Axway Application Studio

Version 1.2

19 November 2014

Contents

Preface

This guide describes the requirements and tasks for installing and starting Axway Application Studio. It also provides information about upgrading and installing Application Studio in a cluster of multiple computers.

About Application Studio

Application Studio combines development and operations in one product. It is a rapid application-development environment for building web applications that implement business processes. It is also an application operations environment for business users.

Application Studio is rapid in the sense it uses model-driven, visual development approaches rather than code for much, if not all, of the application development process. It does involve software-development engineering, but lets developers work at a higher level of abstraction than when coding. Business processes are designed using sequences of activities that involve participants, represented as blocks laid out in swim lanes customary to **BPM**[1]. Forms are created by dragging visual elements from a palette onto the form canvas.

Application Studio is built on Joget Workflow Enterprise Edition, a platform for building enterprise web applications.

Who should use this guide

This guide is for people who oversee planning and installation of Application Studio. This guide presumes you have knowledge of:

- Your company's business processes and practices
- Your company's hardware, software and IT policies
- The Internet, including use of a browser

Axway 5 Suite reference solutions

Application Studio, a **Unified Flow Management** product, is integral to all Axway 5 Suite reference solutions. These meld selected Axway products into seamlessly integrated systems:

[1]Business process management (BPM) is the discipline of managing processes for improving business performance outcomes and operational agility. Processes span organizational boundaries, linking people, information flows, systems and other assets to create and deliver value to customers and constituents.

- Manged File Transfer to securely transfer data in one-to-one, one-to-many and many-to-many scenarios.

- B2B Integration to exchange, transform and process standardized business documents withing an enterprise's B2B community

- Data Flow Integration to provide services for standardizing the exchange of business data with internal and external partners.

- Financial Integration to support data transfers in finance channels such as SWIFT and EBICS and transformations of data in financial protocols.

Your organization might use this product in context of a reference solution. Find details about the product's role in documentation on the Axway Sphere support website at support.axway.com. After logging on click **Axway 5 Suite** on the top toolbar.

Other documentation

Refer to the Application Studio Help Center and the Joget knowledge base for information about installing, getting started and using Application Studio. The URLs are:

Application Studio Help Center

http://app-studio-help-center.squarespace.com/

You also can click Help Center in the user interface to access it.

Joget knowledge base

http://dev.joget.org/community/display/KB/Joget+Workflow+Community+Knowledge+Base

Application Studio uses Joget Workflow 3, and this URL opens version 3 of the knowledge base.

Axway online

Go to Axway Sphere at support.axway.com to contact a representative, learn about training programs, or download software, documentation and knowledge-base articles. Check Sphere before installing and at intervals thereafter for the latest product documentation. Sphere is refreshed periodically with updated user documentation.

Sphere is for customers with active Axway support contracts. You need a user name and password to log on.

Post comments and questions to the Axway user forum at forums.axway.com/index.php

System requirements   1

The following are the system requirements for Application Studio.

If you use multiple Axway products, you can compare this product's system requirements with the supported operating systems, databases, browsers and more for other products. See the *Axway 5 Suite Supported Platforms* guide on Sphere at support.axway.com.

Operating systems

The following are the supported 64-bit operating systems.

- SUSE Linux Enterprise Server 11
- Oracle Linux 5.10
- Microsoft Windows Server 2008 R2

Windows administrator user

You must install Application Studio as an administrator user on Windows. You also must start the product with the same administrator user that installed it.

When upgrading from an older to a newer version of Application Studio, you must install the upgrade with the same user that installed the earlier version.

Linux root user

You must install and run Application Studio as the root user on Linux.

When upgrading from an older to a newer version of Application Studio, you must install the upgrade as root.

Databases

The following databases are supported.

- MySQL 5.6.12
- Axway Database 4.6

- Oracle 11g Release 2 (11.2)
- Oracle **RAC**[1] 11g Release 2 (11.2)

RAC note

Application Studio has been tested with Oracle RAC 11g R2 using the Single Client Access Name (SCAN) feature. Oracle JDBC driver version ojdbc6-11.2.0.4 is included in Application Studio to support RAC. Other RAC configurations were not tested.

Browsers

The following browsers are supported for navigating the user interface.

- Google Chrome 24 and later
- Mozilla Firefox 18 and later
- Microsoft Internet Explorer 9 and 10

---

[1]Oracle Real Application Clusters (RAC) provides software for clustering and high availability in Oracle database environments.

