

# **Scarab**

## **Administration Guide**

**\$Rev: 9720 \$**

**The Scarab Development Team**

---

**Scarab: Administration Guide: \$Rev: 9720 \$**

The Scarab Development Team  
Copyright © 2003 CollabNet, Inc.

---

---

---

# Table of Contents

Preface .....	vi
Audience .....	vi
How This Book is Organised .....	vi
I. Scarab Installation .....	7
1. General .....	9
How to install Scarab .....	9
The steps in a normal Scarab installation .....	9
2. Preparing the installation .....	10
Choosing an operating system .....	10
Choosing a database manager .....	10
DBMS that have been tested with Scarab (and are supported) .....	10
DBMS that have been tested and are not fully supported by Scarab .....	11
DBMS that are not supported by Scarab .....	11
Choosing a servlet container .....	11
Gathering the other elements required for the Scarab installation .....	11
Download and install a Java SDK .....	12
Download and install Ant .....	12
Download Scarab .....	12
3. Installing the database software and preparing for Scarab installation .....	13
MySQL .....	13
Installing MySQL .....	13
Create a database user for Scarab .....	13
Setting up the MySQL parameters in Scarab .....	13
Option: downloading an optimized driver .....	14
PostgreSQL .....	14
Installing PostgreSQL .....	14
Setting up the PostgreSQL in Scarab .....	14
Oracle 9i .....	14
Installing Oracle 9i .....	14
Setting up the Oracle 9i parameters in Scarab .....	14
4. Building Scarab .....	15
Finishing Scarab configuration .....	15
Defining the server name and protocol .....	15
Choosing the name and email address of the Scarab administrator .....	15
Defining the mail server .....	16
Localizing your Scarab installation .....	16
Preparing the deployment on another servlet engine .....	16
Building Scarab .....	16
5. Building the Scarab database .....	18
MySQL .....	18
On Unix/Linux .....	18
On Windows .....	18
6. Running Scarab .....	19
Running Scarab with its own Tomcat server .....	19
II. Scarab system administration after installation .....	20
7. Backups and Restores .....	22
Backing up Scarab information .....	22
Backing up the database .....	22
Backing up other information .....	22
Restoring Scarab information from a backup .....	22
Restoring the database .....	23
Restoring other information .....	23
8. Moving an existing Scarab installation to another server .....	24

Modifying the database when the Scarab server changes .....	24
Modifying the SCARAB_GLOBAL_PARAMETER table .....	24
Modifying the SCARAB_MODULE table .....	24
Modifying the SCARAB_ISSUE table .....	24

---

# Preface

## Audience

This book is written for Scarab System administrators. It does not tackle usage questions, which are covered in another ebook, the Scarab User's Guide.

## How This Book is Organised

This book is divided in two parts. The first part covers Scarab installation; the second part answers a few questions about Scarab administration after installation (such as backups and restores).

---

# **Part I. Scarab Installation**

---

---

## Table of Contents

1. General .....	9
How to install Scarab .....	9
The steps in a normal Scarab installation .....	9
2. Preparing the installation .....	10
Choosing an operating system .....	10
Choosing a database manager .....	10
DBMS that have been tested with Scarab (and are supported) .....	10
DBMS that have been tested and are not fully supported by Scarab .....	11
DBMS that are not supported by Scarab .....	11
Choosing a servlet container .....	11
Gathering the other elements required for the Scarab installation .....	11
Download and install a Java SDK .....	12
Download and install Ant .....	12
Download Scarab .....	12
3. Installing the database software and preparing for Scarab installation .....	13
MySQL .....	13
Installing MySQL .....	13
Create a database user for Scarab .....	13
Setting up the MySQL parameters in Scarab .....	13
Option: downloading an optimized driver .....	14
PostgreSQL .....	14
Installing PostgreSQL .....	14
Setting up the PostgreSQL in Scarab .....	14
Oracle 9i .....	14
Installing Oracle 9i .....	14
Setting up the Oracle 9i parameters in Scarab .....	14
4. Building Scarab .....	15
Finishing Scarab configuration .....	15
Defining the server name and protocol .....	15
Choosing the name and email address of the Scarab administrator .....	15
Defining the mail server .....	16
Localizing your Scarab installation .....	16
Preparing the deployment on another servlet engine .....	16
Building Scarab .....	16
5. Building the Scarab database .....	18
MySQL .....	18
On Unix/Linux .....	18
On Windows .....	18
6. Running Scarab .....	19
Running Scarab with its own Tomcat server .....	19

---

# Chapter 1. General

## How to install Scarab

In the current state of development -- and maybe in the long term as well -- Scarab does not install fully assembled, as is usually the case with an installation program.

