NETWORK 23

OTR - Oracle Tablespace Report

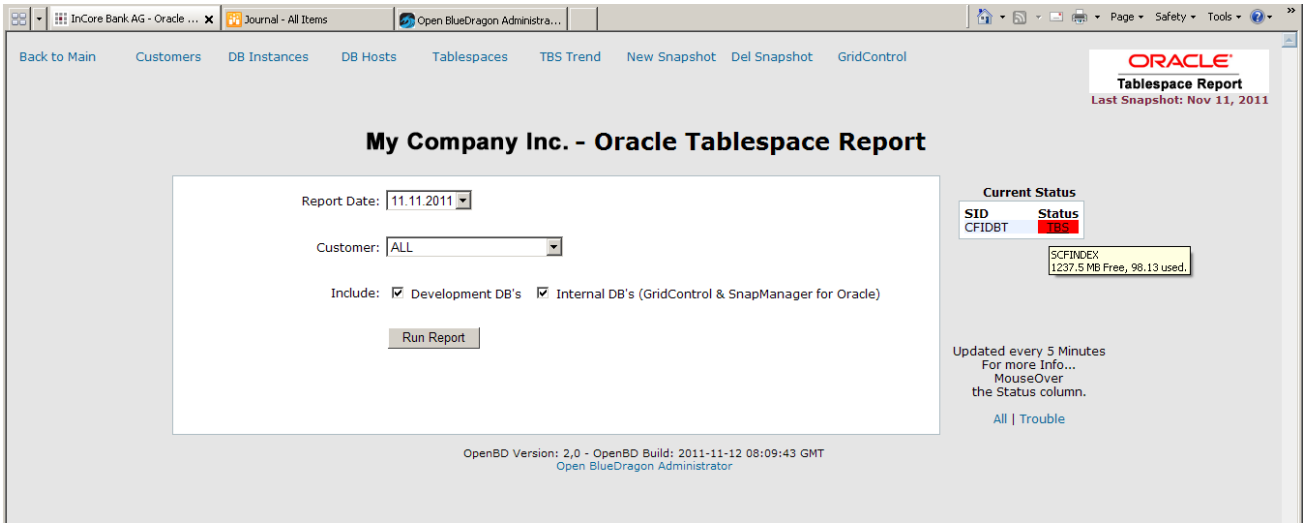
Open Source Project

Mats Strömberg

2011

http://www.network23.net/otr/

Table of Contents



Oracle® Tablespace Report

Open Source Project

Introduction

Oracle Tablespace Report is used to gather various statistics e.g. tablespace usage (allocated, used and free) space. The statistics are stored centrally in the OTR Repository located in an OTR Instance.

This project got started out of a work from a Danish friend, Lars-Bo Vanting, at the time we worked together at T-Systems Schweiz AG back in 2005.

Initially it was only based on Tables, Views and PL/SQL. The current version has expanded on the basis and added a web GUI (based on the excellent open source project Open BlueDragon (http://www.openbd.org).

The need for the enhancement of this tool was due to the fact that I had roughly 80+ Oracle instances, about 100 Linux/Solaris Servers and 6 NetApp Storage systems to manage… all alone. (The advantage was that my Team meetings went very fast… didn't have to argue too much with my dual personality ☺).

This version of OTR does a bit more than just collect tablespace usage on a weekly basis. It monitors each database instance every 5 minutes and reports back if a tablespace is getting full and one can directly act on the upcoming problem, extend an tablespace or add a new tablespace file without the need to do this over the Grid Control or manually.

Since I was all alone, having to manage this amount of databases, along with application servers and storage I needed a way of getting this kind of work as easy as possible. That way I could let anyone solve any acute problem coming up without me being around all the time. During 4 years in this company I haven't had more than roughly 2 weeks holiday/year… and those 2 weeks I get stuck having to solve issues over my Mobile phone.

The reason for putting this up as open source is thanks to the great Project run by the OpenBD Team. They have created a great tool which is a very serious alternative for Adobe ColdFusion and all at no cost. The team around this project is great and issues coming up are solved very fast. Support is done on Google Groups and there is always someone around helping out, core developers or regular users on the list… no matter who, you will always get help.

So if I can give something back to the OpenBD project and the community and at the same time help other DBA's making their work easier, this is a small step in that direction.

Short background about me…

I've been working with ColdFusion since the mid 90'es back in the days when Jeremy and JJ Allaire were running the business. Started with Cold Fusion 2.0 and up to ColdFusion 4 (some time in the years between 3.1 and 4.0 the space disappeared in the ColdFusion name) and was, back then, an early adopter of the FuseBox framework.

In 1999 I moved from Sweden down to Switzerland and in 2001 came in contact with Oracle and for all with some really good Danish Oracle cracks like, Lars-Bo Vanting (now at BlueGecko in Denmark) and some of his Oracle friends, and have been working with Oracle 8.1.7 – 12g since then. I'm a big fan of the Oracle Enterprise Manager and later Oracle Grid Control (today Oracle Cloud Control). All these years though I never left the ColdFusion train. It has been more of a hobby since my daytime work has been around Oracle and System Administration, but the passion for the CFML world is there to stay.

What's needed to get OTR running?

- Basis for the Oracle® Tablespace Report, from here on simply OTR, tool is the Oracle® Grid Control 10g so this is the first thing to be installed if not already done. Anyone running 10 or more Oracle Instances should never be without the Grid Control!!!

  I'm about to do tests on the Oracle® Cloud Gontrol 12c and make sure we're able to use this as well…

- Open Bluedragon release 2.0.1 can be downloaded at http://www.openbd.org/download/

  The easiest installation is using the Ready2Run Jetty+OpenBD download. Recommended is to download the OpenBD Desktop as well. With this you will be able to simply test and make additions on your own on your local PC or workstation.

- JDK 6
- The otr.war which can be found at http://www.network23.net/otr/

  The complete source will be available on Google Code http://code.google.com/p/oracle-tablespace-report/

- The SQL files needed to setup the Repository on your OTR Repository Database. Also downloaded from http://www.network23.net/otr/

  To avoid license problem with Oracle, the OTR Repository should NOT be installed in the Grid Control, Cloud Control or a RMAN Respository Database! The OTR can very well be run on a Standard Edition DB or even an Oracle XE instance. Future releases of OTR might even be possible to use MySQL as a Repository.

What does it look like?

We're assuming the OTRREP schema and its objects have been created on the OTR Repository database.

The Main screen.

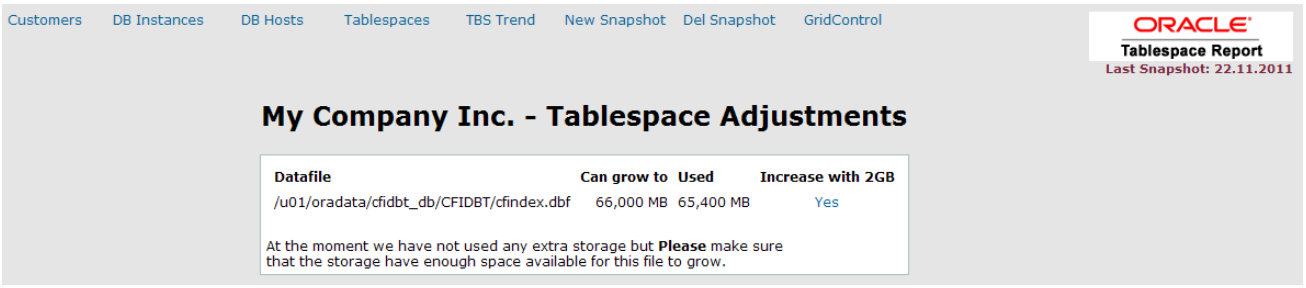
From here we will administrate our Customers, the Database Instances and the relationship of Customer/Database instance(s) and the Tablespaces used.

From here one will generate reports of space usage at a defined point in time. This can be a report containing database instances for all customers or for a single customer. Reports can be stored as Excel files or as PDF files.

On the right side is the monitoring/alert pane where Instances with some sort of problem coming up will be listed. It will display if the Instance is down or if a Tablespace has a problem. With a mouse-over on a red alert the actual tablespace will be shown and how much free space in MB is still available and the "real" % used. With "real" means it's calculating the free space in % based on the "can grow to" value for the tablespace.

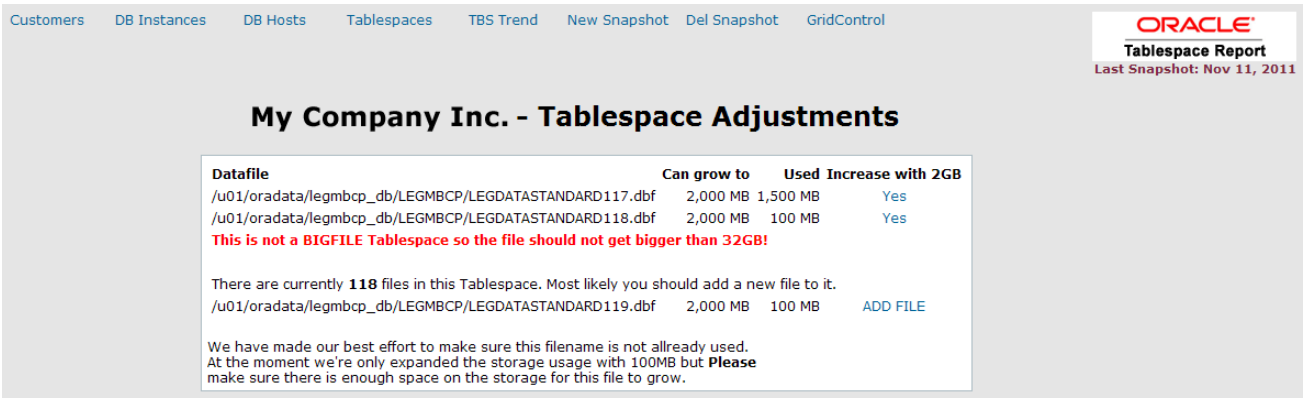
With a click on the red TBS alert, you get the possibility to adjust the tablespace with just one click.

Assuming we have a space problem on a BIGFILE tablespace.



With just one click this tablespace will extend the “Can Grow to” with another 2GB.

If it would be a non-BIGFILE tablespace…



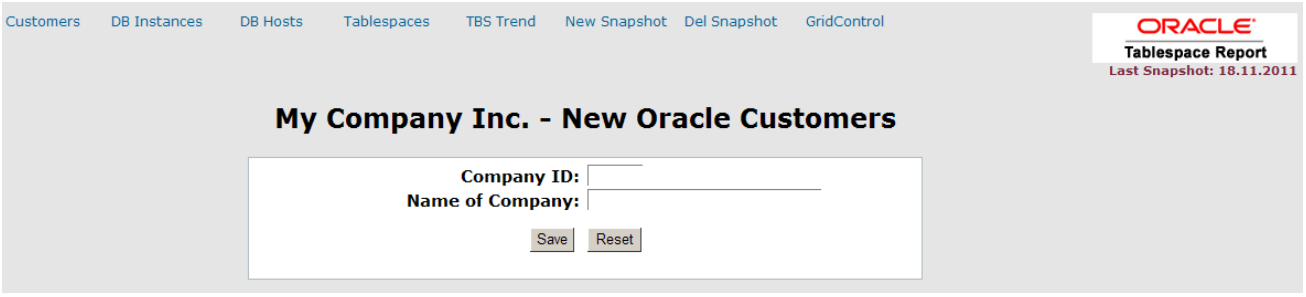
A list of the files within this tablespace, that have autoextend still on, will be displayed. You can select to increase the “Can grow to” on one of these datafiles or add a new 2GB file which will have its initial size set to 100MB and the “Can grow to” to 2GB.