Prerequisites

2

The following topics include tasks to perform or information to know before installing Application Studio.

Database prerequisites

The following are the prerequisites for using a supported external database. See *Databases* on page 7 for the list of supported databases.

Using a database requires having a **DBA**[1] or other technically qualified person with knowledge of the database software. This person must perform setup and maintenance tasks, such as database backups.

MySQL and Axway Database

The following are the prerequisites for using a MySQL database or Axway Database. Axway Database also is a MySQL database.

- The database must be installed and running before installing Application Studio.

- Set the `max_allowed_packet` property to 16 MB.

- You must create a database in the database system before installing Application Studio. The database can have any name. For example, **appstudio**.

- The database user must have full privileges for the database you create.

- If Application Studio and the database are on different computers, Application Studio must have rights to access the remote database. See *Remote access example* on page 10.

Create database example

The following is an example of creating an empty database in a MySQL session. This example also is valid for Axway Database.

```
Welcome to the MySQL monitor. Commands end with ; or \g.

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights
reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective
```

[1]database administrator

```
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> create database appstudio;

mysql> quit;
```

Remote access example

The following is an example of granting Application Studio access to a remote database.

```
mysql> grant ALL on *.* to 'root'@'aas-ip-address' identified by
'password';
```

Where:

- `aas-ip-address` is the IP address of the AAS server
- `password` is the MySQL root user password

Oracle

The following are the prerequisites for using an Oracle database.

- A schema must exist for the APPSTUDIO Oracle database.
- An APPSTUDIO user must exist for the schema with the following permissions:

  connect

  resource

  create session

  create table

The following is an example of creating an empty schema in a SQL*Plus session.

```
SQL*Plus: Release 11.2.0.1.0 Production on Fri Jan 31 11:47:28 2014

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Enter user-name: SYSTEM

Enter password: <as specified in Oracle installation>

Connected to:

Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit
Production

With the Partitioning, OLAP, Data Mining and Real Application Testing
options

SQL> create user APPSTUDIO identified by password;
```

```
User created.

SQL> grant connect, resource to APPSTUDIO;

Grant succeeded.

SQL> grant create session to APPSTUDIO;

Grant succeeded.

SQL> grant create table to APPSTUDIO;

Grant succeeded.

SQL> quit
```

Database case-sensitivity rules

You must follow guidelines to make sure the applications you develop in Application Studio are database agnostic. This ensures portability of applications among instances of Application Studio regardless of the database type. For example, an application developed in an instance of Application Studio with an Oracle database can be exported and imported to an instance of Application Studio with a MySQL database, and vice versa.

Regardless of the database type, apply the following rules when creating a database, tables and column names and in custom queries:

- For database names and table names, mixed case is acceptable, but the names must be case sensitive.
- For database column names, use all upper-case.

The following are the case requirements for database objects by database type and operating system:

For database names and table names

- MySQL and Axway Database on Windows - case insensitive
- MySQL and Axway Database on Linux - case sensitive
- Oracle on Windows and Linux - the Application Studio hibernate layer allows case insensitive

For database column names

- MySQL and Axway Database on Windows - case insensitive
- MySQL and Axway Database on Linux - case insensitive
- Oracle on Windows and Linux - upper-case

License

You do not need a license to install Application Studio; the installer does not prompt for one. After installing and logging on to the user interface, Application Studio is limited to having only three users, including the default admin user. You are allowed to create more than three users, but only the first three alphabetically are allowed to be involved in workflows.

Shortly after installing on your intended production computer, you should request a license key from Axway to fully enable Application Studio. See *Start and log on the first time* on page 16 for more information.

Plan and install

3

The following describes the information you are prompted to enter when installing Application Studio. You might want to gather this information before starting the installer.

Note   If you already are running Application Studio and want to install an upgrade, skip this topic and see *Upgrade* on page 21.

Installation directory

You can use or replace the default value for the installation directory.

Database

The installer prompts for values to connect to your external database. It also verifies the database connection before allowing you to proceed. Complete the tasks in *Database prerequisites* on page 9 before installing.

MySQL, Axway Database

When the database is MySQL or Axway Database:

- Host name is the fully qualified domain name or IP address of the computer running the database Application Studio will use.
- Port is the listening port of the database server.
- User name is the user connecting to the database.
- Password is the password of the database user.
- Database name is name of the database you have created for Application Studio.

Oracle

When the database is Oracle:

- Host name is the fully qualified domain name or IP address of the computer running the database Application Studio will use.
- Port is the listening port of the database server.
- User name is the name of the user and schema.