Scarab is built dynamically for every particular installation, according to local parameters among which the database used to store the issues and other information that Scarab needs to work.

This could make you anxious at first sight, but the process is simple and proven: you can achieve this task easily like most other people did until now.

## The steps in a normal Scarab installation

A normal installation is performed in four steps :

- Preparing the installation
- Installing and configuring the database management software
- Building Scarab
- Building the Scarab database

---

# Chapter 2. Preparing the installation

## Choosing an operating system

Scarab is theoretically portable on any system that supports Java.

Practically, Scarab has been used and intensively tested on three kinds of platforms:

- Linux: the platform on which the most experience has been gathered (and thus the system on which you will easily get support from the user community if ever you face a problem).
- Windows
- MacOS X: in fact, Scarab has been developed in part on this platform, and the first Scarab demonstrator, the "runbox", was running on this platform.

## Choosing a database manager

Scarab needs a database management system to store the issues and other structured information (parameters, queries, reports, modules,...). Thus, you have to choose an RDBMS (for the database installation, see the next chapter).

Theoretically, you could use Scarab on any RDBMS that has a JDBC driver.

Practically, you are faced with some SQL compatibility issues, and moreover, you need to have correct support for this DBMS in Torque, the persistence and object-relational mapping component associated with the Turbine framework, on which Scarab is built.

## DBMS that have been tested with Scarab (and are supported)

### MySQL

MySQL is the DBMS on which Scarab has been developed and fully tested. If you don't have any particular reason to choose a database or another to run Scarab, MySQL should be your first choice.

Which version to choose? Scarab will work with any 3.x or 4.x version of MySQL. It could happen in the future that the support of version 3 could be dropped to use the transactional functionalities that are available only in version 4. So, if you don't have a particular reason to choose a particular version, you may choose long term peace of mind and use a version 4.x. from the start.

MySQL is open-source software, and it is available for free on Linux, Windows and a few other platforms (commercial licences and support are available for a fee).

### PostgreSQL

Scarab has been tested successfully on PostgreSQL, version 7.2.1 and the following.

PostgreSQL is open-source software, and it is available for free on Linux and Windows.

### Oracle 9i

The compatibility of Scarab with Oracle 9i has also been tested

Because of a problem with the manner Torque manages BLOBs, using Scarab with Oracle 9i is a bit more difficult than with other DBMS (you will have to patch the code generated by Torque).

## **DBMS that have been tested and are not fully supported by Scarab**

### **Microsoft SQL Server**

The Scarab configuration on Microsoft SQL Server has been carefully defined and tested.

A few problems remain, related to the mapping of certain data types by Torque and to the use of the SQL CONCAT function. These problems have not been solved at the moment.

People interested in the technical details on this question may refer to the appropriate documentation page on the Scarab web site.

### **Hypersonic SQL**

Scarab has been configured and tested carefully on HyperSonic.

However, the tests have shown that Scarab is not fully operational on this platform.

People interested in the technical details on this question may refer to the appropriate documentation page on the Scarab web site.

## **DBMS that are not supported by Scarab**

### **Oracle 8i**

Oracle 8i has been supported with some success at the beginning of the Scarab development.

This support has been discontinued in 2003 due to compatibility problems in the syntax of some SQL joins. Unless someone tackles this problem, Oracle 8i will not be supported again.

But Scarab is now compatible with Oracle 9i, as it has been discussed before.

## **Choosing a servlet container**

Scarab runs on any Servlet 2.3-compatible servlet engine; in fact, a patch could make Scarab compatible with 2.2 servlet engines, among which WebSphere Application Server 4.0.

Practically, most Scarab users use it on Tomcat; in fact, the Scarab installation process builds a "light" Tomcat server, the "sandbox", which you can use to test or run Scarab once it has been built.