Customers

Main screen for customers

From this screen we will administrate our customers. The company info contains Company ID or Mandator and a Customer name.

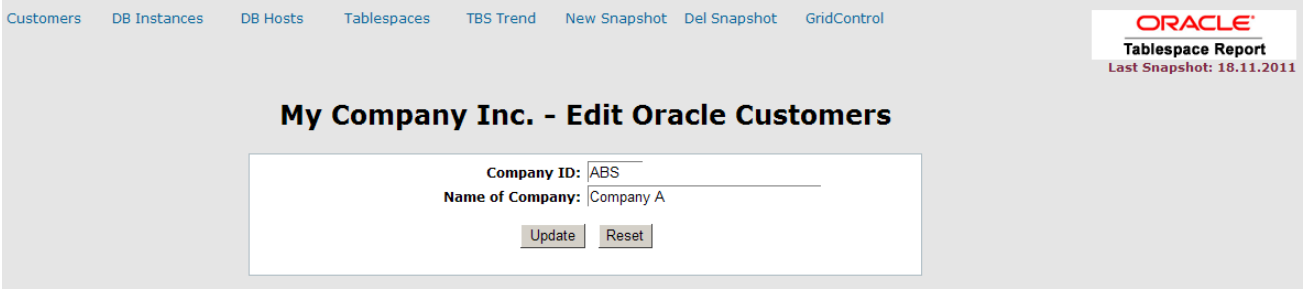
Adding a new customer



Simply fill out the form and click on Save.

Customer ID is a 3 letter short name of the customer. This is later used as a connection to the database instance and the tablespaces used by this customer.

Edit a customer

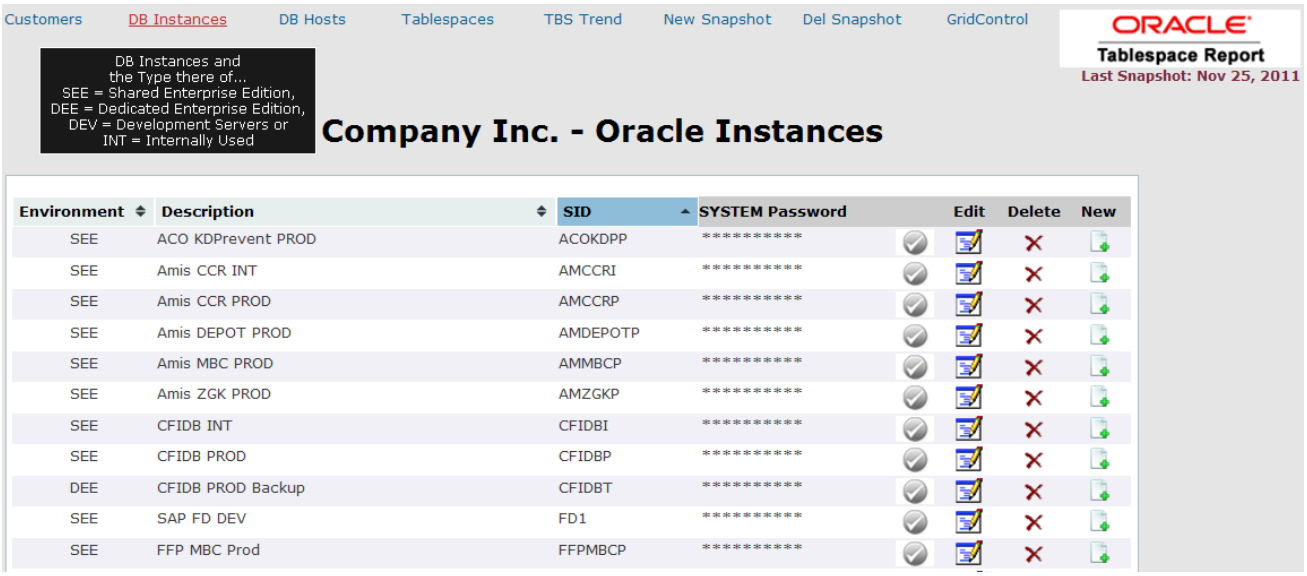


Note: Changing the Customer ID will bring a problem with the collected statistics and with the connection to the tablespaces, so try to keep this unchanged…

DB Instances

In this module you will register all your Oracle Instances.

Main screen for Oracle Instances

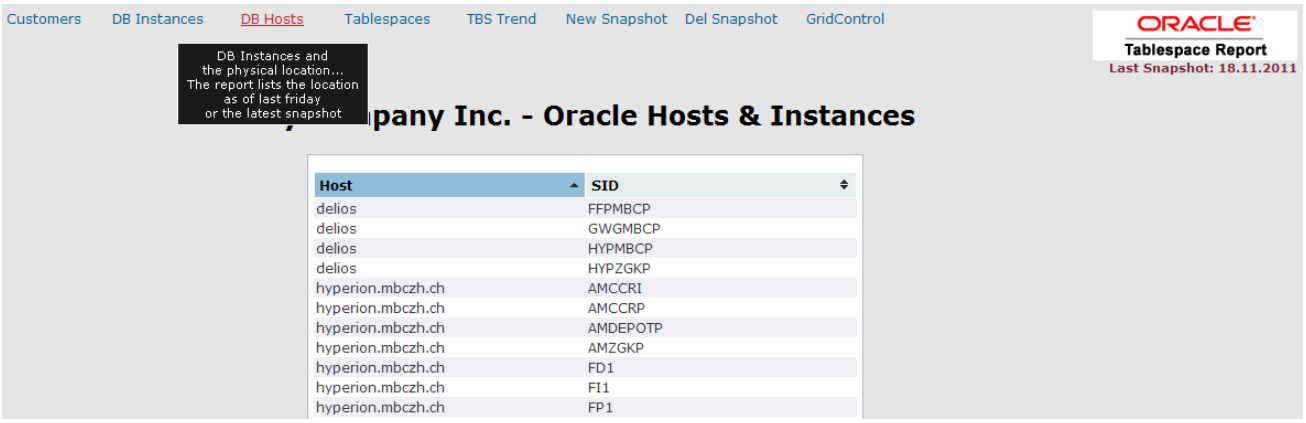


It contains Info like Oracle SID, what type of instance this is, DEE = Dedicated Enterprise Edition, DSE = Dedicated Standard Edition, SEE = Shared Enterprise Edition, SSE = Shared Standard Edition, DEV = Development Instances or INT = Internal Enterprise or Standard Edition (might be the Grid Control Instance, a RMAN Instance or a SnapManager for Oracle Instance as an example). It also contains a short description for the Instance. This is usually related to an Application and/or Production/Integration/Test Instance.

System password (used to monitor and increase Tablespaces). This password is encrypted in the OTR repository. To check if the password is OK, just click on the  . It will turn if OK otherwise  .

DB Hosts

Main screen for DB Hosts

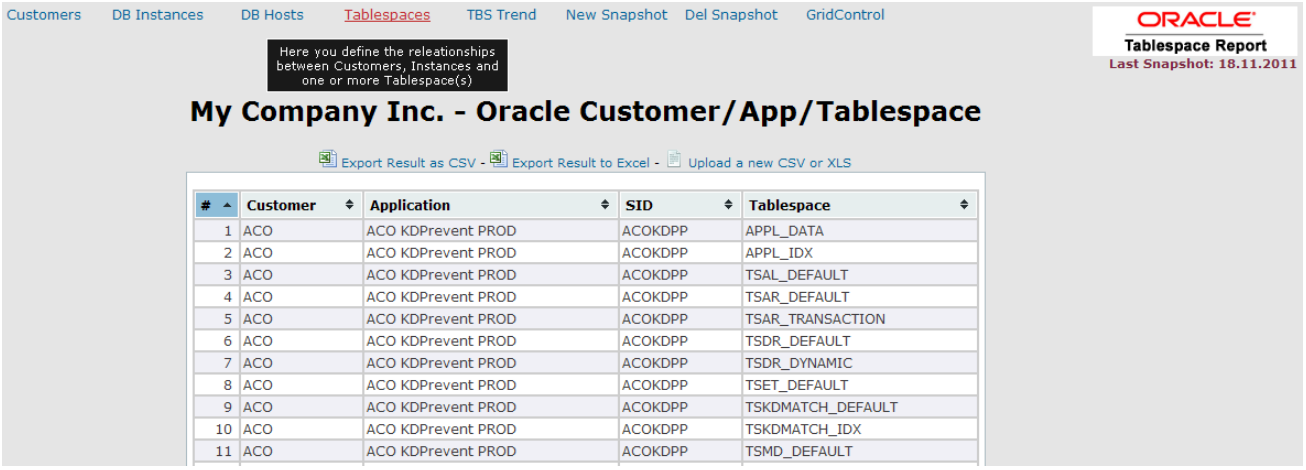


This is simply a list of which Instance is running on which physical host as of the latest snapshot, usually the automated Friday snapshot. A pdf file will be generated on a weekly basis to keep track of where an Instance once where in case of DB Instances has to get moved around and you have had some setup or maintain scripts stuffed away on the previous server.

Tablespaces

This is the heart of OTR. Here the connection between Customer, DB Instance and the Tablespaces are made.

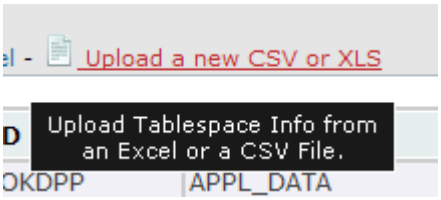
Tablespaces main screen



#	Customer	Application	SID	Tablespace
1	ACO	ACO KDPrevent PROD	ACOKDPP	APPL_DATA
2	ACO	ACO KDPrevent PROD	ACOKDPP	APPL_IDX
3	ACO	ACO KDPrevent PROD	ACOKDPP	TSAL_DEFAULT
4	ACO	ACO KDPrevent PROD	ACOKDPP	TSAR_DEFAULT
5	ACO	ACO KDPrevent PROD	ACOKDPP	TSAR_TRANSACTION
6	ACO	ACO KDPrevent PROD	ACOKDPP	TSDR_DEFAULT
7	ACO	ACO KDPrevent PROD	ACOKDPP	TSDR_DYNAMIC
8	ACO	ACO KDPrevent PROD	ACOKDPP	TSET_DEFAULT
9	ACO	ACO KDPrevent PROD	ACOKDPP	TSKDMATCH_DEFAULT
10	ACO	ACO KDPrevent PROD	ACOKDPP	TSKDMATCH_IDX
11	ACO	ACO KDPrevent PROD	ACOKDPP	TSMD_DEFAULT

The source for this information can be a .CSV file or an Excel Document. This file will be uploaded to the repository server (usually the same as the Grid Control server). In case of an Excel source the file will be converted into a .csv file and stored on a defined location where it will be used as source for an external table.