- Password is the password of the schema user.

- Service name is the alias used when connecting to the database.

Shared secret

The installer prompts for a shared secret. This is used for all users and the database and nodes in a cluster environment. It is for generating a common **salt**[1] . It is used to encrypt user passwords and the database password.

You should record and save your shared secret in a safe place, such as in a secure password manager. The secret is shared between the database and one or more instances of Application Studio.

You will need the shared secret to reattach the database to a fresh installation of Application Studio should the original installation become corrupted.

Ports

Default ports for connections to the user interface are 80 for HTTP and 443 for HTTPS. You can use or replace the default values. The installer verifies whether the ports are already in use.

SSL certificate

The installation process generates a self-signed **SSL**[2] web server certificate to secure the HTTPS connections from web browser users to Application Studio and its applications.

You can specify

- A certificate key size of 1024, 2048 (default), 3072 or 4096 bits.

- A validity period of 1 to 10 years for the certificate.

You also are prompted to enter certificate details:

- Two-letter country code

- State or province

- Locality name (for example, your city)

- Organization name (for example, your company)

---

[1]In cryptography, a salt is random data that are used as an additional input to a one-way function that hashes a password or passphrase. The primary function of salts is to defend against dictionary attacks and pre-computed rainbow table attacks. A new salt is randomly generated for each password. In a typical setting, the salt and the password are concatenated and processed with a cryptographic hash function, and the resulting output (but not the original password) is stored with the salt in a database. Hashing allows for later authentication while defending against compromise of the plaintext password in the event that the database is somehow compromised.
[2]Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols for providing communication security over the Internet.

- Organizational unit name (for example, your department or division)

- Common name (for example, name of computer running Application Studio)

Ready to install

Once your planning is completed, you are ready to install Application Studio.

Windows

In Windows, double-click the EXE file you downloaded from Sphere to start the installer's graphical user interface. Follow the prompts to install Application Studio.

You must install Application Studio as an administrator user on Windows. You also must start the product with the same administrator user that installed it.

Linux

In Linux, extract the compressed file you downloaded from Sphere. Run `install.sh` to launch the installer in text mode (GUI mode is Windows only). Follow the prompts to install Application Studio.

You must install and run Application Studio as the root user on Linux.

Start and log on the first time 4

Use this procedure to start Application Studio and log on the first time after installing.

After logging on, click **Help Center** at the top of the page for information about getting started with Application Studio.

1. Start Application Studio. Run `start-appStudio.bat` (Windows) or `start-appStudio.sh` (Linux) in the installation directory.

   In Windows, you alternately can start the product by clicking **Start App Studio** in the installation directory or under **Axway Application Studio** on the **Start** menu.

   On Windows, you must start Application Studio with the same administrator account that was used to install it. On Linux, you must start the product as the root user. See *System requirements* on page 7 for more information.

2. Use a URL in the following format to open the log-on page in a browser:

   `https://<host>:<port>/appstudio`

   Where:

   - `host` is the fully qualified domain name or IP address of the computer running Application Studio.

   - `port` is the HTTPS port specified during installation. If the HTTPS port is 443, you do not need to specify it in the URL.

   In Windows, you alternately can open the log-on page by clicking **Web Console** under **Axway Application Studio** on the **Start** menu.

   If a message displays that a certificate is not trusted,click the appropriate button to proceed to the log-on page.

   > **Note** If you instead attempt to connect to the log-on page via the HTTP port specified during installation, you are redirected to HTTPS. If the HTTP port is 80, you do not need to specify it in the URL.

3. On the log-on page, type **admin** as the user name and **axway** as the password. After you submit, you are prompted to set a new password for the admin user. Rules are provided for the new password format.

4. Once logged on, scroll down to the bottom of the page. Notice the message that says **Licensed Users: 3**. You need a license to remove this restriction.

   To expand the number of licensed users, you must request from Axway a license that is bound to a specific computer. If you intend to run Application Studio on another computer, you may want to wait before requesting the license. Meantime, you can experiment with Application Studio on other computers with the three-user limit in place.

5. To request a license:

    a. Click **Axway Application Studio** at the bottom of the page to open the License page.

    b. Copy the system key on the License page and send it to Axway with a request for a license. Follow the instructions in your Axway welcome letter for requesting the license. This was the email message that informed you how to download the Application Studio software from Axway Sphere.

    c. Once you have received the license, open the License page and enter the admin user name and password and license and click **Submit**.

