

BR*Tools Studio for Oracle DBA



September 2008



Copyright

© Copyright 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group. Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.






Java is a registered trademark of Sun Microsystems, Inc

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, xApps, xApp, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help → General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
<i>Example text</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example text	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

BR*Tools Studio for Oracle DBA.....	5
Getting Started with BR*Tools Studio.....	7
Installing BR*Tools Studio	7
Running BR*Tools Studio for the First Time	10
BR*Tools Studio: Console.....	11
BR*Tools Studio: Wizard.....	12
BR*Tools Studio: Administration	13
BR*Tools Studio: Landscape.....	14
BR*Tools Studio: Case Study.....	15
BR*Tools Studio: Role Concept	18



BR*Tools Studio for Oracle DBA

BR*Tools Studio is a graphical web-based front end for BR*Tools. You can use BR*Tools Studio for all Oracle database administration tasks.

BR*Tools Studio has a three-tier architecture in which BR*Tools users no longer directly control jobs, but instead create job packages on the client and pass these packages over to BR*Tools Studio server. The server in turn executes the jobs independently on the target database instance. This architecture meets the requirements for quick-start favorite tasks and role-based, user-specific contexts. The BR*Tools Studio engine executes and keeps track of these items.

Integration

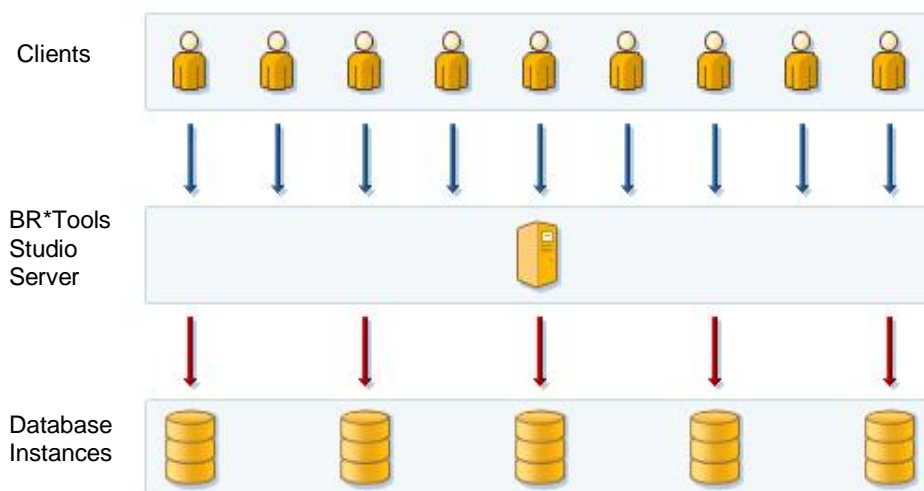
We have designed BR*Tools Studio as follows:

- Multi-instance three-tier architecture

BR*Tools Studio is a browser-based, three-tier application, which uses the internet browser for the graphical display on the client. The advantage is that you do not need to install software on the client. All you need is a normal internet browser. This means you can run the client anywhere, on any regular PC or workstation.

In addition, there is no need to install BR*Tools Studio multiple times – that is, for each of your database instances. You only need to install BR*Tools Studio once. A single instance of BR*Tools Studio server can handle all your database instances simultaneously.

The BR*Tools Studio architecture is configured like a star with multiple clients logging in to a single central server, which in turn connects to multiple database instances.



BR*Tools Studio Architecture

- Role-based multiuser concept

BR*Tools Studio has a full multiuser concept. For each database instance you administer with BR*Tools Studio, you can create multiple, password-protected administration user accounts in BR*Tools Studio.

You can personalize all users with a special role that defines their administration privileges throughout the database instance and BR*Tools Studio server. The different roles enable full auditing of operations in BR*Tools Studio.