Upload CSV or XLS



The .CSV contains 4 fields/row and will have the following structure:

```
ICB;Amis CCR INT;AMCCRI;TSDATLARGE
ICB;Amis CCR INT;AMCCRI;TSDATNORM
ICB;Amis CCR INT;AMCCRI;TSDATSN
ICB;Amis CCR INT;AMCCRI;TSIDX
```

It contains the Customer ID, The Instance Description, OraSID and Tablespace name

The other possibility and also the easiest way, is to keep this info in an Excel sheet.

	A	B	C	D
1	ACO	ACO KDPrevent PROD	ACOKDPP	APPL_DATA
2	ACO	ACO KDPrevent PROD	ACOKDPP	APPL_IDX
3	ACO	ACO KDPrevent PROD	ACOKDPP	TSAL_DEFAULT
4	ACO	ACO KDPrevent PROD	ACOKDPP	TSAR_DEFAULT
5	ACO	ACO KDPrevent PROD	ACOKDPP	TSAR_TRANSACTION
6	ACO	ACO KDPrevent PROD	ACOKDPP	TSDR_DEFAULT
7	ACO	ACO KDPrevent PROD	ACOKDPP	TSDR_DYNAMIC
8	ACO	ACO KDPrevent PROD	ACOKDPP	TSET_DEFAULT
9	ACO	ACO KDPrevent PROD	ACOKDPP	TSKDMATCH_DEFAULT
10	ACO	ACO KDPrevent PROD	ACOKDPP	TSKDMATCH_IDX
11	ACO	ACO KDPrevent PROD	ACOKDPP	TSMD_DEFAULT
12	ACO	ACO KDPrevent PROD	ACOKDPP	TXAL_DEFAULT
13	ACO	ACO KDPrevent PROD	ACOKDPP	TXAR_DEFAULT
14	ACO	ACO KDPrevent PROD	ACOKDPP	TXAR_TRANSACTION
15	ACO	ACO KDPrevent PROD	ACOKDPP	TXDR_DEFAULT
16	ACO	ACO KDPrevent PROD	ACOKDPP	TXDR_DYNAMIC
17	ACO	ACO KDPrevent PROD	ACOKDPP	TXET_DEFAULT
18	ACO	ACO KDPrevent PROD	ACOKDPP	TXMD_DEFAULT
19	ICB	Amis CCR INT	AMCCRI	TSDATLARGE
20	ICB	Amis CCR INT	AMCCRI	TSDATNORM
21	ICB	Amis CCR INT	AMCCRI	TSDATSN
22	ICB	Amis CCR INT	AMCCRI	TSIDX
23	ICB	Amis CCR PROD	AMCCRP	TSDATLARGE
24	ICB	Amis CCR PROD	AMCCRP	TSDATNORM
25	ICB	Amis CCR PROD	AMCCRP	TSDATSN
26	ICB	Amis CCR PROD	AMCCRP	TSIDX

The content in Excel is the same as for the .csv

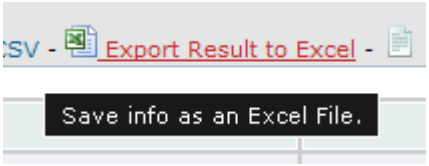
Company ID, Instance description, OraSID and Tablespace name.

Export as CSV

This info can also be exported locally as either a .csv file

Export as XLS

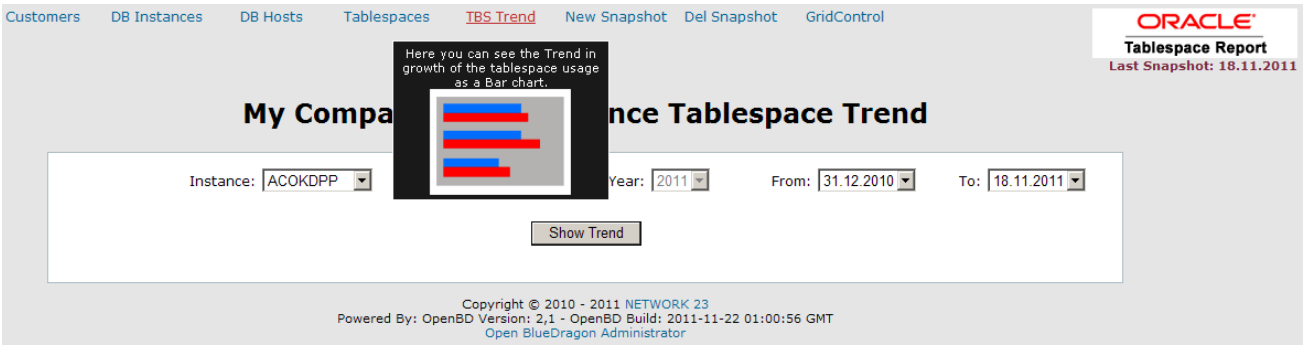
Or as an Excel document



TBS Trend

This will display the trend of growth graphically in a Bar chart.

Main screen for Tablespace Usage Trends



Trend defined by 2 snapshots

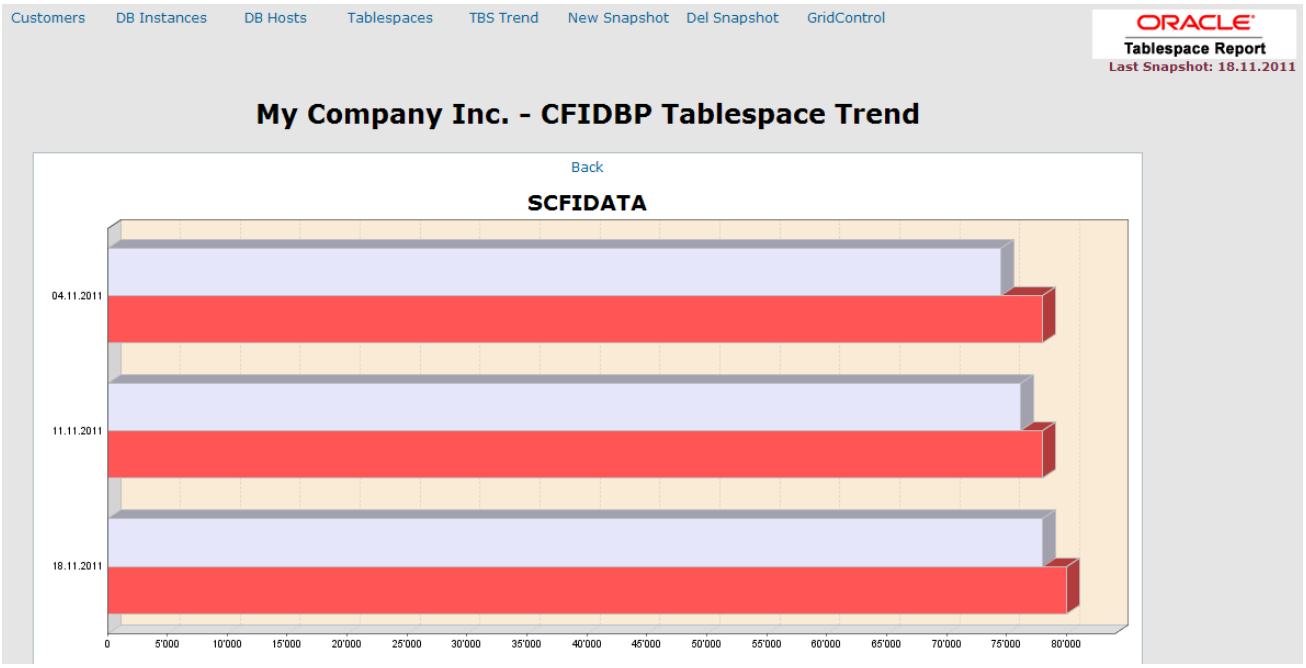
Statistical data can be displayed from a time period between 2 snapshots.

Trend defined on a Monthly basis

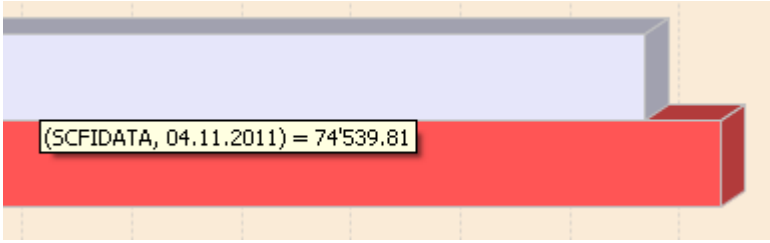
This will pick the last snapshot from each month within the selected year.

Graphical Output

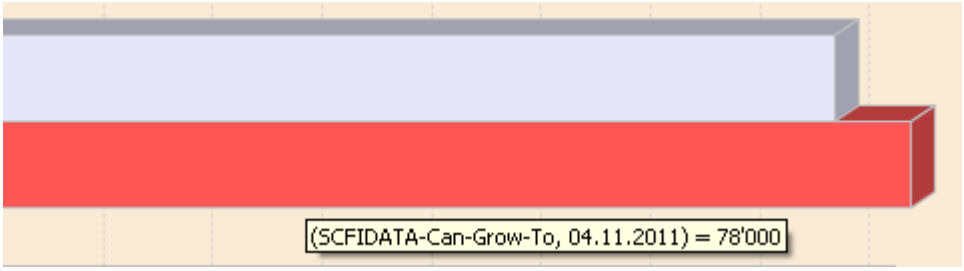
The output will display the output as a bar-chart.



The currently used space in the tablespace…



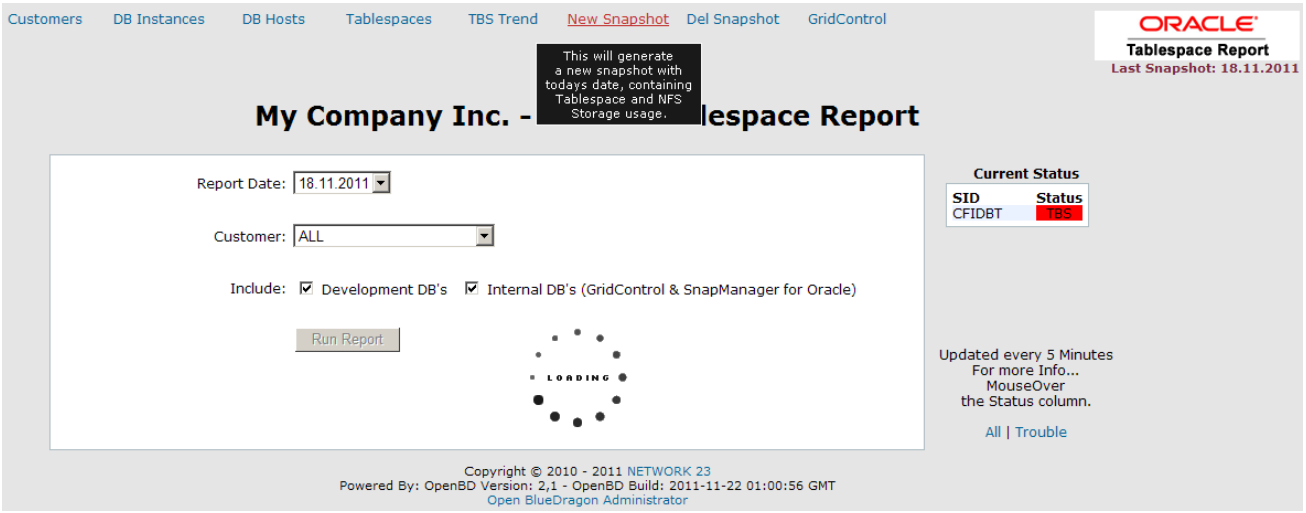
and the size the tablespace can grow to.