After a valid license is entered, the three-user limitation is removed. You might have to refresh the browser page after submitting the license to display the new license summary.

Install a cluster

5

The following topics describe installing Application Studio in an **active-active**[1] cluster environment on multiple computers.

Cluster architecture

A typical architecture of a cluster environment for Application Studio can consist of:

- Two or more application server nodes running Application Studio
- One shared database used by all nodes
- One shared file system (for example, **NAS**[2] running the **NFS**[3] protocol).
- A load balancer to distribute processing among the application server nodes

In this architecture, if one application server node goes down, users can access Application Studio through other nodes. However, this does not protect from failures in a shared database or shared file system. To further improve availability, you can use a database cluster and **RAID**[4] in the shared file system.

Cluster architecture
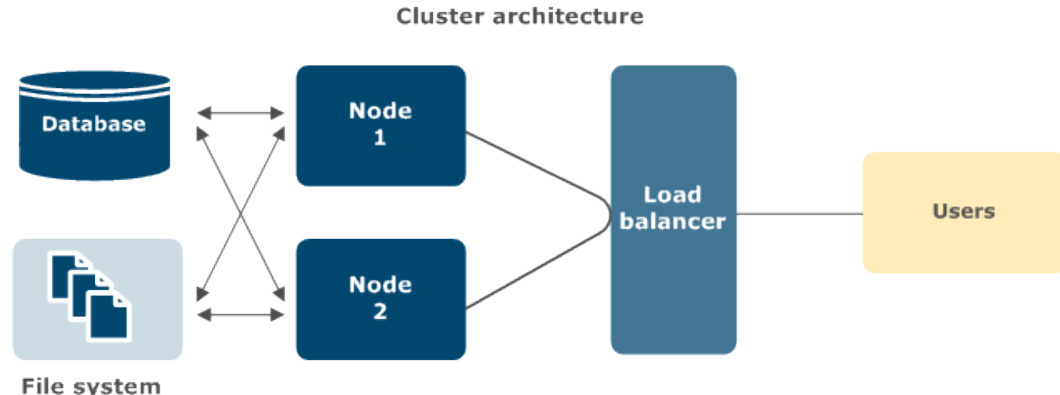
[1]Active-active is a type of high-availability cluster environment. In an active-active cluster, traffic intended for the failed node is forwarded to an existing node or load-balanced across the remaining nodes. This is usually possible only when the nodes have a homogeneous software configuration.
[2]Network-attached storage (NAS) is file-level computer data storage connected to a computer network providing data access to a heterogeneous group of clients. NAS operates as a file server. It is specialized for this task by its hardware, software or configuration of those elements.
[3]Network File System (NFS) is a distributed file system protocol originally developed by Sun Microsystems in 1984. It allows a user on a client computer to access files over a network in a manner similar to how local storage is accessed.
[4]Redundant array of independent disks (RAID) is a storage technology that combines multiple disk drive components into a logical unit for the purposes of data redundancy and performance improvement. Data are distributed across the drives in one of several ways, referred to as RAID levels, depending on the specific level of redundancy and performance required.

Why cluster

The following are reasons for operating Application Studio in a cluster environment:

- Scalability. The processing load can be distributed among two or more nodes.

- High availability. If a node goes down, a load balancer can direct user requests to nodes that are still running.

Sticky sessions

Application Studio supports active-active clustering without session synchronization. Nodes do not share sessions. You must use **sticky sessions**[1] to have the load balancer send you to the same node you logged on to originally. If that node fails, the load balancer sends you to the other node, where you have to log on again.

Clustering guidelines

There are various ways to set up clustering and load balancing. Refer to the third-party user documentation for the methods you want to implement. However, the following are points to keep in mind when setting up Application Studio in an active-active cluster.

- Make sure Application Studio servers (all nodes) have full access to the shared database.

- Use the same shared secret on all nodes.

- Configure Application Studio servers to use the shared file system. Depending on how the shared file system is set up, you might have to edit the start script in the installation directory to point to the correct location. The shared file system normally shares the `wflow` directory found in each node's installation directory.

- Make sure all Application Studio servers have read-write access to the shared file system.

- Edit the `server.xml` file at `<install directory>\apache-tomcat-<version>\conf` to point to the load balancer or proxy or firewall so traffic is routed correctly.

- Do not use URLs in the format `https://<node IP_address>/appstudio/<other content>`. Such URLs cannot be relied upon to redirect to the load balancer.