- Clients

A user logs on as a client to BR*Tools Studio server. Not only does BR*Tools Studio allow several users to be logged in at the same time, it even allows any individual user to be connected to BR*Tools Studio server in multiple client sessions at the same time. In this case, you can see any changes you make in one client session in all other client sessions of your user.

- Server

The BR*Tools Studio server is the engine room of BR*Tools Studio. It executes incoming jobs and administers all items. If the server is down, all jobs scheduled for the downtime period fail. No clients can connect or log on to BR*Tools Studio until the server is up again.

- Job queue, history, and scheduling

With completion of a job on the client, the job is sent over to BR*Tools Studio server, which then handles the job autonomously:

- If the job is scheduled for immediate execution, BR*Tools Studio server starts the job at once.
- If the job is scheduled for execution at a later time, the server keeps the job in the job queue and automatically executes it at the requested point in time.

With the recurrence option, you can instruct the server to keep repeating the same job with the specified recurrence interval. You can always keep track of executed jobs as BR*Tools Studio server maintains a full history of all jobs that have been executed on the server.

- Favorite tasks

You can save any preconfigured job that you want to keep for frequent use as a favorite task in BR*Tools Studio. You can then call and submit the job with one click, without having to reenter all the job parameters.

- Auto-update status display

A special status display in the BR*Tools Studio console informs you in one glance about the state of the database instance at any time. The status display is automatically refreshed so you do not need to trigger a manual refresh.

- Reconnect

When the client sends an instruction to the server to perform an operation, the BR*Tools Studio server takes over and executes the operation autonomously. The client does not need to wait for the end of the operation. This means you can log off the client and log on again later – at a different location, if required – to see the status of the operation.

- Secure connection

To guarantee security of your data, BR*Tools Studio uses connections based on Secure Socket layer (SSL)/HTTPS between the server and its individual clients.

Features

BR*Tools Studio is divided into these main areas:

- [Console](#)
- [Wizard](#)
- [Administration](#)
- [Landscape](#)



Getting Started with BR*Tools Studio

This section tells you how to install BR*Tools Studio and what to do when you run it for the first time.

Process

1. You install BR*Tools Studio.
2. You run BR*Tools Studio for the first time.



Installing BR*Tools Studio

Prerequisites

To run the *setup program* for BR*Tools Studio you need to meet the following requirements:

- 20 MB of disk space on your BR*Tools Studio host
- Java Runtime Edition (JRE) 1.4.2 or higher

- SAPCAR extraction software as described in SAP Note [212876](#).
- These executables in your `PATH` variable:
 - SAPCAR
 - java

To run BR*Tools Studio *after installation* you need to meet the following requirements:

- Java Development Kit (JDK) 1.4.2 or higher
- SAP BR*Tools 7.10 Patch 12 or higher, or BR*Tools 7.00 Patch 34 or higher
- Web browser Microsoft® Internet Explorer or Mozilla® Firefox
- A remote shell tool of your choice such as RSH, SSH, or PuTTY (optional)
- Oracle Containers for Java EE (OC4J) 10g or higher (optional)

You can install and run BR*Tools Studio in two modes:

- Dedicated server

You install BR*Tools Studio on your chosen database host to administer the local database instance only.

- Multi-instance server

You install BR*Tools Studio on a standalone middle-tier host and add database instances as remote instances to the instance portfolio of the BR*Tools Studio server.

BR*Tools Studio does *not* require its own database.



Recommendation

Although BR*Tools Studio lets you add additional database instances to a dedicated BR*Tools Studio server, we strongly discourage you from doing so. This is due to the increasing load BR*Tools Studio server generates on its host as the instance portfolio expands.

Instead, we recommend you to run BR*Tools Studio server on a platform *without* a database host when running in multi-instance server mode, to avoid overloading the database host.

Procedure

1. Set up the keystore file on the BR*Tools Studio host for incoming client calls:



Note

BR*Tools Studio implements a client-server architecture in which the BR*Tools Studio server resides in a web application server on the database host, while clients connect through a browser from outside.