Snapshots

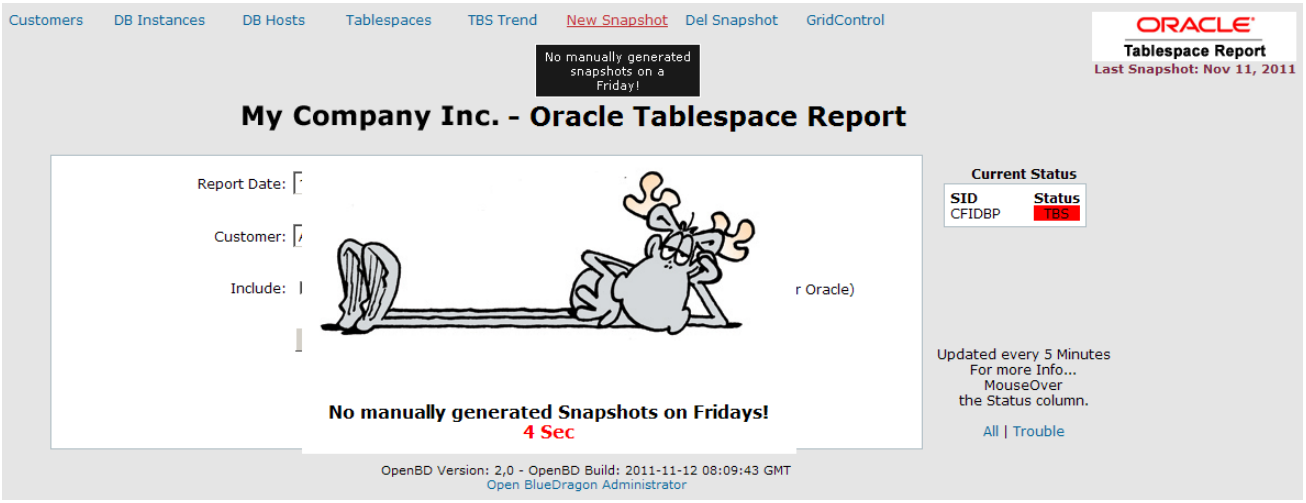
The statistics is stored in the OTR repository as a snapshot. This is done as a weekly Scheduled job defined in the OpenBD Administrator. This job should be scheduled shortly before Friday Midnight.

Snapshots can also be generated manually.

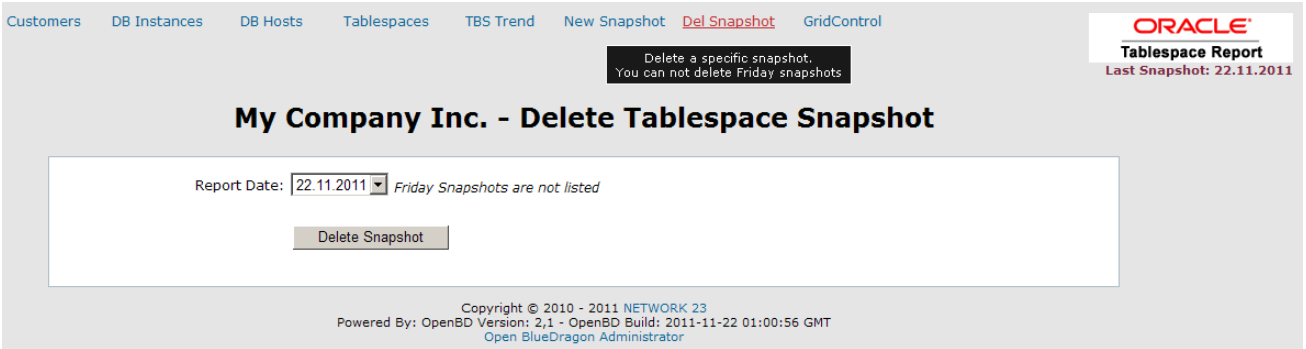


Only one snapshot / day will be stored, so creating a new snapshot again on the same day will simply delete the previous one and create a new snapshot for that day.

No Friday snapshots!



Since Fridays are our scheduled snapshot day you are not allowed to create manual snapshots on this day. It's possible to delete manually generated snapshots but not the Friday snapshots.



Grid Control

Since we're DBA's we of course need access to our "real" toolbox. Therefor we have a direct link to the login for Oracle® Grid Control.

How do we get started?

First of all we need to setup the repository OTR schema and the objects used for the repository, assuming of course that Grid Control is already installed!

DDL Scripts

These scripts are located under DOC\OTR-Reporting\Setup\DDL

```
OTR_DB_SPACE_REP_SCHEMA.sql
OTR_DB_SPACE_REP_DDL.sql
OTR_CR_VIEW_TBS_FREE.sql
OTR_CR_VIEW_DB_HOST.sql
OTR_DB_SPACE_REP_DROP_DDL.sql
OTR_DB_SPACE_REP_TBS+SCHEMA_CLIENT.sql
```

OTR_DB_SPACE_REP_SCHEMA.sql

The first script to run is the OTR_DB_SPACE_REP_SCHEMA.sql which will create new tablespace(s) for the OTR Repository, Create the schema owner OTRREP and setup the grants needed.

To avoid any license problems make sure not to use the Enterprise Manager Repository Instance for your OTR Repository!

```
>@OTR_DB_SPACE_REP_SCHEMA.sql
Enter Database Alias for the OTR Repository [OTR]: SMO
Enter Password for user SYS: ********
Enter path for the otr_rep_data01.dbf [/u01/oradata/otr_db/OTR]: /u01/oradata/smo_db/SMO
Enter path for the otr_rep_indx01.dbf [/u01/oradata/otr_db/OTR]: /u01/oradata/smo_db/SMO
OTR Datafiles will be placed under
/u01/oradata/smo_db/SMO/otr_rep_data01.dbf
/u01/oradata/smo_db/SMO/otr_rep_indx01.dbf
If this is correct press Enter otherwise Ctrl+C
```

OTR_DB_SPACE_REP_DDL.sql

Next script to run is the OTR_DB_SPACE_REP_DDL.sql

```
>@OTR_DB_SPACE_REP_DDL.sql
Enter Database Alias for the OTR Repository [OTR]: SMO
Enter Password for user SYS: ********
Enter Password for user OTRREP: **********
NOTICE: Path for the External table must be LOCALLY on the OGC DB Server (no UNC path allowed!)
Path for the External Table [/orascripts/scripts/monitoring/xt/OTR]: /orascripts/scripts/monitoring/xt/OTR
Enter SQLNET.DEFAULT_DOMAIN [MBCZH.CH]: MBCZH.CH
```

This script will create a directory object for our external table, all other tables used to store the repository data.

This script will also call the 2 scripts OTR_CR_VIEW_TBS_FREE.sql and OTR_CR_VIEW_DB_HOST.sql.

OTR_DB_SPACE_REP_TBS+SCHEMA_CLIENT.sql

This step is optional OTR_DB_SPACE_REP_TBS+SCHEMA_CLIENT.sql. Installing an OTRREP Schema in your target DB's is not needed. This schema will only contain a few views used to display tablespace usage on the current instance. OTR will pick this info up without this schema!

```
>@OTR_DB_SPACE_REP_TBS+SCHEMA_CLIENT.sql
Enter Database Alias for the Target DB (TNSNAMES): HYPCCRP

Choose the Permanent (USER) tablespace for the OTRREP user
----------------------------------------------------------
Below is the list of online tablespaces in this database which can
can be used for storing data and objects.
Tablespace marked with a * is the default permanent tablespace.
selecting the SYSTEM tablespace as tablespace for OTRREP when
there is an USERS tablespace available don't make sence!!!
Select the OTRREP user's Standard tablespace.

TABLESPACE_NAME                CONTENTS  DB DEFAULT USER TABLESPACE
------------------------------ --------- --------------------------
DATA_ASSENTISDEF               PERMANENT
DATA_ASSENTISRT                PERMANENT
FFPMBCDATA                     PERMANENT
FFPMBCINDX                     PERMANENT
FFPZGKDATA                     PERMANENT
FFPZGKINDX                     PERMANENT
HYPICBDATA                     PERMANENT
HYPICBINDX                     PERMANENT
IDX_ASSENTISDEF                PERMANENT
IDX_ASSENTISRT                 PERMANENT
INDX                           PERMANENT
SYSAUX                         PERMANENT
SYSTEM                         PERMANENT
USERS                          PERMANENT *

Pressing <return> will result in the database's default Permanent
tablespace (identified by *) being used.

Enter USER TABLESPACE Name: USERS

Choose the OTRREP user's Temporary tablespace.

TABLESPACE_NAME                CONTENTS  DB DEFAULT TEMP TABLESPACE
------------------------------ --------- --------------------------
TEMP                           TEMPORARY *

Pressing <return> will result in the database's default Temporary
tablespace (identified by *) being used.

Enter Temporary TABLESPACE Name: TEMP


... Creating OTRREP user
```

Entering SYSTEM or SYSAUX as Temporary Tablespace will generate an Error and the script stops.

If you have consistent standard within your company you could simple setup a job in Grid Control for all targets. Copy and Paste the scripts or reference them as external scripts.

In the case of a copy and paste direct into Grid Control SQL scripts job, make sure to check the "WHERE" clauses where a '%' sign is used. Grid Control is using % for internal variables so you must change the pasted script to use '%%'.

Done with Step 1

This rounds up the first part and we have to download the WEB GUI and the OpenBD Server.

Web Frontend using Open BlueDragon

For the Web frontend of OTR we need the server software from the OpenBD project. OpenBD is the world's first truly open source GPL Java and Google App Engine CFML runtime. CFML is a powerful tag/script based language that takes away all the heavy lifting of producing highly scalable web and email based services and sites.

Download Ready2Run Jetty+OpenBD

At http://www.openbd.ord/download we need to download the Ready2Run Jetty+OpenBD. It's also possible to run OpenBD on a Tomcat Server. In this case we only need to download the openbd.war file and dump it in the webapps folder. Tomcat setup is not described in this document.

In this case we will use the OMS/EM Server to install the OTR Web GUI.

If we have internet access direct from the OTR Server and this is a Linux/UNIX Server we can use the wget command.

We will install the OpenBD under /opt/OpenBD

As user root

```
# cd opt
# mkdir OpenBD
# cd OpenBD
# wget http://www.openbd.org/download/2.0.1/jetty-openbd.zip
# unzip jetty-openbd.zip
```

Download JDK or JRE 6 from Oracle

At http://www.oracle.com/technetwork/java/javase/downloads/index.html

At the time of this writing the release is Java SE 6 Update 29. Select the appropriate release for your platform.

