For inbound mail (postings) you need to specify either *pop3* or *imap* (*pop3s* or *imaps* for SSL connections to mail servers) from the *Email protocol* drop-down list. These settings are not used until you enable incoming e-mail in ICEcore.

ICEcore posting uses real or alias e-mail addresses in your folders, according to whether you select the *Use aliases* option:

- ◆ When you use aliases for posting, the system accesses a single e-mail account (posting account). ICEcore routes e-mail from this account to individual folders based on unique e-mail addresses (aliases) which are assigned to individual folders and mapped to the posting account.

- ◆ When you use real e-mail addresses for each folder, you need to create multiple posting accounts using your normal e-mail system management tools. Each posting e-mail address is used for a single folder.

Create the posting accounts using your normal e-mail system management tools. ICEcore needs a host, port, e-mail account ID, and password for the posting feature to work.

15 In the *Presence configuration* window, select the *Enable presence* option to link your ICEcore and Conferencing servers together. You need to obtain the values for the following fields from an installed and configured Conferencing server. If you do not know these values now, you can run the installer's *Reconfigure* option to add this information later. Click *Next* to continue.

16 In the *iChain configuration* window, select *Enable iChain* to use the iChain Single Sign-on Support (SSO), set the Logoff URL to the address used to trigger an iChain logoff, set the IP address of the iChain Proxy server, then click *Next*.

Only transactions from that address and the localhost are allowed. For more information, see "Integrating with iChain" on page 16.

17 Mirrored folders are local/shared directories that are exposed within ICEcore. You must configure the directories here before they are available in ICEcore. For security reasons the set of people who can map ICEcore folders to these shared directories should be limited.

There are three mirrored folder types:

- file - for normal file systems
- webdav - for WebDAV servers that support basic authentication
- sharepoint - for Microsoft Sharepoint WebDAV access

NOTE: Mirrored Folders are available only in the Enterprise Edition.

A mirrored folder is a ICEcore folder that uses a server file system directory as its file storage area instead of the normal ICEcore repository. Typically the directory is a file share, accessed via normal file sharing mechanisms. ICEcore attempts to keep its knowledge about the folder contents in sync with whatever is in the directory.

Because of the way mirrored folders are configured, they are intended to bring common (relatively static) file shares into the ICEcore environment. They are not appropriate for individual file shares. You can use ICEcore to add additional metadata around these files, including ICEcore-specific access controls.

The following fields specify the mirrored folder resource drivers. These drivers then make the files in the specified directory available to the ICEcore folders. It is important to note that the ICEcore process accesses the directories and that the user ID for ICEcore acts as a proxy for all ICEcore users.

If you need more than three mirrored folders, you need to edit the `installer.xml` file directly and run the installer's *Reconfigure* option. For more information, see the "Configuring Mirrored Folders" section in the ICEcore Administration Guide.

Click *Next* to continue.

18 In the *Ready to install* window, click *Next* to begin the installation or use the *Back* button to review the current configuration of the installer.

1.10 Installing a Standalone Lucene Index Server

This section covers how to install the Lucene Index Server as a standalone server.

NOTE: You need to install JDK (Sun JDK 1.5.0_011 or higher) before installing the Lucene Index Server. Linux systems need to have a minimum open file limit of 4096.

To install a Standalone Lucene Index Server for a new ICEcore application:

1 Copy the Lucene installer to the system on which you want to install the standalone Lucene Index Server.

The Lucene installers are in the ICEcore kit you unzipped:
- lucene-installer.exe
- lucene-installer.linux

2 Run the Lucene installer.

3 In the *Welcome* window, select the procedure you want to perform, then click *Next*:

- *New installation*

  Use this option to perform an initial installation of the Standalone Lucene Index Server.

- *Upgrade*

  Use this option to perform an upgrade of an existing Standalone Lucene Index Server.

- *Reconfigure*

  Use this option to reconfigure an existing Standalone Lucene Index Server.

4 Read the license agreement, select the *I accept the license agreement* option to indicate your acceptance of this agreement, then click *Next*.

5 In the *Install location* window, click *Browse* if you need to change the default *Install location* for the luceneserver, then click *Next*.

6 In the *Lucene configuration* window, edit the fields to configure the Lucene index.

Type in the *Host*, and edit *RMI Port* value if the default port cannot be used. Click *Next* to continue.

You can run the Lucene as a server from the same system as ICEcore or from a remote system. You need to install the Standalone Lucene Index Server before running ICEcore. See "Installing a Standalone Lucene Index Server" on page 29.