You may want to deploy Scarab on an existing Tomcat installation; Resin has also been successfully tested as Scarab engine.

## **Gathering the other elements required for the Scarab installation**

To perform the Scarab installation, you will need three more things: a Java SDK, Ant and -- of course -- Scarab itself.

## Download and install a Java SDK

If you don't have a Java SDK handy (pre-installed on some systems such as MacOS X or certain Linux distributions), you need to download and install one. Be careful: you need a Java 2 SDK and not only a JRE (Java Runtime Environment) because you will need to compile Scarab.

The SDK version must be at least 1.3.1.

You may download a Java SDK from <http://java.sun.com>.

Make sure the `JAVA_HOME` environment variable correctly points to the directory in which your SDK is installed.

## Download and install Ant

To build Scarab, you will need the Ant build system.

You can download Ant from <http://ant.apache.org>.

The Ant version should be 1.5.1 or greater.

Make sure the `ANT_HOME` environment variable correctly points to the directory in which Ant is installed

## Download Scarab

You need of course to download Scarab itself. You will find the latest version (1.0-b18) at the following URL: <http://scarab.tigris.org/servlets/ProjectDocumentList>.

Uncompress the Scarab archive in the directory of your choice.

At the beginning of the installation process, you will focus on the `./build` sub-directory.

---

# Chapter 3. Installing the database software and preparing for Scarab installation

## MySQL

### Installing MySQL

Please refer to the MySQL installation instructions.

### Create a database user for Scarab

You will not be able (or you won't like) to run Scarab with the default MySQL security scheme (root user without password on the local machine).

So, you'll probably need to create a MySQL user and grant him/her the appropriate rights. Let's suppose you want to create a database user with the name 'scarab' on a local server et grant him the rights on a 'scarab' database; connect to the admin interface (the unix command is `mysql`) and execute the following statement :

```
grant all privileges on scarab.* to scarab@localhost;
```

If you want the 'scarab' user to authenticate with the password 'secret' when he/she/it connects to the database server, the syntax becomes

```
grant all privileges on scarab.* to scarab@localhost identified by 'secret';
```

### Setting up the MySQL parameters in Scarab

Use a text or program editor to open read-only (if possible) the default properties file in the build directory of your Scarab installation. Do not modify this file.

Use your editor to create a `build.properties` file in the same directory. Any parameter you will define in this file will take precedence on the one defined in `default.properties`. This allows you to keep reasonable default values for most parameters while redefining those you want (or must) change.

### Define the database type

You do not need to redefine the `scarab.database.type` parameter, because by default (as you can see by yourself in the `default.properties` file):

```
scarab.database.type=mysql
```

### Define the database user

Even in a default MySQL installation, you will probably have to define the database user (`scarab.database.username`), for instance:

```
scarab.database.username=scarab
```

If a password is required for this database user, you will also need to define the `scarab.database.password` key, for instance:

```
scarab.database.password=secret
```

## Define the database server

By default, the database is supposed to be on the same machine as the Scarab server:

```
scarab.database.host=127.0.0.1
```

If the database server isn't on the same machine as the servlet engine, you need to define the `scarab.database.host` key, specifying the network name of the MySQL server or its IP address.

If you are faced to database access errors to a local database (access denied), you may try to replace `127.0.0.1` by `localhost`.

## Option: downloading an optimized driver

If you build a "production" Scarab installation, you may choose to use a MySQL driver that is more recent and more performant than the one supplied in the Scarab distribution (this driver may not be bundled with Scarab because its licence is GPL).

Download this driver from the MySQL web site at the following URL: <http://www.mysql.com/downloads/api-jdbc-dev.html> and copy the jar in the lib directory of your Scarab installation.

You also need to declare its use in your `build.properties` file:

```
scarab.jdbc.driver.jar=mysql-connector*.jar
```

# PostgreSQL

## Installing PostgreSQL

Please refer to the PostgreSQL installation instructions.

## Setting up the PostgreSQL in Scarab

# Oracle 9i

## Installing Oracle 9i

Please refer to the Oracle 9i installation instructions.

## Setting up the Oracle 9i parameters in Scarab

---

# Chapter 4. Building Scarab

## Finishing Scarab configuration

The default Scarab configuration supplies reasonable values for most installation and execution parameters. However, you need to supply appropriate values for a few configuration parameters, and you may also choose to redefine a few other settings.