To ensure secure communication between the clients and the server, BR*Tools Studio uses the Secure Hyper Text Transfer Protocol (HTTPS) for communication, based on encryption via Secure Socket Layer (SSL). This method requires a keystore file, which serves as a server certificate for incoming client calls.

- If there is an SSL keystore file already available on the server that you intend to use, keep the corresponding file name and directory and the keystore password available during the installation of BR*Tools Studio.
- If there is no SSL keystore file available, create a keystore file on the server before starting the installation.



Note

You can create a keystore file using the `keytool` tool contained in your Java installation. You can find more information on this tool in your Java documentation.

2. Install BR*Tools Studio in dedicated server or multi-instance server mode:

- Dedicated server mode,

Log on to your database server as user `orasid` (UNIX) or `sidadm` (Windows) and create the following directory as your BR*Tools Studio home directory:

```
oracle/<SID>/sap/brtools/studio
```

- Multi-instance server mode

Log on to the chosen middle-tier host and choose any directory as your BR*Tools Studio home directory.

3. Move to the home directory of BR*Tools Studio and extract the BR*Tools Studio installation package to the directory with this command:

```
SAPCAR -xvf BRSTUDIO710EN_XXX.SAR
```

4. In your BR*Tools Studio home directory, call one of the following:

- UNIX

```
ssh setup.csh
```

- Windows

```
setup.cmd
```

The BR*Tools Studio installer window appears and guides you through the installation process.

5. If you have installed BR*Tools Studio in multi-instance server mode, change to BR*Tools Studio home directory and install Oracle Containers for Java EE (OC4J) in the following subdirectory:

web-app-server/oc4j/

Oracle Containers for Java EE (OC4J) is available as a download from the Oracle Technology Network (OTN).

In dedicated server mode, BR*Tools Studio automatically uses the installation Oracle Containers for Java EE (OC4J) contained in the Oracle home directory of the database.



Note

BR*Tools Studio does not come with its own copy of BR*Tools executables. Instead, it uses the executables already contained on the individual database instances. Therefore, a correctly set-up SAP environment is crucial for BR*Tools Studio to work with each instance after installation

Result

After successful installation, BR*Tools Studio runs “out of the box”. You do not need to do anything else. However, make sure that you read [Running BR*Tools Studio for the First Time](#).



Running BR*Tools Studio for the First Time

Since BR*Tools Studio is designed as a client-server application, you need to start up BR*Tools Studio server first as described below.

Procedure

1. In your BR*Tools Studio home directory, call:

- o UNIX

```
    csh server.sh
```

- o Windows

```
    server.cmd
```

This script starts a web application server based on Oracle Containers for Java EE (OC4J), which serves as the runtime platform for the web application of BR*Tools Studio server.

You need to run this command whenever you want to start the server for BR*Tools Studio in future.

2. Log on to BR*Tools Studio server through a web browser from any client using the following URL:

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



Note

Once BR*Tools Studio server is available, a client logon window appears. Make sure that you have enabled pop-ups for this site so that the logon window is visible.

You need to run this command whenever you want to start a client session with BR*Tools Studio.

The client logon window provides you with a list of all instances currently maintained for administration with BR*Tools Studio. The only instance available directly after installation is the *Server* instance, with which you can maintain the server landscape.

3. Log on to this instance with the Server Administrator user, called “Administrator,” and the password specified during installation.
4. On the *Landscape* tab of the server administration, choose ► *Server* → *Startup* ◀ and follow the roadmap to start up the BR*Tools Studio server.

You need to run this command whenever you want to start a client session with BR*Tools Studio.



Note

After startup, BR*Tools Studio server keeps running until an explicit shutdown call is made. We recommend you to keep the server up continuously to allow users to log on and enable the server to autonomously execute scheduled jobs.