In our case we're on a 64-bit Oracle Linux so our download would be jdk-6u29-linux-x64-rpm.bin

Installing this with

```
# ./jdk-6u29-linus-x64-rpm.bin
```

Making this as our default Java setup we will use the "alternatives" to maintaining symbolic links to our newly installed java.

```
# /usr/sbin/alternatives --install /usr/bin/java java /usr/java/jdk1.6.0_29/bin/java 16029
# /usr/sbin/alternatives --display java
```

Fixing the start script for OpenBD

Create a file /etc/default/jetty to define the JETTY_HOME

```
# vi /etc/default/jetty

JETTY_HOME=/opt/OpenBD
```

Change the mod of the start/stop script

```
# chmod 755 /opt/OpenBD/bin/jetty.sh
```

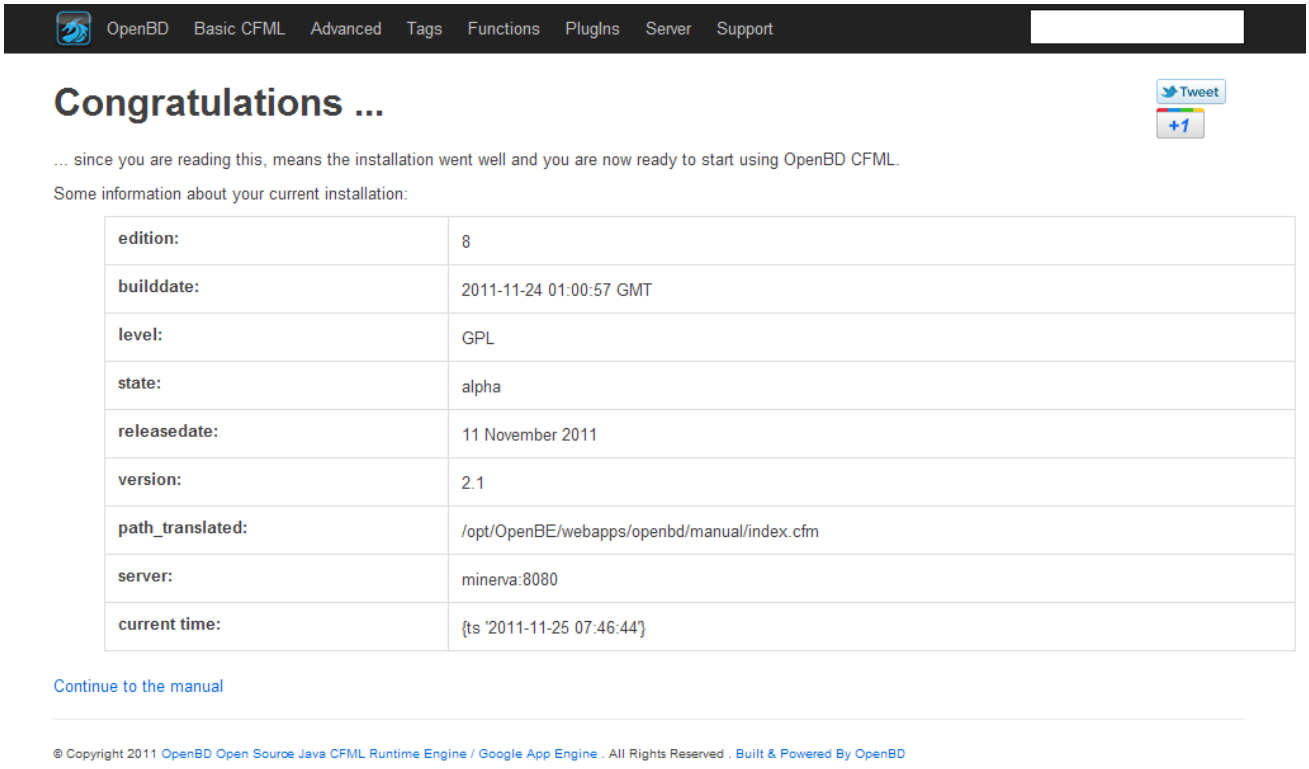
And start the Jetty/OpenBD

```
# bin/jetty.sh start
```

Test your OpenBD Installation

Open a web browser and goto the URL http:// your_ogc_otr_server:8080

You should get a message that your Installation was successful.



By default Jetty, like most every Java Server, is configured to use port 8080. You can easily change this to port 80 since most likely you will not have any conflict using this port. Grid Control usually, with a normal installation, will use a different port (typically 4889)

Change listener port for Jetty

If you prefer to use port 80 instead of port 8080 simple stop the OpenBD Server again.

```
# bin/jetty.sh stop
```

Go into the etc folder

```
# cd /opt/OpenBD/etc
```

Edit the file jetty.xml

```
# vi jetty.xml
```

Look for the property jetty.port, change the default=”8080” to default=”80”

Also change the file jetty-fileserver.xml

```
# vi jetty-fileserver.xml
```

Look for the `<Set name=”port”>8080</Set>` and change this to

```
<Set name=”port”>80</Set>
```

Start the OpenBD Server again

```
# cd ..
# bin/jetty.sh start
```

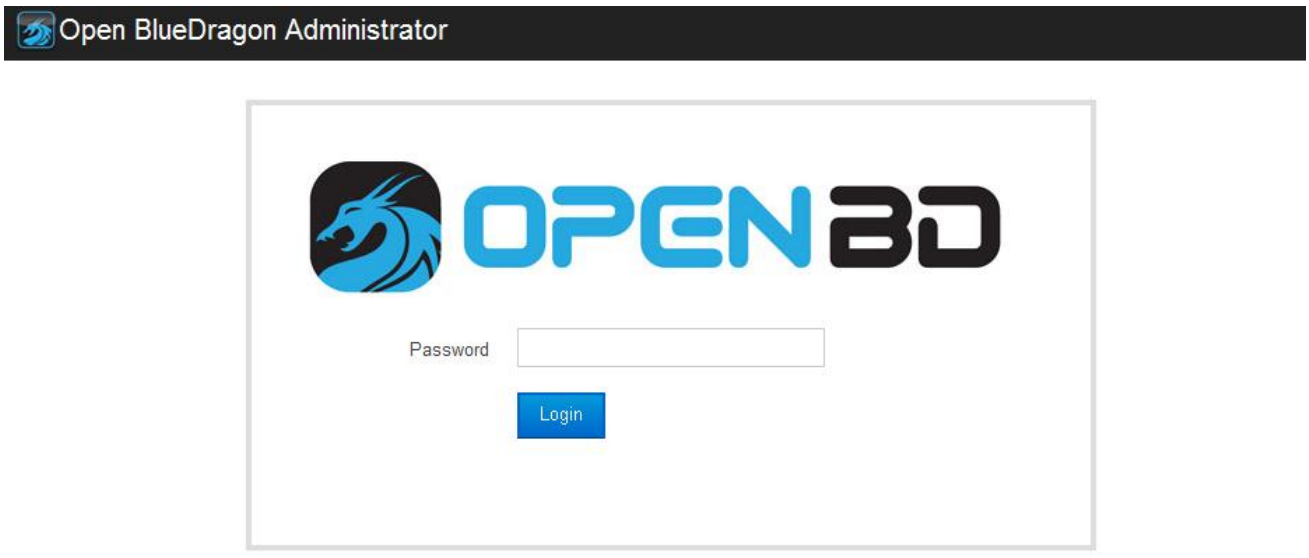
From now on your OpenBD should respond on standard port 80

Configuring OpenBD for OTR

First we need to define 2 Datasources for OTR to be able to communicate with the OGC and OTR Repositories.

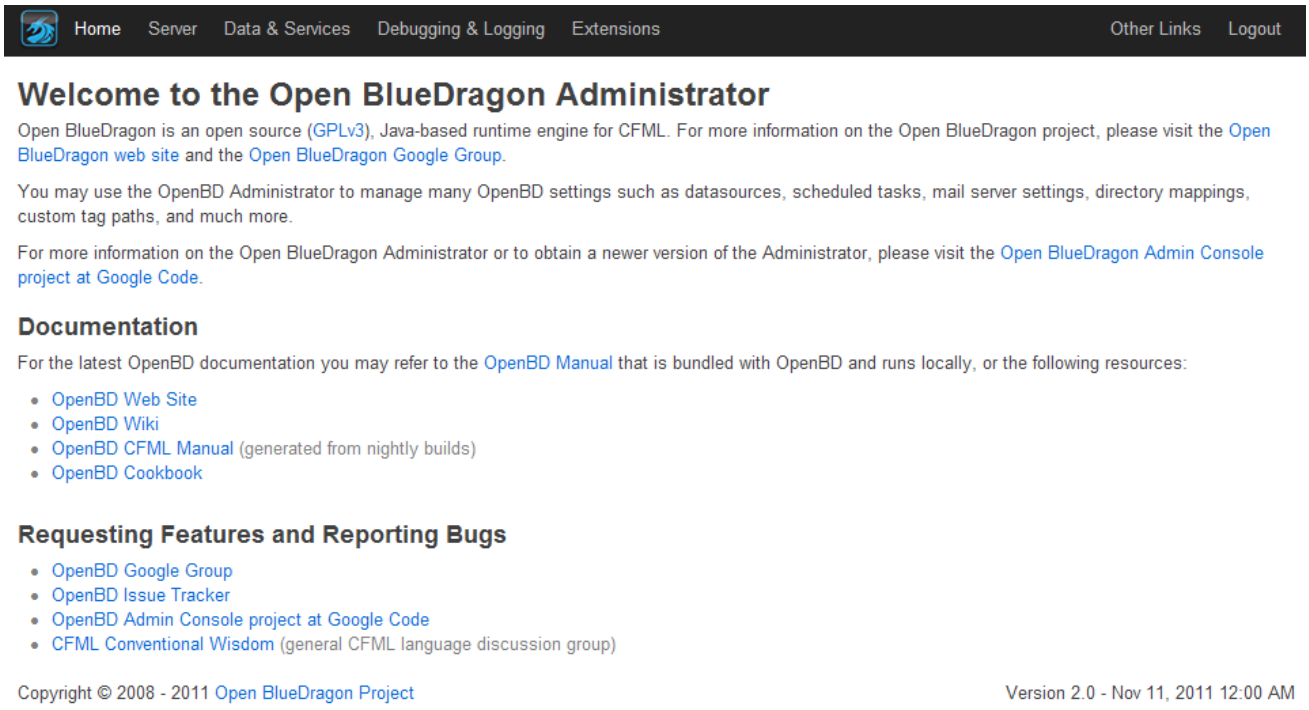
Login to OpenBD Administrator

Go to the URL, with or without the portnumer depending on if you reconfigured your Jetty Setup or not, http:// your_ogc_otr_server[:8080]/bluedragon/administrator