7 In the *Ready to install* window, click *Next* to begin the installation or use the *Back* button to review the current configuration of the installer.

To install a Standalone Lucene Index Server for an Existing Novell Teaming Application:

1 Shut down ICEcore on the ICEcore server (see).

2 Run the Lucene installer (see above procedure).

3 On the standalone server, check the `lucene-server.properties` file and find the setting for `index.root.dir` (you can edit this setting to change the location of the root directory).

4 Create a `liferay.com` subdirectory in the `index.root.dir` directory.

5 On the ICEcore server, look at the `ssf.properties` file and find the `data.root.dir` setting.

6 Copy all the files from the *data.root.dir*`/lucene/liferay.com` directory on the ICEcore server to the *index.root.dir*`/liferay.com` directory you just created on the standalone machine.

IMPORTANT: If you are using FTP, make sure to copy in binary mode.

7 For UNIX systems, make sure permissions are set so that the ICEcore user has full access to these files and the `liferay.com` directory.

8 Start the server at the end of the installation (if it is not already started):

  8a Change directories to `lucene-server-installation-dir`

  8b Run the `rmiregistry-startup` script in the `bin` directory

  8c Run the `indexserver-startup` script in the `bin` directory

9 Restart ICEcore (see).

1.11 Starting and Stopping ICEcore

1.11.1 Starting ICEcore/Liferay

NOTE: These are Liferay-only instructions.

On Windows:

From the command prompt, change directories to the Liferay bin directory, and then run the startup command:

```
> cd C:\yourinstall\liferay-portal-tomcat-5.5-jdk5-4.3.0\bin\
> startup.bat
```

WARNING: On Windows, you need to run the `startup.bat` command from the `bin` directory.

On Linux:

```
/yourinstall/liferay-portal-tomcat-5.5-jdk5-4.3.0/bin/icecore
start
```

NOTE: Depending on your system configuration, it can take more than of 60 seconds before ICEcore/Liferay starts accepting Web transactions. Initial transactions also tend to be slower as various caches load into RAM. These delays are amplified somewhat when working with a new installation or updated software because the JSPs are recompiled as they are referenced.

In Windows, startup is complete when the Tomcat window displays the following message:
```
INFO:Server Startup in ##### ms
```

In Linux, the system does not automatically open a monitor window. You can start one by navigating to your *your-installation*/life-ray-portal-tomcat-5.5-jdk-4.30/logs directory and executing `tail --f catalina.out`. Startup is complete when this window displays:
```
INFO:Server Startup in ##### ms
```

1.11.2 Setting up ICEcore to Start on System Startup

On Windows:

1 Enter `C:\yourinstall\liferay-portal-tomcat-5.5-jdk5-4.3.0\bin\service.bat install icecore`

2 Use the Services Control Panel to configure the service to your needs.

   2a Select *Control Panel > Administration Tools > Services*.

   2b Right-click the *ApacheTomcat icecore* service and select *Properties*.

   2c Change the *Startup Type* to *Automatic* and click *OK*.

   2d Set your firewall to allow port 8080.

On Linux, from the root account:

1 Enter `cp /yourinstall/liferay-portal-tomcat-5.5-jdk5-4.3.0/bin/icecore   /etc/init.d`

2 Enter `chkconfig --add icecore`

1.11.3 Log Files and Monitoring

On Windows:

A Tomcat window appears when you issue the `startup.bat` command. System messages and error messages appear here.

On Linux:

Unlike Windows, the Tomcat process starts as a background process and no window appears. To monitor the messages in real time, enter:

```
tail -f /yourinstall/liferay-portal-tomcat-5.5-jdk5-4.3.0/logs/
catalina.out
```

1.11.4 Stopping ICEcore

On Windows:

Enter `C:\yourinstall\liferay-portal-tomcat-5.5-jdk5-4.3.0\bin\shutdown.bat`

On Linux:

Enter `/yourinstall/liferay-portal-tomcat-5.5-jdk5-4.3.0/bin/icecore stop`

1.12 ICEcore Backup and Restore Procedures

If your system has a period of minimal activity (such as at 1:00 AM) you can do the backups without taking the system offline. If you perform this step every night, it is usually possible to restore without problems. The worst case scenario would be that the Lucene directory was caught adding an entry when the snapshot was taken. This can be remedied by re-building the search index from the *ICEcore Administration* portlet.