5. On the *Landscape* tab of the server administration, choose ► *Instance* → *Create* ◀ to add an initial database instance for administration with BR*Tools Studio.
6. Log off from the server instance to return to the logon screen, and log on to the newly added database instance with the user and password specified during instance creation.

Result

The instance is now ready for administration with BR*Tools Studio.



BR*Tools Studio: Console

You can check the instance, start your favorite tasks, and view the jobs in process using the console, which is always visible on the left of the screen.

- *Instance status*

You can see current status of the database, whether an SAP system is connected, and the archiving status.

BR*Tools Studio automatically updates the display at frequent intervals to give you a real-time view of the database. The traffic lights help you intervene quickly to avoid problems developing.

- *My favorite tasks*

You can see a list of your favorite tasks, which you can start with a single click.

- *Jobs in Process*

You can see a list of jobs in process with traffic lights indicating status.



BR*Tools Studio: Wizard

The tab *BR*Tools Studio Wizard* contains the representation of the BR*Tools interface in BR*Tools Studio, yet with a slightly different menu structure. You can perform a variety of operations on the database using this tab.



Note

Depending on your BR*Tools Studio role, you might not be able to see all the entries listed below.

Features

- *Instance*

You can start up or shut down the database instance, or perform further instance-specific operations. You can also view a detailed status report.

- *Database*

You can get a detailed view of:

- Database architecture: tablespaces, tables, and indexes
- File structures on operating-system level: data files, control files, redolog files
- SAP object owners

- *Maintenance*

You can perform the major database administration functions, such as storage administration, reorganization, statistics maintenance, and export/import of database objects.

- *Availability*

You can safeguard the availability of your database by performing backups, database copies, restore, recovery, and disaster recovery.

- *Advanced*

You can perform advanced BR*Tools functions not available in BR*Tools Studio, using the command line.



Caution

Only perform *batch operations* that do not require input from you during execution.



BR*Tools Studio: Administration

You can administer all BR*Tools Studio items that belong to the instance you are currently logged in to with the tab *Administration*.



Note

Depending on your BR*Tools Studio role, you might not be able to see all the entries listed below.

Features

- *Users*

Each user in BR*Tools Studio has an individual password-protected account with a user profile. The user profile contains details of the default database user and initialization profile for BR*Tools. The user account also contains all jobs and favorite tasks created by the user.

The roadmap lets you view your profile (which you can also change), favorite tasks, and jobs.

If your role has sufficient authorization, you can also view and alter the profiles of other users, and create or delete BR*Tools Studio users.

- *Roles*

Every user is assigned one of four discrete roles in BR*Tools Studio. The role describes which operations the user is allowed to perform. We designed the available roles to cover the likely task areas required by DB administrators at different levels.

You can view your own role here. If your role has sufficient authorization, you can list all roles.

For more information, see [BR*Tools Studio: Role Concept](#).

- *Favorite tasks*

If you repeatedly execute an operation, BR*Tools Studio lets you store it as a favorite task, which you can later start with a single click. You do not need to

enter all the settings again. Of course, you can change your existing favorites as required.

You can create, change, or delete your own favorite tasks. If your role has sufficient authorization, you can also delete a favorite task of another user.

- *Jobs*

Jobs already sent to the server are stored here. Jobs for immediate execution are executed immediately. You can also schedule jobs for one-off, periodic, or later execution. BR*Tools Studio also stores already executed jobs here with a result status. From here you can check the associated job logs.

You can view your own jobs here and, depending on your role, clear your job history if required. If your role has sufficient authorization, you can also view jobs of other users and clear their job history users.



BR*Tools Studio: Landscape

The BR*Tools Studio pseudo-instance Server lets you maintain the server landscape. You can log on to the server instance with the Server Administrator user, called “Administrator”, and the password provided during installation.

Features

- *Server*

You can shut down the server and change the Server Administrator password. Server shutdown is suspended until all active jobs have finished before safely shutting down.