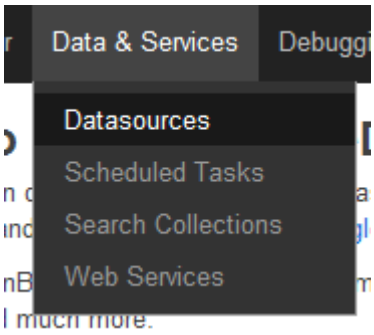


The default Password is **admin**

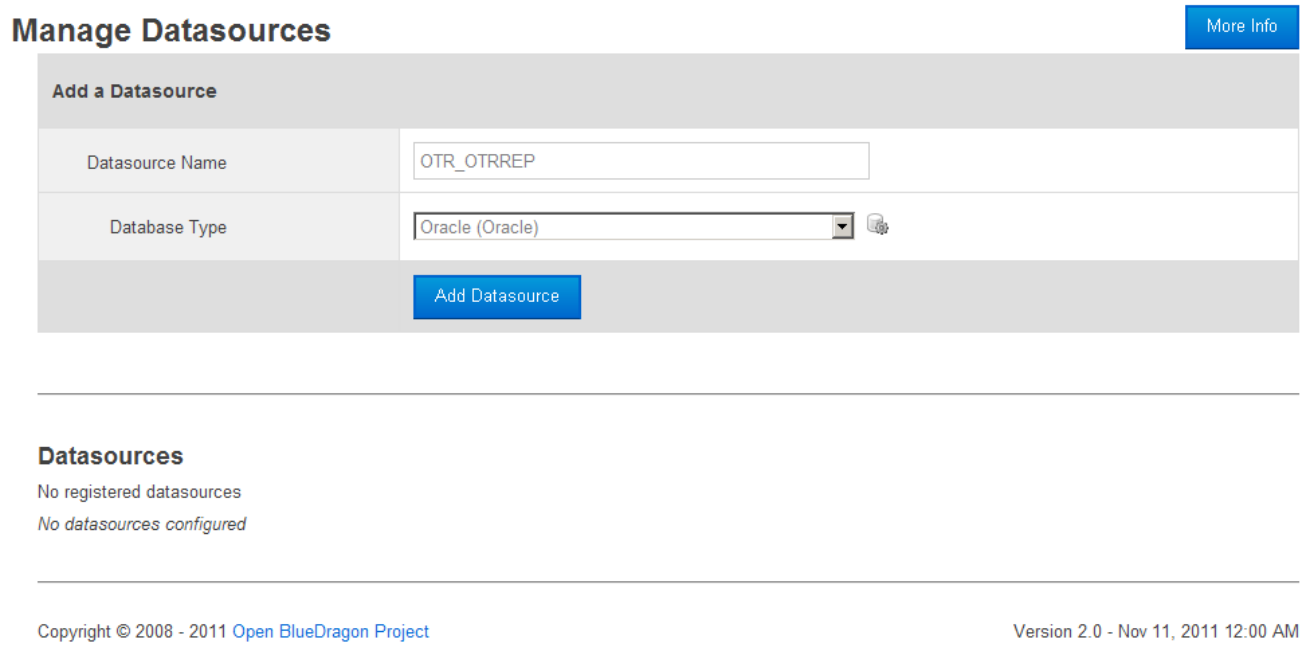
OpenBD Administrator Main Screen



To add new Datasources select the menu Data & Services

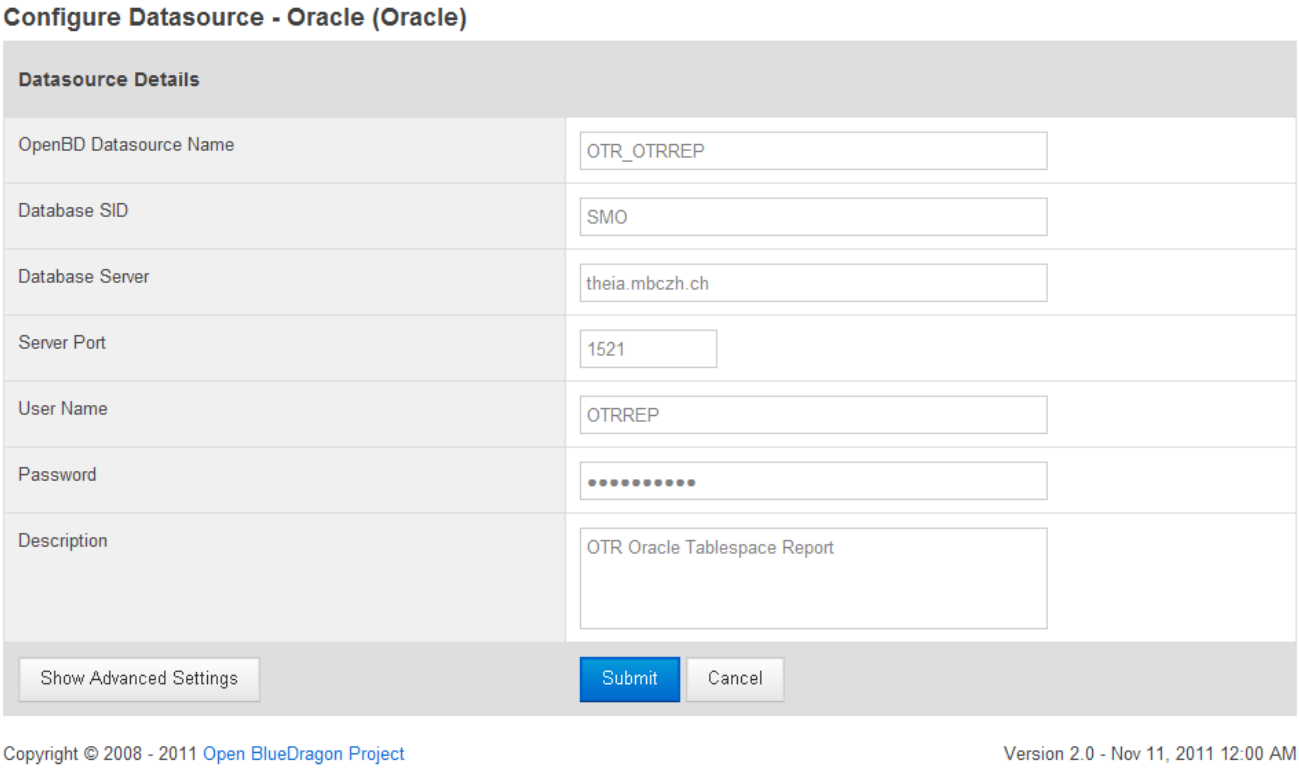


Add Datasource OTR_OTRREP

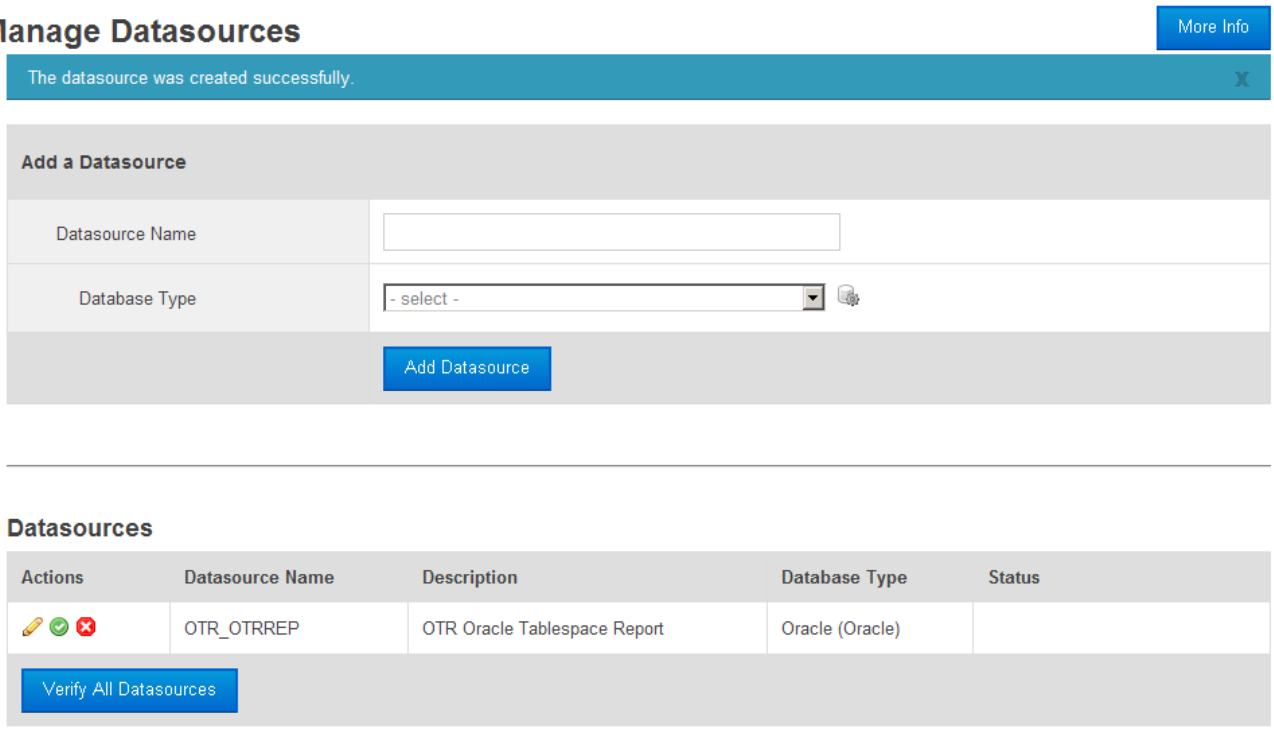


Datasource Name: **OTR_OTRREP** and Type is of course **Oracle** and click **Add Datasource**