---

[1]A sticky session is a feature of many commercial load balancing solutions for web-farms to route requests for a session to the same physical machine that serviced the first request for that session. This is mainly used to ensure an in-proc session is not lost as a result of requests for a session being routed to different servers. Since requests for a user are always routed to the same machine that first served the request for that session, sticky sessions can cause uneven load distribution across servers.

Licenses for cluster nodes

You must request and apply a license for each Application Studio node before joining the nodes together in a cluster. Moreover, before replacing or updating the license for a node, you must remove the node from the cluster and rejoin the node only after the license has been replaced or updated. See the steps for requesting and applying a license for a single Application Studio in *Start and log on the first time* on page 16.

If nodes already are clustered, which means the `server.xml` files have been updated to the cluster settings, you must restore the `server.xml` files to the default, non-clustered state and then license the nodes. A good approach is to save the default `server.xml` as `server.xml.original` on each node. However, if the default `server.xml` is not available, save a copy of the current `server.xml` as `server.xml.cluster` and remove the cluster settings from `server.xml`. Once restored to the default state, you can use it in the licensing procedure and save the file as `server.xml.original`. Then you can resume using `server.xml.cluster`.

Stop Application Studio before working on a `server.xml` file. The file is at `<install directory>\data`.

Other resources

Application Studio is built on Joget Workflow Enterprise Edition, a platform for building enterprise web applications.

Joget does not have a special mechanism for clustering. It uses clustering support provided by the Apache Tomcat application server. This is described at:

http://tomcat.apache.org/tomcat-7.0-doc/cluster-howto.html

Upgrade

6

Use this procedure to upgrade Application Studio from version 1.1 to version 1.2. If you have never installed Application Studio or want to install a new instance of the product, skip this topic and see *Plan and install* on page 13.

Although Application Studio supports multiple types of external databases, the upgrading process only enables installing a new product version that uses the same database as the previous version. You cannot change databases or database types. For example, you cannot change from an existing MySQL database to a new MySQL database and preserve data from the old database. You also cannot change from MySQL to Oracle. Changing the database requires performing a complete installation with a new database.

Note   If you have a version earlier than 1.1, you must upgrade to 1.1 before upgrading to 1.2. See upgrading in the 1.1 installation guide. You also can go to the Application Studio Help Center at http://app-studio-help-center.squarespace.com/installation/.

Prerequisites

- Verify the existing installation of Application Studio. There should be an installation directory (for example, on Windows the default path is `C:\Axway\ApplicationStudio`). There also should a security data directory.
    - On Windows, if installed as Administrator, the security data directory is at `C:\Users\Administrator\AppData\Local\Axway\AppStudio`.
    - On Linux, if installed as root, the security data directory is at `/root/.appStudio`.
- Make sure you have the upgrade package. If Windows, this is a executable file. If Linux, this is a tar.gz file you must unpack.
- Review the release notes for the upgrade to determine whether any issues might affect installation or operations.
- On Windows, you must install the upgrade with the same administrator user that installed the earlier version.
- On Linux, you must install the upgrade as root.

Upgrade steps

1. Stop Application Studio. See *Stop Application Studio* on page 23.

2. Back up the database and the existing installation of Application Studio, including the security data directory. In the event of problems, you can revert to the previous version of Application Studio only if you perform backups before upgrading. See *Save and reuse sensitive files* on page 24.

3. Launch the installer of the new version.

   - In Windows, double-click the EXE file to open the installer's graphical user interface.

   - In Linux, run `upgrade.sh` to start the installer in text mode (GUI mode is Windows only).

4. In Windows, after you click past the welcome and license agreement pages, the installation location page is displayed. In the Destination Folder field is the detected path of the Application Studio already installed. The installer verifies whether it is a valid installation, aborting if the validations fails. If the installer does not find a current installation or you select a new path, it lets you perform a full installation of the new version and not an upgrade.

   In Linux, you must select the path for the currently installed version to proceed with the upgrade.

   Click **Next**.

5. Read the warning about stopping and backing up Application Studio and its database. Click **OK** to proceed with the upgrade or **Cancel** if you need to stop Application Studio or perform backups.

6. If you clicked **OK**, click **Upgrade** to install the new version.

7. Click **Finish** to close the installer.

8. Start Application Studio and log on to the user interface. You can confirm you are running the new version by scrolling to the bottom of the page and reading the Application Studio version number.