Caution

After the downtime has finished, recurrent jobs scheduled for future execution are preserved and run later at the correct time. However, all jobs scheduled for execution *during* the downtime are lost.

- *Clients*

You can see all clients currently logged on to BR*Tools Studio server.

- *Instances*

You can see all instances that are currently on the administration portfolio of this BR*Tools Studio server and you can add new instances to the portfolio.



BR*Tools Studio: Case Study

We have designed BR*Tools Studio to be largely self-explanatory. Nevertheless, this case study shows you how to perform a simple operation.

Prerequisites

- The BR*Tools Studio server is up and running.
- You log on as BR*Tools Studio administrator or database administrator to a database instance of your choice. You cannot perform this operation if your authorization is database operator.
- The database is down at the start of the operation.
- You perform this operation from the BR*Tools Studio Wizard.

Procedure

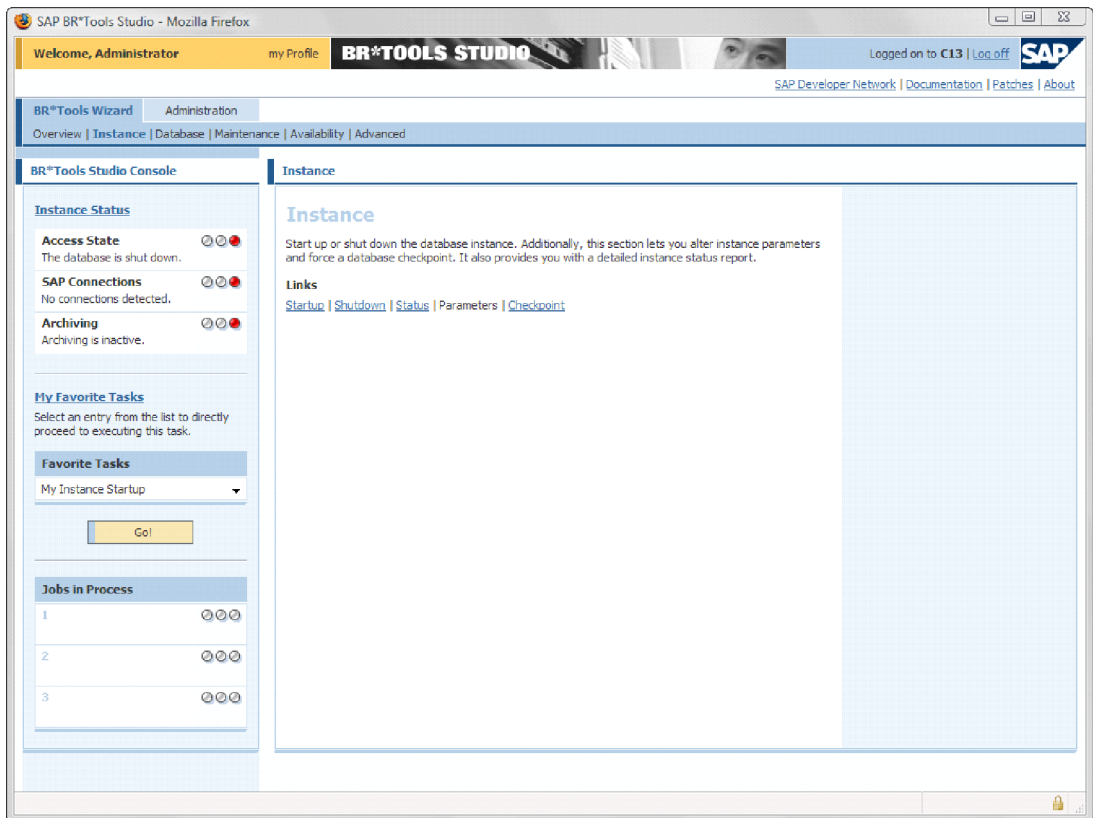
1. Choose ► *BR*Tools Studio Wizard* ⇒ *Instance* ◀.