### Defining the server name and protocol

These parameters are required.

You will probably keep the default values for the following parameters :

- `scarab.http.context` (default value: `scarab`)
- `scarab.http.scheme` (default value: `http`)
- `scarab.http.scriptname` (default value: `/${scarab.context}/issues`)
- `scarab.http.redirect` (default value: `${scarab.http.scriptname}`)

but you will need to define the `scarab.http.domain` property (no default value) with your server name, and you may need to redefine the port as well (`scarab.http.port`, default value: 8080).

If you make a mistake defining these server parameters, your users will not be able to access Scarab from the emails sent by the issue tracker; correcting the mistakes will require rebuilding Scarab (which remains easy) and tweaking values in the database, which is far more tedious.

### Choosing the name and email address of the Scarab administrator

By default, this name will be `turbine@tigris.org`.

To change this default value, define the `scarab.user.username` property in your `build.properties` file:

```
scarab.user.username=scarab-superuser@mydomain.net
```

This user name will also become the default email address. If the name is not a valid address or if you prefer to use another email address, you could also define the `scarab.user.email` property; otherwise, the user name you supplied just before will also be the email address.

If you choose (later in the installation process) not to install the sample data supplied with Scarab, this user will be the first and only one in your brand new Scarab installation. Its password will be "1" (the digit one, without quotes).

The default first name will be "Scarab" and the name will be "Administrator" (the first and last name of the connected user are displayed in the top right corner when Scarab is running).

For a localized installation you may prefer to define the properties

```
scarab.user.firstname=Administrateur
```

```
scarab.user.lastname=Scarab
```

(which allows you to translate and change the order of the words) or you may choose to user here the first and last name of the person who manages Scarab in real life.

## Defining the mail server

The default mail server (for sending emails) is the machine on which Scarab is running (localhost). Thus Scarab expects to find a running SMTP server on the same machine to send its messages. If this is not the case (which happens quite often), you need to define a property with the network name of your SMTP server, e.g. the SMTP server of your ISP:

```
scarab.system.mail.host=smtp.example.net
```

## Localizing your Scarab installation

The language of the Scarab user interface (as it appears in the user's browser) isn't defined by the installation or configuration parameters (it is defined by the user's browser linguistic settings). So there isn't anything special to configure to have Scarab localized in another language -- if the corresponding resource bundle is available, of course.

There are, however, two settings you may want to define at this stage.

## Character set used for sending email

You may define for this purpose the `scarab.email.encoding` property, for instance:

```
scarab.email.encoding=ISO-8859-1
```

```
scarab.email.encoding=UTF-8
```

## Default language definition

You may also define the `scarab.locale.default.language` and `scarab.locale.default.country` properties, for instance:

```
scarab.locale.default.language=fr
```

```
scarab.locale.default.country=FR
```

These parameters are used (at least) to define the language of the default/sample data in the database (if a corresponding resource bundle is available).

## Preparing the deployment on another servlet engine

If you plan not to use the Tomcat server that is built together with Scarab ("sandbox"), you must have the Velocity templates -- used to generate the web pages -- in the web application hierarchy. To do this, you must define the following property:

```
scarab.copy.templates=true
```

## Building Scarab

From here, building Scarab is a fairly simple process: go to the `./build` directory and type

ant

---

# Chapter 5. Building the Scarab database

## MySQL

### On Unix/Linux

From the root of your Scarab installation, type

```
cd src/sql  
./create-db.sh
```

For a production installation, you may want not to include the sample data supplied with Scarab and start with an empty database; just add the `-e` command line argument:

```
./create-db.sh -e
```

If the MySQL server is on another machine, or if you must authenticate to execute administration commands, you may supply additional arguments to the script using the usual MySQL syntax, for instance :

```
./create-db.sh -u jon --password -P 3306 -h mysqlserver.example.com
```

### On Windows

From the root of your Scarab installation

```
cd src/sql  
create-mysql-database.bat
```

If your database server is on another machine, or if you have to authenticate to execute administration commands, you will need to edit the `create-mysql-database.bat` script.