Method 1:

1 Take the system down on a scheduled basis.

2 During the down period:

   2a Take a snapshot of the database

   2b Make a backup copy of the Lucene index, default location is:

      `/opt/icecore/luceneserver/liferay.com`

   2c Make a backup of the file repositories that are in use, for example:

      `/home/icecoredata/filerepository`

3 Restart your ICEcore system.

NOTE: This approach is the most conservative. It guarantees a consistent snapshot of your whole system. However, doing a backup of the file repository can take a long time depending on your backup approach.

Method 2:

1 Take the system down on a scheduled basis.

2 During the down period:

   2a Take a snapshot of the database

   2b Make a backup copy of the Lucene index

3 Bring the system back online.

4 After the system is running again, start the backup of the file repositories.

NOTE: This approach minimizes the down time. However, it is possible that during a restore after a catastrophic failure the files may be out of synchronization with the metadata. This does cause any problems when running the system, but users may find that slightly newer versions of the files exist in the restored system than were present at the time of the database snapshot. This is usually a good thing.

A Sample installer.xml File

A sample `installer.xml` file is shown below:

NOTE: For Windows, use forward slashes for all directory paths in the `installer.xml` file.

```
<!--                                                           -->
<!--          ICEcore Installation Configuration File          -->
<!--                                                           -->
<ICEcoreConfig version="4">
      <Install type="0"/>
      <Environment>
          <!--  The path to JAVA_HOME -->
            <JDK JAVA_HOME="" type="Sun" />
            <!--  What userid to run as (Linux-only)  -->
            <!--  Also what userId and groupId to use -->
            <!--  as owner of the data directories.   -->
            <Ids userId="" groupId="" />
            <!--  Where does the ICEcore software reside?  -->
            <SoftwareLocation path="" />
      </Environment>
    <!--                                                      -->
    <!--                 Network Settings                     -->
    <!--                                                      -->
    <!--  The host name or IP address of the server must be   -->
    <!--  specified here.  The default, localhost, is only    -->
    <!--  appropriate for test configurations with no remote  -->
    <!--  access.                                             -->
    <!--                                                      -->
    <!--  If you have a dedicated server, setting the port    -->
    <!--  to "80" and/or securePort to "443" avoids having    -->
    <!--  to specify a port number in browser URLs.           -->
    <!--                                                      -->
    <!--  NOTE: On Linux/UNIX based systems, it is very dangerous -->
    <!--  to run this server on any port number less than 1024,   -->
    <!--  since that would require the web server to run as the   -->
    <!--  root user.  If you must run on a port lower than 1024,  -->
    <!--  please see the ICEcore documentation referring to port  -->
    <!--  forwarding via iptables.                            -->
    <!--                                                      -->
    <!--  If you use some sort of port forwarding or proxying  -->
    <!--  (e.g., iptables or iChain), use the listenPort and   -->
    <!--  secureListenPort settings to specify the port numbers -->
```

```
<!-- to use on the SERVER. The port and securePort settings  -->
<!-- should refer to what external users use to access        -->
<!-- the system.  (If not specified, the listen ports use     -->
<!-- the corresponding port/securePort setttings.)            -->
<!--                                                           -->
<!-- A default certificate is supplied for SSL connections.   -->
<!-- You should replace this with your certificate.           -->
<!-- This can be placed in the conf/.keystore file or         -->
<!-- change the keystoreFile attribute to point to the        -->
<!-- certificate.                                              -->
<!--                                                           -->
<!--                                                           -->
<!-- OES2 servers will need to reassign the ajpPort and        -->
<!-- shutdownPorts as they will collide with other software.  -->
<!-- ajpPort="8010" and shutdownPort="8011" seem to work      -->
<!-- well.  See the Install Guide Network Planning section    -->
<!-- for more details on port conflicts.                      -->
<!--                                                           -->

<Network>
    <Host name="localhost"
        port="8080"        listenPort=""
        securePort="8443"    secureListenPort=""
        shutdownPort="8005"
        ajpPort="8009"
        keystoreFile=""
        />
    <WebServices enable="true" />
    <Liferay sessionTimeoutMinutes="240" />
    <TomcatAccessLog enable="false" />
</Network>

<!--                                                           -->
<!--            Memory (RAM) Settings                          -->
<!--                                                           -->
<!-- ICEcore requires a minimum of 512m to operate.           -->
<!-- 1g is recommended for basic production operation.        -->
<!-- Generally do not allocate more than 75% of available     -->
<!-- physical memory to ICEcore.                               -->
<!--                                                           -->
<!-- Specify amounts as  Nm for N megabytes (e.g., 1500m) or  -->
<!-- Ng for N gigabytes (e.g., 3g).                           -->
<!--                                                           -->

<Memory>
    <JavaVirtualMachine mx="1g" />
```

```
</Memory>

<!--                                                      -->
<!--              File System Configuration               -->
<!--                                                      -->
<!-- Modify the configName to your desired configuration  -->
<!-- in the FileSystem element below. You must set the    -->
<!-- configName to the exact configuration in the file:   -->
<!--     basic      - Simple one-directory setup          -->
<!--     advanced   - Advanced multiple-directory setup   -->