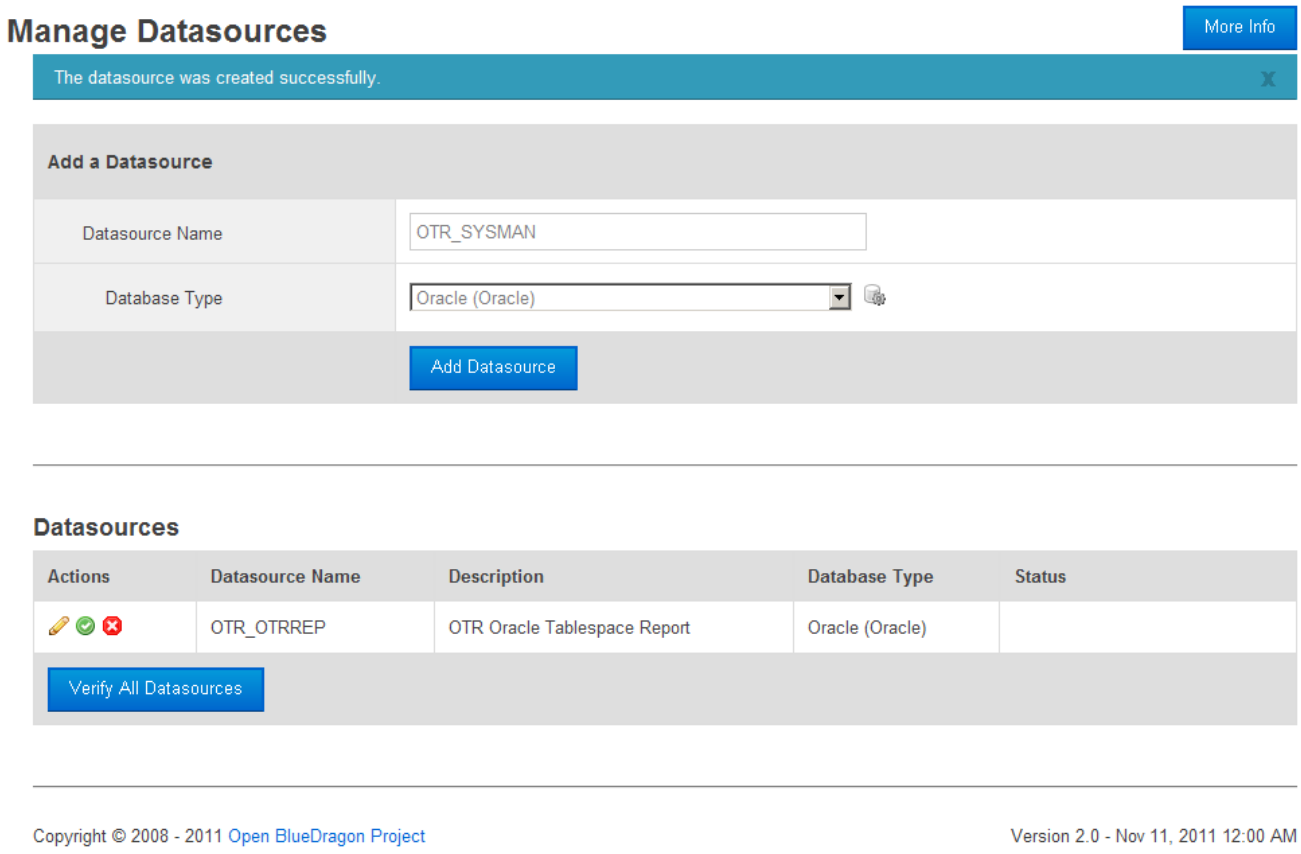
Adding connection info



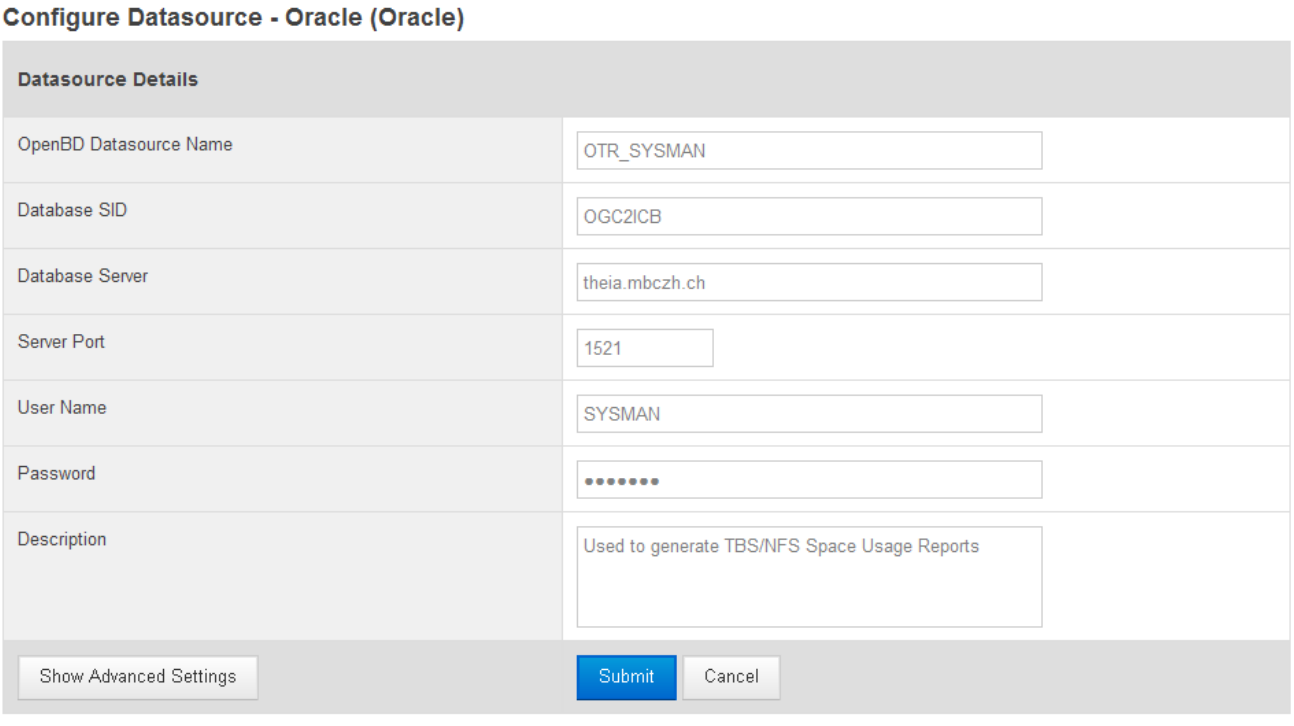
Database SID: **<Your OTR OracleSID>**
Database Server: **<Host of your OTR Instance>**
Server Port: **<Listener Port for your OTR Instance>**
User Name: **OTRREP**
Password: **otrrep4otr**
Description: **OTR Oracle Tablespace Report**



Add Datsource OTR_SYSMAN



Datasource Name: **OTR_SYSMAN** and the type **Oracle** and click **Add Datasource**

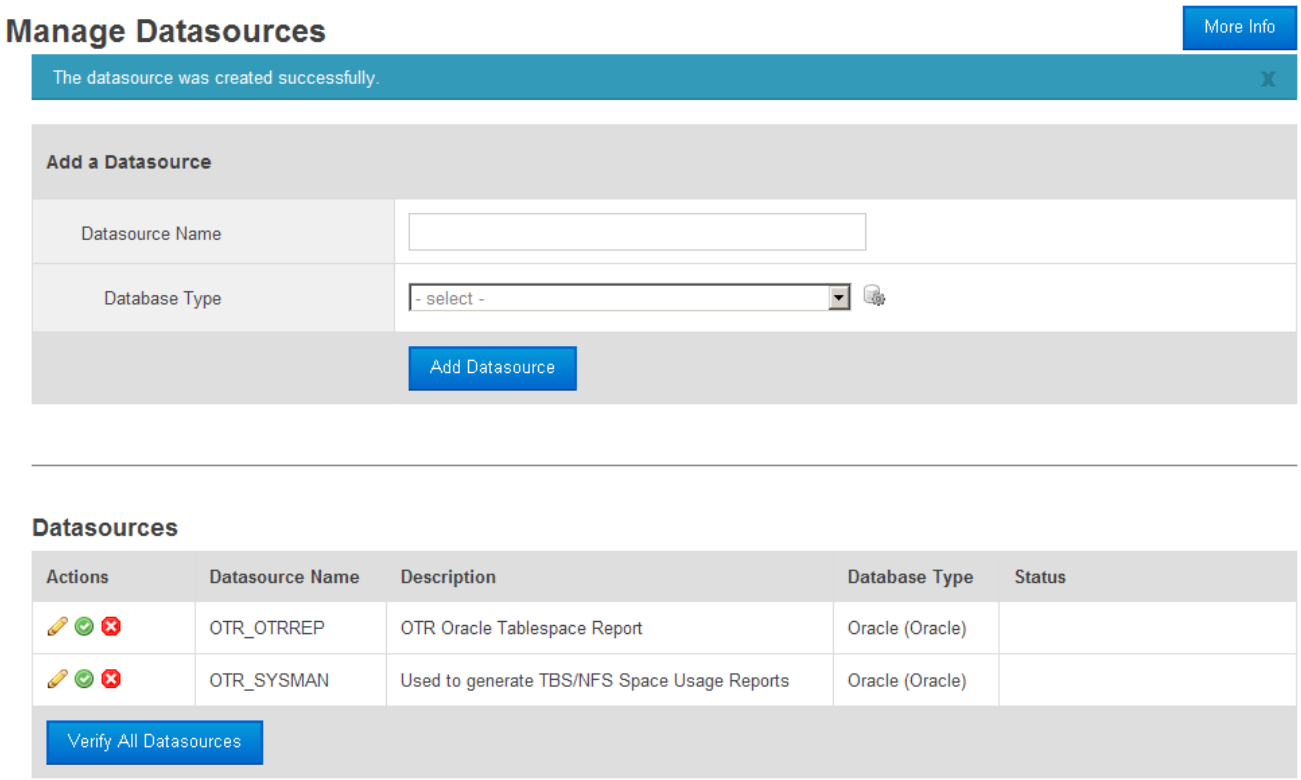


User Name: **SYSMAN** <User on your OGC Instance!!!>

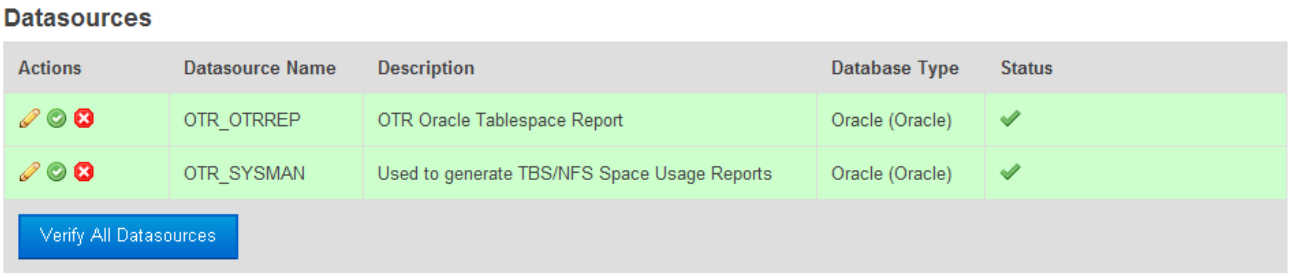
Password: **<SYSMAN Password>**

Description: **Used to generate TBS/NFS Space Usage Reports**

Test the new Datasources



By clicking on **Verify All Datasources** you will get a confirmation of the settings and if they are OK



Installing the OTR WebApp

In this section we will install the web application for OTR.

Download the otr.war

The otr.war can be downloaded from http://www.network23.net/otr or get the complete source from Google Code http://....

Copy the otr.war over to your server

Simply copy the otr.war to your server under /opt/OpenBD/webapps/openbd using WinSCP

Most likely is the Linux/UNIX server, where the Grid Control's EM and/or OMS is running, not open for remote access for user root. So transfer the file into /tmp as user oracle. SSH connect to the server as user oracle and then with su – change to root. Move the file from /tmp to /opt/OpenBD.

Jetty is not default configured for hot deploy so simply use unzip to unpack the content

```
# unzip otr.war
```

Change the mod for the directory and its content so you, for future needs, can access the OTR application from outside the server with user Oracle

```
# chmod –R 777 otr
```

Make changes to the file Application.cfc to fit your Company and setup

Most all parameters for the OTR Application is defined in the file otr/Application.cfc

Update the following settings.

Oracle Settings

```
<!--- SQLNET.DEFAULT_DOMAIN for DB-Links --->
<cfset Application.oracle.domain_name = "MBCZH.CH" />
<!--- Datasource Settings --->
<cfset Application.datasource = "OTR_OTRREP" />
<cfset Application.dbusername = "OTRREP" />
<cfset Application.dbpassword = "otrrep4otr" />
```

The **Application.oracle.domain_name** should correspond to the SQLNET.DEFAUL_DOMAIN within your Oracle environment.