Upgrade nodes in cluster

If you are running Application Studio in a cluster of multiple nodes, upgrade each node using the procedure in *Upgrade* on page 21. Then edit the `setenv` file in the installation directory to set the AppStudioWflow property to the location of the shared workflow folder. For example, if wflow is mapped to drive z, use **set AppStudioWflow=z:\wflow**.

Upon upgrading, the values in the old `server.xml` file migrate to the new `server.xml` file in `<install directory>\data`. You do not have to edit the file after upgrading. The file contains cluster settings.

Stop Application Studio

7

To stop Application Studio, run `stop-appStudio.bat` (Windows) or `stop-appStudio.sh` (Linux) in the installation directory.

In Windows, you alternately can stop the product by clicking **Stop App Studio** in the installation directory or under **Axway Application Studio** on the **Start** menu.

Uninstall

8

Use the procedure for your operating system to uninstall Application Studio.

Save and reuse sensitive files

Before uninstalling, you might want to save sensitive files or directories to reuse in a future installation of Application Studio. You copy the saved files to the new instance, overwriting existing files if necessary.

To re-use application configuration

Save any changes made to files in `<install directory>/apache-tomcat-<version>/conf` (Linux and Windows).

To re-use apps and plugins

Save the directory `<install directory>/wflow` (Linux and Windows).

To re-use a database after uninstalling

Save the sensitive files in `C:\Users\<user name>\AppData\Local\Axway\AppStudio` (Windows).

Save the sensitive files in `~/.appStudio` (Linux).

Uninstall on Windows

Before uninstalling on Windows, stop Application Studio. See *Stop Application Studio* on page 23 for details.

Run `Uninstall.exe` in the installation directory to start the uninstall wizard. Follow the prompts to uninstall.

The Windows uninstall process does not delete sensitive data files. Review the message at the end of the uninstalling process and determine whether you want to keep or discard data files that were not deleted. Manually delete the files to remove Application Studio completely.

Uninstall on Linux

Before uninstalling on Linux, stop Application Studio. See *Stop Application Studio* on page 23 for details.

Next, review the sensitive data files in `~/.appStudio` and determine whether to save or discard them.

After reviewing the sensitive data files, delete the Application Studio installation directory. This uninstalls Application Studio.

Glossary

This is a glossary of terms for Axway Application Studio.

active-active

Active-active is a type of high-availability cluster environment. In an active-active cluster, traffic intended for the failed node is forwarded to an existing node or load-balanced across the remaining nodes. This is usually possible only when the nodes have a homogeneous software configuration.

BPM

Business process management (BPM) is the discipline of managing processes for improving business performance outcomes and operational agility. Processes span organizational boundaries, linking people, information flows, systems and other assets to create and deliver value to customers and constituents.

DBA

database administrator

MAC address

A media access control address (MAC address) is a unique identifier assigned to network interfaces for communications on the physical network segment. MAC addresses are used as a network address for most IEEE 802 network technologies, including Ethernet.

NAS

Network-attached storage (NAS) is file-level computer data storage connected to a computer network providing data access to a heterogeneous group of clients. NAS operates as a file server. It is specialized for this task by its hardware, software or configuration of those elements.

NFS

Network File System (NFS) is a distributed file system protocol originally developed by Sun Microsystems in 1984. It allows a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

RAC

Oracle Real Application Clusters (RAC) provides software for clustering and high availability in Oracle database environments.

RAID

Redundant array of independent disks (RAID) is a storage technology that combines multiple disk drive components into a logical unit for the purposes of data redundancy and performance

improvement. Data are distributed across the drives in one of several ways, referred to as RAID levels, depending on the specific level of redundancy and performance required.

salt

In cryptography, a salt is random data that are used as an additional input to a one-way function that hashes a password or passphrase. The primary function of salts is to defend against dictionary attacks and pre-computed rainbow table attacks. A new salt is randomly generated for each password. In a typical setting, the salt and the password are concatenated and processed with a cryptographic hash function, and the resulting output (but not the original password) is stored with the salt in a database. Hashing allows for later authentication while defending against compromise of the plaintext password in the event that the database is somehow compromised.

SSL

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols for providing communication security over the Internet.

sticky sessions

A sticky session is a feature of many commercial load balancing solutions for web-farms to route requests for a session to the same physical machine that serviced the first request for that session. This is mainly used to ensure an in-proc session is not lost as a result of requests for a session being routed to different servers. Since requests for a user are always routed to the same machine that first served the request for that session, sticky sessions can cause uneven load distribution across servers.