<!--                                                      -->
<!-- WARNING!  Changing a directory path does NOT relocate -->
<!-- data after the product has been installed.  You must  -->
<!-- coordinate any changes here with appropriate file system -->
<!-- modifications (including directory protections and   -->
<!-- ownership where applicable).                         -->
<!--                                                      -->

<FileSystem configName="basic">
  <!-- The basic configuration only requires that you  -->
  <!-- specify a root directory for the data and        -->
  <!-- ICEcore takes care of the rest.                  -->
  <!--                                                   -->
  <!-- IMPORTANT NOTE FOR WINDOWS USERS: Use forward    -->
  <!-- slash as directory separator!                    -->

  <Config id="basic">
    <RootDirectory path="/home/icecoredata"  />
  </Config>

  <!-- With the advanced configuration you can specify     -->
  <!-- individual directory locations.                     -->
  <!-- Add path attributes for those directories you       -->
  <!-- want to locate elsewhere. For example,              -->
  <!-- <SimpleFileRepository path="/share/icecoredata" />  -->
  <!--                                                     -->
  <!-- IMPORTANT NOTE FOR WINDOWS USERS: Use forward slash -->
  <!-- as directory separator!                             -->

  <Config id="advanced">
    <RootDirectory path="/home/icecoredata"  />
    <SimpleFileRepository />
    <JackrabbitRepository />
    <ArchiveStore   />
    <CacheStore     />
    <LuceneIndex    />
```

```
        </Config>

            <!-- The liferay data dirs configuration allows you to
-->
            <!-- change the directories that liferay uses for
-->
            <!-- its internal data directories.  The value typically
-->
            <!-- is a subdirectory where the other ICEcore data
-->
            <!-- resides.                                          -->
            <!--                                                   -->
            <!-- Early beta sites may want to use a different setting
-->
            <!-- called "default", which means the *default directory*
-->
            <!-- of whatever user id the server runs as.  Generally
-->
            <!-- speaking this is not a great idea, and the option
-->
            <!-- is there solely to accomodate beta sites who predate
-->
            <!-- the ability to explicitly set where Liferay stored
-->
            <!-- its files.                                        -->
            <!--                                                   -->
            <!-- NOTE: If you change the use setting after you have
-->
            <!-- already started using liferay, you need to copy
-->
            <!-- all the data from the default dirs to the new dirs
-->
            <!-- BEFORE you restart the server!                  -
->
        <!--                                                   -->
        <!-- IMPORTANT NOTE FOR WINDOWS USERS: Use forward slash   -->
        <!-- as directory separator!                           -->

            <LiferayDataDirs use="custom">
              <RootDirectory path="/home/icecoredata/liferay"  />
              <DocumentLibraryRoot dir="documentlibrary/root" />
              <DocumentVersionLibrary dir="documentlibrary/vroot" />
              <JackRabbit dir="jackrabbit" />
              <AutoDeploy dir="deploy" />
              <Lucene dir="lucene" />
            </LiferayDataDirs>

    </FileSystem>
```

```
<!--                                                          -->
<!--                Database Configuration                    -->
<!--                                                          -->
<!-- Modify the configName to your desired configuration      -->
<!-- in the Datatabase element below.  You must set the       -->
<!-- configName to the exact configuration in the file:       -->
<!--    MySQL_Default      - For MySQL                         -->
<!--     SQLServer_Default  - For Microsoft SQL Server         -->
<!--                                                          -->
<!-- Change the Resources for the configuration you chose      -->
<!-- (the defaults are pretty good for a simple configuration  -->
<!-- with the database running locally, but you will probably  -->
<!-- have different passwords!).                               -->

    <Database configName="MySQL_Default">

        <!--                                                  -->
        <!--                  MySQL_Default                   -->
        <!--                                                  -->

        <Config id="MySQL_Default" type="MySql">
            <Resource for="liferay"
                driverClassName="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost:3306/
lportal?useUnicode=true&amp;characterEncoding=UTF-8"
                username="root"
                password="root"
            />
            <Resource for="jboss"
                driverClassName="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost:3306/
jbossportal?useUnicode=true&amp;characterEncoding=UTF-8"
                username="root"
                password="root"
            />
            <Resource for="icecore"
                driverClassName="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost:3306/
sitescape?useUnicode=true&amp;characterEncoding=UTF-8"
                username="root"
                password="root"
            />
        </Config>

        <!--                                                  -->
        <!--                  SQLServer_Default                -->
```

```
        <!--                                              -->

        <Config id="SQLServer_Default" type="SQLServer">
            <Resource for="liferay"
                driverClassName="net.sourceforge.jtds.jdbc.Driver"
                url="jdbc:jtds:sqlserver://localhost/
lportal;SelectMethod=cursor"
                username="sa"
                password="sa"
             />
            <Resource for="jboss"
                driverClassName="net.sourceforge.jtds.jdbc.Driver"
                url="jdbc:jtds:sqlserver://localhost/
jbossportal;SelectMethod=cursor"
                username="sa"
                password="sa"
             />
            <Resource for="icecore"
                driverClassName="net.sourceforge.jtds.jdbc.Driver"