---

# Chapter 6. Running Scarab

## Running Scarab with its own Tomcat server

To run Scarab with the minimal Tomcat server that has been built together with it, simply go to the target directory of your Scarab installation and type:

```
./scarab.sh &
```

---

## **Part II. Scarab system administration after installation**

---

---

## Table of Contents

7. Backups and Restores .....	22
Backing up Scarab information .....	22
Backing up the database .....	22
Backing up other information .....	22
Restoring Scarab information from a backup .....	22
Restoring the database .....	23
Restoring other information .....	23
8. Moving an existing Scarab installation to another server .....	24
Modifying the database when the Scarab server changes .....	24
Modifying the SCARAB_GLOBAL_PARAMETER table .....	24
Modifying the SCARAB_MODULE table .....	24
Modifying the SCARAB_ISSUE table .....	24

---

# Chapter 7. Backups and Restores

To restore a Scarab installation, if the server fails for instance, you must perform backups on a regular basis.

You will have to backup the data contained in the Scarab database, but also the directories that contain the attachments, the indices used by Lucene for full-text search and the objects serialized by Intake.

## Backing up Scarab information

### Backing up the database

#### MySQL

On the command line, type for instance:

```
mysqldump scarab > filename.sql
```

If your MySQL server requires authentication, you will have to supply the appropriate arguments on the command line, for instance:

```
mysqldump scarab --user=scarab --password > filename.sql
```

### Backing up other information

The directories used to store attachments and Lucene indices, and the file containing the objects serialized by Intake have been defined at build time. The corresponding properties are:

- `scarab.attachments.path`
- `scarab.lucene.index.path`
- `scarab.intake.serialize.file`

You need to back up these two directories and this file as well as the database information.

If you hadn't redefined the three parameters above when you built Scarab, all these objects are in the Scarab file hierarchy under `target/webapps/scarab/WEB-INF`.

On Windows, the simplest solution is probably to copy the two directories and the Intake file in a zip archive.

On Linux, you can do the same with the following command (for those not familiar with unix commands):

```
tar zcf scarab.tgz attachments index intake-xml.ser
```

If one or several of the parameters has/have been redefined, you will need to find where the directories and the file are to be able to archive them.

## Restoring Scarab information from a backup

## Restoring the database

### MySQL

Connect to MySQL, for instance :

```
mysql
```

You may have to specify a user name and/or password if your MySQL server requires authentication :

```
mysql --user=name --password
```

If necessary, you may have to delete a corrupted Scarab database :

```
drop database scarab;
```

Create a new Scarab database and select it :

```
create database scarab;
```

```
use scarab;
```

Restore the data you had backed up with mysqldump :

```
source filename.sql;
```

That's it !

## Restoring other information

Uncompress the archive you made previously with attachments, Lucene indices and objects serialized by Intake, in the directories defined when Scarab was built, as explained above.

Your Scarab installation is now ready to restart.

---

# Chapter 8. Moving an existing Scarab installation to another server

If your Scarab server fails -- or for any other valid admin reason -- you may want to move an existing Scarab installation to another server.

Unless the settings and the network name of the new machine are strictly identical, you will need to perform a new Scarab installation from scratch, tuning a few configuration parameters.

If the network name of the target machine differs from the original one, you will also need to perform three modifications in the database, even if the database server itself does not change.

## Modifying the database when the Scarab server changes

### Modifying the SCARAB\_GLOBAL\_PARAMETER table

If you change any of the following parameters on the new installation: `scarab.http.domain`, `scarab.http.scheme`, `scarab.http.scriptname` or `scarab.http.port`, you will need to update the corresponding entries in the `SCARAB_GLOBAL_PARAMETER` table. For instance :

```
update SCARAB_GLOBAL_PARAMETER set VALUE='newserver.example.com' where
NAME='scarab.http.domain' ;
```

```
update SCARAB_GLOBAL_PARAMETER set VALUE='8081' where
NAME='scarab.http.port' ;
```

### Modifying the SCARAB\_MODULE table

You may need to update the domain (server) name in the `SCARAB_MODULE` table :

```
update SCARAB_MODULE set DOMAIN='newserver.example.com' ;
```

### Modifying the SCARAB\_ISSUE table

You may need to update the domain (server) name in the `SCARAB_ISSUE` table :

```
update SCARAB_ISSUE set ID_DOMAIN='newserver.example.com' ;
```