Note

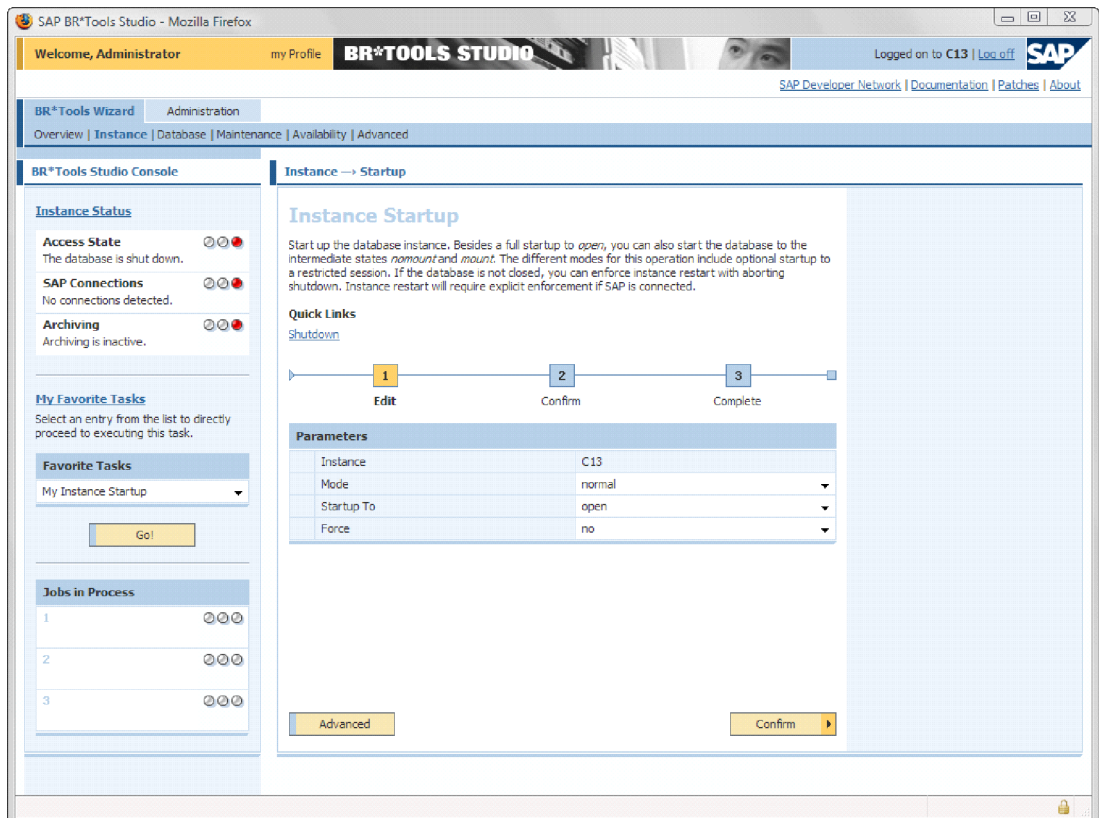
Since it is a commonly used operation, you can also find it on the *Overview* tab of BR*Tools Studio Wizard.

You see the following screen with the red lights in the *Instance Status* display indicating that the database instance is currently shut down:



2. Choose *Startup* to begin the startup.

You see the following screen with the default parameters for the instance startup:

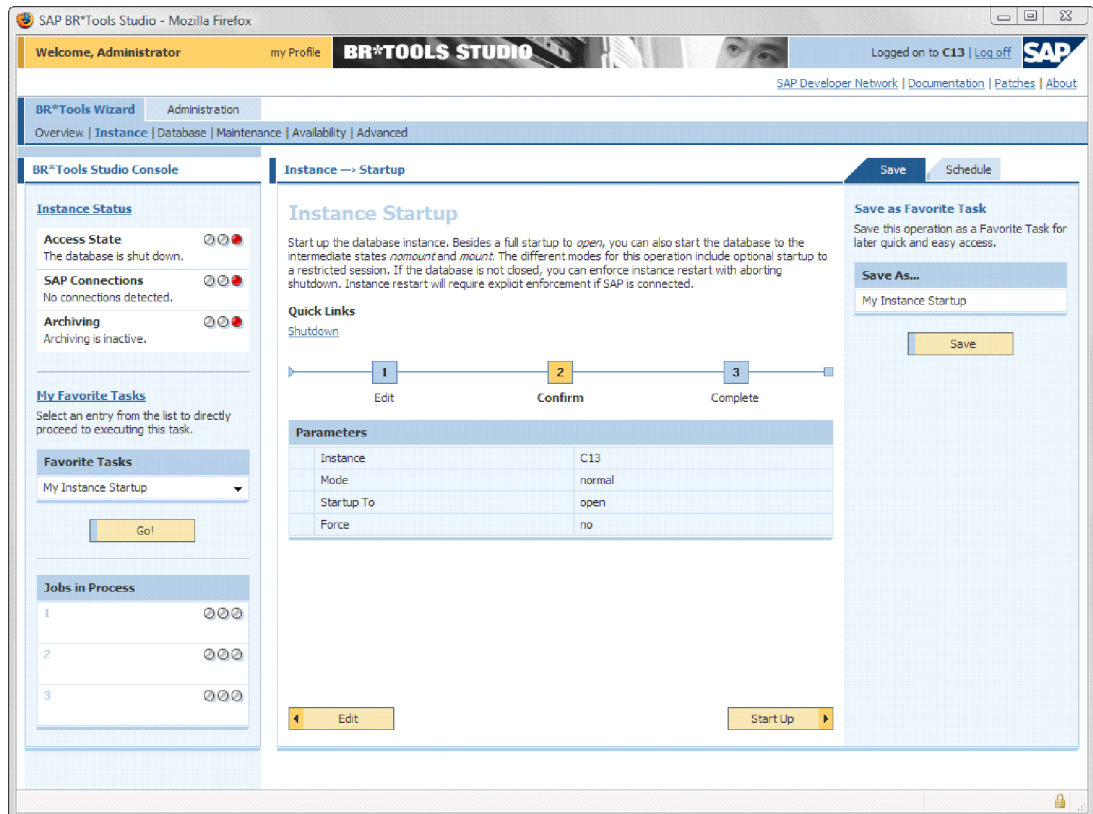


The progress indicator highlights *Edit Parameters* in the first step, indicating that this is where you currently are.

3. Choose one of the following:

- *Advanced* to change the default parameters
- *Confirm* to accept the parameters displayed and continue the startup

After *Confirm*, you see the following screen:



The progress indicator now highlights *Confirm Startup* in the second step.

4. Choose one of the following:

- *Edit* to go back and change the parameters
- *Startup* to start the database instance