                url="jdbc:jtds:sqlserver://localhost/
sitescape;SelectMethod=cursor"
                username="sa"
                password="sa"
             />
        </Config>

        <!--                                              -->
        <!--                  Oracle_Default               -->
        <!--                                              -->

        <Config id="Oracle_Default" type="Oracle">
            <Resource for="liferay"
                driverClassName="oracle.jdbc.driver.OracleDriver"
                url="jdbc:oracle:thin:@//localhost:1521/orcl"
                username="lportal"
                password="pw"
             />
            <Resource for="icecore"
                driverClassName="oracle.jdbc.driver.OracleDriver"
                url="jdbc:oracle:thin:@//localhost:1521/orcl"
                username="sitescape"
                password="pw"
             />
        </Config>

    </Database>
```

```
<!--                                                             -->
<!--              Lucene Configuration Settings                  -->
<!--                                                             -->
<!--                                                             -->
<!--      The Lucene index can be run "local" (within           -->
<!--      the context of this application) or "server"          -->
<!--      (run as its own server).  Additionally it             -->
<!--      can be run as its own server on this system           -->
<!--      or on a remote system.                                -->
<!--                                                             -->
<!--      Note: The rmi port need only be set if running        -->
<!--      in server mode. (And then, only if the default        -->
<!--      port cannot be used.                            -->

<Lucene luceneLocation="local">
    <Resource
        lucene.index.hostname="localhost"
              lucene.flush.threshhold="100"
              lucene.max.booleans="10000"
              lucene.max.merge.docs="1000"
              lucene.merge.factor="10"
              lucene.rmi.port="1199"
    />
</Lucene>

<!--                                                             -->
<!--              RSS Configuration Settings                     -->
<!--                                                             -->
<!--  RSS feeds are inherently insecure as they do not          -->
<!--  use the standard authentication mechanism.  If you        -->
<!--  do not want to have users create RSS subscriptions,       -->
<!--  set the enable property to "false".                       -->
<!--                                                             -->
<!--  If you have RSS feeds enabled, you can tune the maximum    -->
<!--  number of days to keep in the feed and the maximum        -->
<!--  number of days to keep updating a feed when there are     -->
<!--  no clients reading it.                                    -->
<!--                                                             -->
  <RSS enable="true">
        <Feed max.elapseddays="31" max.inactivedays="7" />
  </RSS>

<!--                                                             -->
<!--              Email Configuration Settings                  -->
<!--                                                             -->
<!--  Edit the Outbound and Inbound settings as required.       -->
```

```xml
<!--                                                       -->
<!--                                                       -->
<!--  For inbound mail (postings) you need to specify either -->
<!--  pop3 or imap (or pop3s/imaps for SSL connections to  -->
<!--  mail servers), and fill out the settings for which one -->
<!--  you choose.  These settings are not used until you    -->
<!--  enable incoming email within the product.  If you do  -->
<!--  not plan on using inbound email, you can ignore these -->
<!--  settings.                                             -->
<!--                                                       -->
<!--                                                       -->
<!--                                                       -->

<EmailSettings>
    <Outbound>
        <Resource
                    mail.transport.protocol="smtp"

            mail.smtp.host="mailhost.yourcompany.com"
            mail.smtp.user="icecore@yourcompany.com"
            mail.smtp.password=""
            mail.smtp.auth="false"
            mail.smtp.port="25"

            mail.smtps.host="mailhost.yourcompany.com"
            mail.smtps.user="icecore@yourcompany.com"
            mail.smtps.password=""
            mail.smtps.auth="false"
            mail.smtps.port="465"

        />
     </Outbound>

    <Inbound useAliases="false">
        <Resource
            mail.store.protocol="pop3"

            mail.pop3.host="localhost"
            mail.pop3.auth="true"
            mail.pop3.user="popEmailUserId"
            mail.pop3.password="passwordHere"
            mail.pop3.port="110"
            mail.pop3s.host="localhost"
            mail.pop3s.auth="true"
            mail.pop3s.user="popEmailUserId"
            mail.pop3s.password="passwordHere"
```

```
                mail.pop3s.port="995"
                mail.imap.host="localhost"
                mail.imap.auth="true"
                mail.imap.user="imapEmailUserId"
                mail.imap.password="passwordHere"
                mail.imap.port="143"

                mail.imaps.host="localhost"
                mail.imaps.auth="true"
                mail.imaps.user="imapEmailUserId"
                mail.imaps.password="passwordHere"
                mail.imaps.port="993"
            />
        </Inbound>
    </EmailSettings>

    <!--                                                          -->
    <!--            Presence Configuration Settings               -->
    <!--                                                          -->
    <!--                                                          -->
    <!--  Use those settings to link your ICEcore and Conferencing -->
    <!--  servers together.   Leave the presence.service.enable   -->
    <!--  setting at false if you do not use the conferencing     -->
    <!--  real time software.                                     -->
    <!--                                                          -->
    <!--  The values to fill in this section are obtained when    -->
    <!--  ICEcore Conference has been installed and configured.   -->
    <!--  If you do not know them now, you can update this        -->
    <!--  configuration later and apply the settings.             -->
    <!--                                                          -->
    <!--  The jabber.server is the IP or host of the ICEcore      -->
    <!--  Conference  XML Router.                                 -->
    <!--                                                          -->
    <!--  The default ICEcore Conference admin.id is admin, and   -->
    <!--  default password is also admin. Change the admin.passwd -->
    <!--  to match the ICEcore Conference administrator password  -->
    <!--  you have set.                                           -->
    <!--                                                          -->
    <!--  The jabber.domain is the host name of the XML router    -->
    <!--                                                          -->
    <!--  The community.id is the name of the community you       -->
    <!--  created with the ICEcore Conference console.            -->
    <!--                                                          -->
    <!--  Change the hostname part of the conferencing.url to     -->
    <!--  the ip or host of the ICEcore Conference Web Portal     -->
    <!--  (do not change the port or the rest the URL).           -->
```

```
<!--                                                           -->

<Presence>
    <Resource
        presence.service.enable="false"
       presence.service.jabber.server="zon-server.yourcompany.com"
        presence.broker.admin.id="admin"
        presence.broker.admin.passwd="admin"
        presence.broker.jabber.domain="newzon"
        presence.broker.default.community.id="yourcommunity"
        presence.broker.zon.url="http://zon-
server.yourcompany.com:8000/imidio_api/"
      />
</Presence>

<!--                                                           -->
<!--         Mirrored Folders Configuration Settings           -->
<!--                                                           -->
<!--  Mirrored folders are local/shared directories that      -->
<!--  are exposed within ICEcore.  The directories must be     -->
<!--  configured here first before they are available to      -->
<!--  the folder configuration interface within ICEcore (see  -->
<!--  Modify a Folder).                                        -->
<!--                                                           -->
<!--  For security reasons the set of people who can map      -->
<!--  ICEcore folders to these shared directories is limited  -->
<!--  Specify the specific ICEcore users or groups that are   -->
<!--  allowed to map each folder (separated by semi-colons).  -->
<!--                                                           -->
<!--  Each mirrored folder configuration must have a unique   -->
<!--  id (use a-z,0-9), and a title to be used in the user    -->
<!--  interface.  Set enabled to true to make the mirrored    -->
<!--  folder configuration active.   The examples below show  -->
<!--  how to set up both file system and Sharepoint mirrors.  -->
<!--                                                           -->
<!--  By default the mirrored folder is set to readonly.      -->
<!--  This means that ICEcore users can access the files, but -->
<!--  cannot modify them. Set the readonly attribute to false -->
<!--  to allow read/write access to the folder (based on both -->
<!--  this server's access to the directory and the ICEcore   -->
<!--  user's access).                                         -->
<!--                                                           -->
<!--  There are three mirrored folder types: file (for        -->
<!--  normal file systems), webdav (for WebdDAV servers) that -->
<!--  support basic authentication, and sharepoint (for       -->
<!--  Microsoft Sharepoint WebDAV access).                    -->
```

```xml
<!--                                                          -->

    <MirroredFolders>

            <MirroredFolder enabled="false" type="file"
                            id="fs1" title="Shared Files 1"
                            rootPath="k:/somedir" readonly="true">
                <AllowedUsers idList="admin;u1;u2;u3" />
                <AllowedGroups idList="g1;g2;g3" />
            </MirroredFolder>

            <MirroredFolder enabled="false" type="file"
                            id="fs2" title="Shared Files 2"
                            rootPath="/sharedFiles/someDirectory"
readonly="true">
                <AllowedUsers idList="admin;u1;u2;u3" />
                <AllowedGroups idList="g1;g2;g3" />
            </MirroredFolder>

            <MirroredFolder enabled="false" type="sharepoint"
                            id="sp1" title="Sharepoint 1"
                            rootPath="/Shared Documents/cool-dir"
readonly="true">
                <WebDAVContext hostUrl="http://hostname"
user="accessId" password="pass" />
                <AllowedUsers idList="admin;u1;u2;u3" />
                <AllowedGroups idList="g1;g2;g3" />
            </MirroredFolder>

    </MirroredFolders>

    <!--                                                     -->
    <!--               iChain Single Sign-On Support         -->
    <!--                                                     -->
    <!--                                                     -->
    <!--  To use iChain SSO, set the enable attribute to true. -->
    <!--  Set the Logoff URL to the address used to trigger an -->
    <!--  iChain logoff.  Also set the ip address of the iChain -->
    <!--  proxy server.  Only transactions from that address and -->
    <!--  localhost are allowed.                             -->
    <!--                                                     -->

    <SSO>
        <iChain type="1" enable="false">
            <Logoff url="http://something" />
            <Proxy  ipaddr="ipaddr" />
        </iChain>
```

```
</SSO>

<!--                                                   -->
<!--              Custom Configuration Settings         -->
<!--                                                   -->
<!--  Custom properties set here are placed in the      -->
<!--  ssf-ext.properties file.                          -->
<!--                                                   -->

<CustomProperties>
    <Resource
    />
</CustomProperties>

</ICEcoreConfig>
```