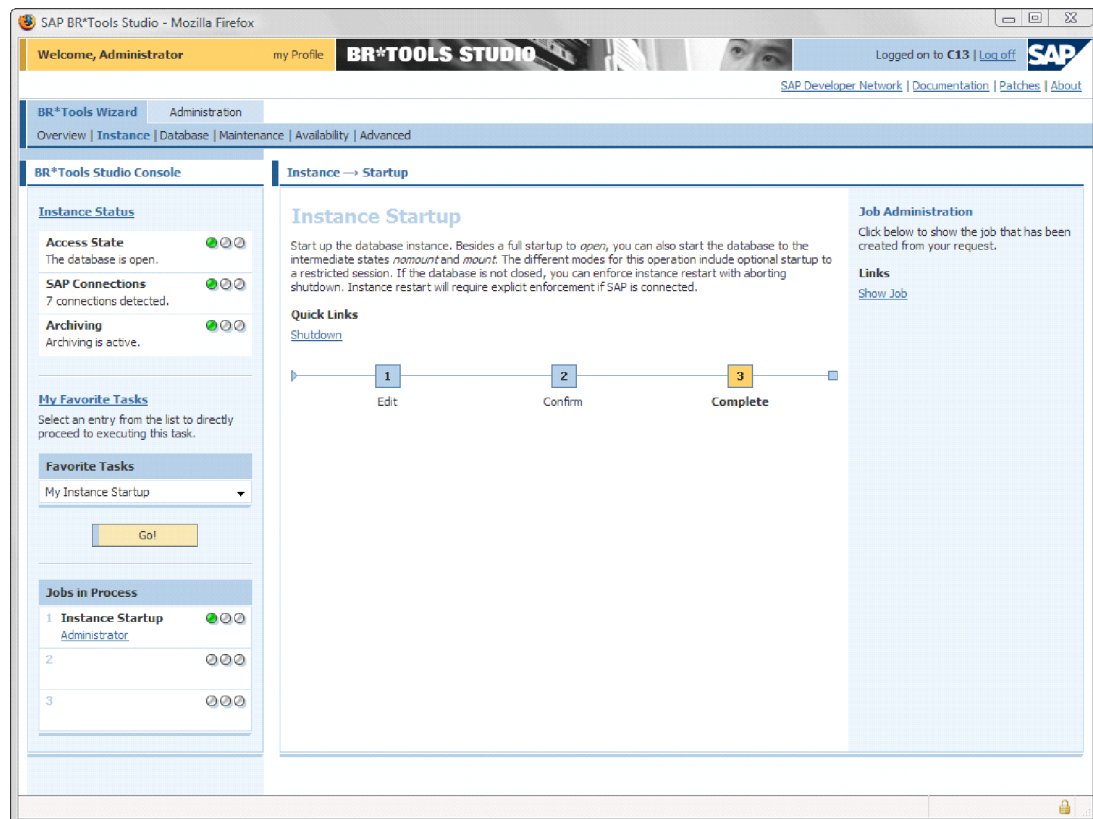
Note

If required, from this screen you can:

- Use the *Save* tab to save this operation as a favorite task with a name of your choice
- Use the *Schedule* tab to schedule this operation to start at a later time or to run repeatedly at a specified interval

Result

The database instance starts, which might take several minutes. Finally, you see the green lights in the *Instance Status* display.



The progress indicator finally highlights *Startup Complete* in the last step, indicating that you have now finished the operation.

If required, you can choose *Show Job* under *Job Administration* on the right side to reach the detail display of the completed job in BR*Tools Administration. There you can see the BR*Tools job log to check for errors.

BR*Tools Studio: Role Concept

BR*Tools Studio offers a role concept in which you assign specific privileges to each BR*Tools Studio user. When you create a user, you assign the user exactly one role, which you *cannot* change in future.

BR*Tools Studio offers the following roles:

- BR*Tools Studio Administrator

Users of this role have full access to the following functions:

- Database administration (DBA)
- Instance-specific Studio administration

They can see their own Studio objects and those of all other Studio users.

- BR*Tools Studio Operator

Users of this role only have full access to DBA functions. They do not have any Studio objects of their own but can see Studio objects of all other users.

- Database Administrator

Users of this role have full access to DBA functions and limited access to the functions of instance-specific Studio administration. They can only see their own Studio objects.

- Database Operator

Users of this role have limited access to DBA functions and instance-specific Studio administration. They can only see their own Studio objects.



Note

While not strictly a role, there is also the Server Administrator user, called “Administrator”, with which you can maintain the server landscape by logging on to the “instance” Server.

The roles of Database Administrator and Database Operator are fully subject to the auditing concept. This means that operations executed by these users can always be uniquely traced to the user. It is impossible for these users to eradicate traces of their activity.