Glossary

access control
The tool that determines who has the right to perform which tasks in which places. See also role-based access control.

accessibility mode
An optimized user interface that facilitates use by assistive devices, such as auditory readers.

accessory
A section located at the top of a workspace or folder page that provides a summary view, most likely of the information contained within the item. For example, an accessory can show all of the entries within a folder that are authored by someone designated as a subject-matter expert.

advanced search
Extra search tools that allow you to specify more specific criteria (such as the author of an item, or restricting the search to a portion of the workspace tree).

alias
See e-mail alias.

attachment
A file attached to an entry.

author
The person who created an entry.

blog
A folder that contains a chronological listing of journal entries.

blog archive
A feature of blog folders that allows you to see entries authored in a specific month.

buddy list
A list of people whose presence you want to check and whom you contact frequently.

calendar
A folder containing entries for scheduled appointments.

clipboard
A tool that gathers people's names. Later, when using a tool that requires names, you can take them from your clipboard.

comment
A reply to an entry.

community tag

A keyword tag applied to an item by the owner of a workspace or folder. Other users of the workspace or folder can perform searches based on community tags.

configuration

A set of tools that alter the way item content is presented. There are many types of configuration, ranging from setting allowable views for an item, selecting a color scheme, creating custom entries, and enabling workflow processes.

default view

The configuration of the information you see when you first view a workspace or folder. Some items might be configured to allow alternate views, which you can select.

definition

A set of elements for both the form and view of a workspace, folder, or entry.

designer

A tool used to create definitions or workflow processes.

discussion

A folder whose entries are discussion topics and comments about those topics.

e-mail alias

An alternative e-mail address for an e-mail account. To enable e-mail posting into a folder, you must provide an e-mail alias for the one account used to post into all folders in your ICEcore installation. Consult with your ICEcore site administrator for further assistance; site administrators, consult with the IT person responsible for creating e-mail accounts to create new aliases.

e-mail notification

An e-mail message that ICEcore sends indicating new or changed entries in a folder or subfolder.

entry

An item in a folder.

favorites panel

A tool used to save links to workspaces and folders most important to you, providing a method of accessing these places quickly.

file folder

A folder whose entries are configured to highlight an attached file and to facilitate file management.

filter

A setting that limits a folder listing to only the entries that match the filter's search criteria. For example, you can create a filter that shows only the contents of a folder authored by you or that were created past a certain date.

folder

A container for entries and other folders. Each folder has a type, such as blog, wiki, or calendar, that determines its appearance and features.

form

An HTML form used to create a workspace, folder, or entry.

global workspace

A workspace that, by default, allows everyone in your organization to participate.

guestbook

A folder or accessory whose entries indicate who has visited the place.

help mode

A dimmed page and information icons ("i"). When you click on an information icon, ICEcore presents a panel of information about that section of the page.

inherit

A process by which a workspace or folder automatically uses configuration settings from its parent.

instant message (IM)

A quick communication between teammates, using the Conferencing messaging software.

Liferay

The portal software within which ICEcore runs by default.

meeting

An online communication by teammates, using the Conferencing messaging software. Conferencing provides tools that assist with online meetings, such as people designated as running the meeting, a way for participants to "raise their hands," and a whiteboard.

milestone

A folder that, by default, summarizes the status of tasks in a task folder as they relate to meeting project milestones.

navigator

A set of tools that you can use to go anyplace within ICEcore you want to go. The tools include "My workspace," "Favorites," viewing your teams, search, Help, and a workspace tree.

owner

The person who created the workspace, folder, or entry.

parallel workflow process

A set of state transitions that happen at the same time as other state transitions. A state in the main thread initiates the parallel process, and a state later in the main thread can wait for the completion of the parallel thread.

parent

A workspace or folder that contains another workspace or folder. The item contained within the parent is sometimes called its child.

participant

An access role that, generally, by default, allows people to author entries in a folder.

permalink

A Web address (URL) for a ICEcore workspace, folder, or entry that you can copy, paste, and send to teammates so that they can access a page directly by specifying the address in their web browsers.

personal tag

A keyword tag that you apply to an item, and that only you can see and use.

personal workspace

A workspace that serves as a person's home page in ICEcore, including contact information, pictures, a personal blog, and more.

photo gallery

A folder whose entries are pictures.

portal page

A Web page that can run various applications in sections of its page. For example, Google and Yahoo use portal pages. Sections within a portal page can display the local time, the local weather, your favorite stock quotes, and more.

portlet

A section on a portal page. ICEcore runs within portlets.

presence

A person's online status (online, away from the computer, offline, status unknown), represented in ICEcore by people icons of certain colors.

project-management workspace

A workspace configured to facilitate the tracking of tasks and completion of complex project work.

role-based access control

A mechanism that controls access by assigning people and groups to roles, and the roles determine the rights assigned to those people. See the online help for a list of ICEcore default role definitions.

site administrator

The person or people who have the right to perform any task in the ICEcore installation.

state

See workflow state.

subscription

A way to track new or changed items in ICEcore.

tag

A keyword that anyone can apply to a workspace, folder, or entry to make it easier to find. See also personal tag and community tag.

task

A folder that, by default, contains entries that track progress in regard to completing an assignment.

team

An access role that, by default, allows people to participate in a workspace or folder, to do some minor administrative tasks, and to communicate easily with each other.

team workspace

A workspace that restricts participation to teammates.

template

A set of default configuration settings used to create a new workspace or folder. A template includes at least one definition, access control, and a possible hierarchy of defined items.

type to find

A ICEcore tool that uses the characters you have typed so far to present a list of possible choices. This tool is active for text boxes whose values are within a defined set: for example, people, places (workspaces and folders), and tags.

view

A presentation of an item's content. For example, you can view a discussion folder in either a list or table format. By default, most folders use one view (calendar folders use a calendar view, blog folders use a blog view, and so on).

visitor

An access role that, by default, allows people to read entries and make comments on them, but not create new entries.

WebDAV

The Web Distributed Authoring and Versioning protocol. If your system provides a tool that uses this protocol, it allows you to manage ICEcore file and folder entries by using the WebDAV window.

wiki

A folder whose entries are authored by all participants.

workflow

An online representation of a business process (for example, document review, paid time-off requests, document sign-off, and so on). An entry can have an associated workflow process, which places the entry into various workflow states.

workflow state

A status label for an entry in a workflow process. A state determines who has the right to work with an entry (including who can see it), who needs to be notified, who needs to perform the next task, and which subsequent states are possible.

workspace

A container for folders and other workspaces.

workspace tree

A tool that allows you to navigate the hierarchy of workspaces, sub-workspaces, folders, and subfolders within ICEcore.