If you change the password for the Schema Owner OTRREP it needs to be changed here also.

Company Settings

```
<!--- Company Settings --->
<cfset Application.company = "My Company Inc." />
<!--- Excel Document Info --->
<!--- Foreign Characters for Excel
      ß = chr(223)
      å = chr(229)
      ä = chr(228)
      ö = chr(246)
      Å = chr(197)
      Ä = chr(196)
      Ö = chr(214) --->
<cfset Application.excel_doc_info_author = "Mats Str#chr(246)#mberg" />
<cfset Application.excel_doc_info_subject = "Customer Tablspace Usage" />
<cfset Application.excel_doc_info_title = "My Company Inc. - Tablespace Report" />
<cfset Application.excel_doc_info_lastauthor = "ustr" />
```

Application.company is Your Company Name. This will be displayed on every screen in the application.

Application.excel_doc_info_xxx will be used as document info when generating Excel files. Some character values are provided for foreign character which Excel will understand.

Snapshots

```
<!--- Snapshot Day / Sunday = 1 --->
<cfset Application.snapshot_day = 6 /><!--- 6 = Friday --->
```

It's possible to change the snapshot day but it's not really recommend.

The week starts on Sunday = 1 and stops on Saturday = 7

General Settings

```
<!--- General Application Settings --->
<cfset Application.obd_host = "http://minerva/" />
<cfset Application.obd_desktop_host = "http://localhost/" />

<cfset Application.ogc_logon_url = "http://minerva:4889/em/console/logon/logon" />
<cfset Application.ogc_external_table = "/orascripts/scripts/monitoring/xt/OTR" />
<cfset Application.host_instance_pdf_dir = "/opt/pro/dir/ccr/oracle/" />
```

Application.obd_host is the host of the OTR web server. If Jetty isn't re-configured for port 80 this should contain the correct port number. **http://YourServer:8080/**

Application.ogc_login_url is the URL for your Grid Control login screen.

Application.ogc_external_table is the path where your external table source is located. This file contains info about your Customers, Instances and Tablespaces. (See chapter ***Tablespaces***)

Application.host_instance_pdf_dir is the location where the weekly PDF reports will be located.

SFTP Settings

```
<!--- SFTP Settings for the host of the GridControl Repository --->
<cfset Application.sftpHost = "minerva.mbczh.ch">
<cfset Application.sftpUser = "oracle">
<cfset Application.sftpPass = "orambc">
```

The SFTP setting is important for the setup and updates of the External Table (Customer - Instance – Tablespace) relation. This file is located somewhere locally or on a NFS mountpoint where the server for the OTR Repository is installed. This doesn't have to be, and most likely won't be, the same server where the Web GUI for OTR is installed.

Application.sftpHost is name of the server where the access to the external table source file is located.

Application.sftpUser is normally oracle since your Oracle Instance most likely is running under the account oracle.

Application.sftpPass is your Oracle OS user password.

With these settings correctly configured for your environment you are now ready to do the final setup of OTR.

Restart Jetty/OpenBD

The values contained in the Application.cfc will only be activated at the time the OTR application is started. Changes done during the time the OpenBD/Jetty is running will not be visible.

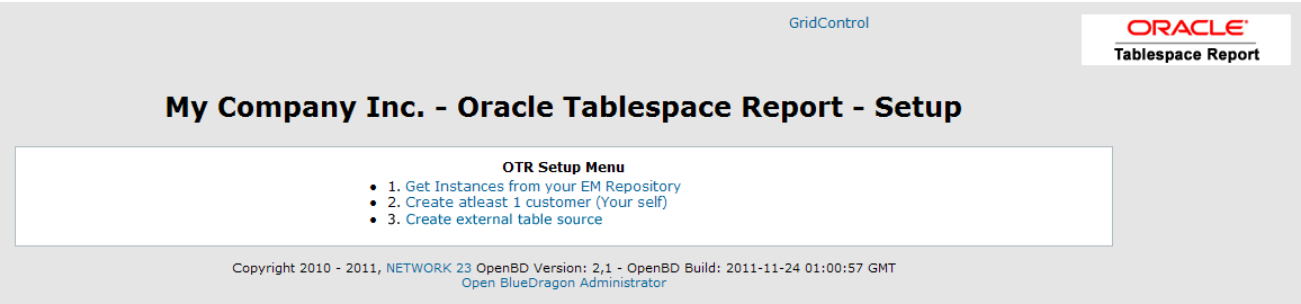
So to get this change you have just made, you simply stop and start Jetty again. (Make sure you're user root for this)

```
$ su -
# cd /opt/OpenBD
# bin/jetty.sh restart
```

Now when you go to the URL http://yourserver[:8080]/otr you should see the Setup screen now, displaying your Company Name.

Get your OTR ready for use

To get your OTR read to be used we now need to fill it with usable data.



Get Instances from EM Repository

Since this is a new setup you won't have any database Instances in OTR. Start by selecting the menu 1. Get Instances from your EM Repository. When this step is done the link will be inactive.

Create your first Customer

There are no customers in your OTR Repository. Select the menu 2. Create at least 1 customer (Your self). When this step is done the link will be inactive and the Main screen of OTR will be displayed.

Create the Tablespace relationships

The relation between a Customer, DB Instance and a Tablespace is preferably done using an Excel sheet. As this is a new setup your external table source file doesn't exists yet.

By selecting the menu 3. Create the external table source and the system will create a new file for you that will contain all your instances and the description. Customer ID will be set to the Customer ID of the customer you just created and Tablespace will be set to the value <NOT_DEFINED>.

Export this as XLS and edit this file locally on your PC.

Copy each row for as many Tablespaces you want to monitor for each Instance and customer. Replace the XYZ with correct Customer ID and save the Excel file. Finally upload the file again to OTR and your Tablespace list will now be usable.

Add all your customers

Make sure to add all customers to your system and that the Customer ID is corresponding to your updated Excel file / External table source file. This will be needed when you create your first snapshot, may it be a manually created snapshot or the weekly generated snapshot.

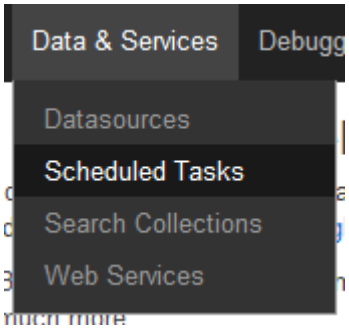
Define a Gather TBS/NFS Space Usage Statistics Job

To get the weekly monitoring to collect the Tablespace usage statistics you should now define a Job in the OpenBD Administrator.

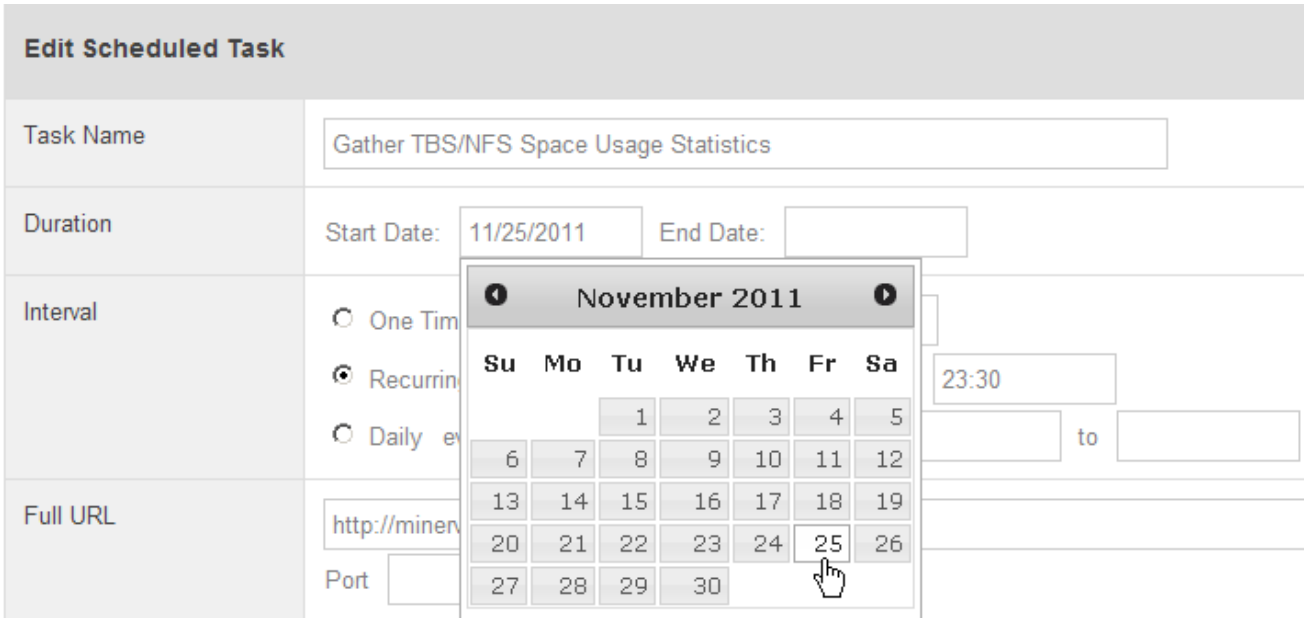
Login to the Administrator using the URL http://your_server[:port]/bluedragon/administrator/

If you haven't changed the Administrator Password it will be **admin**.

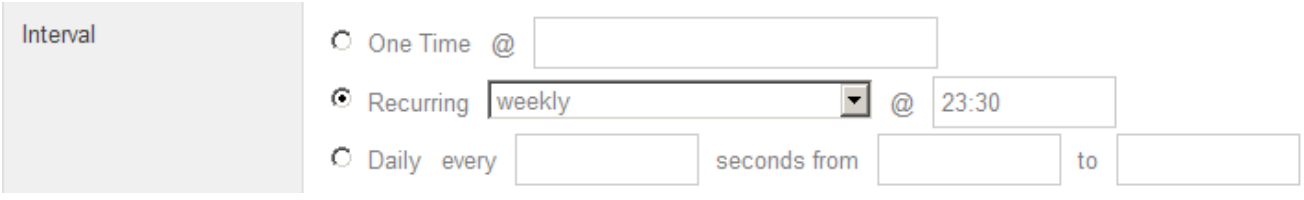
Select the Menu Data & Services and the Scheduled Tasks



Enter **Gather TBS/NFS Space Usage Statistics** as Task Name.

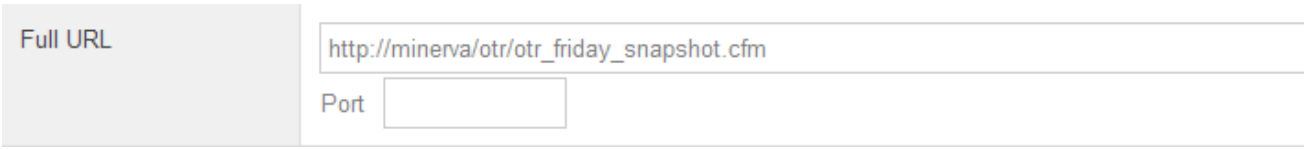


The Job should be run on a Friday night so select a Date matching a Friday.

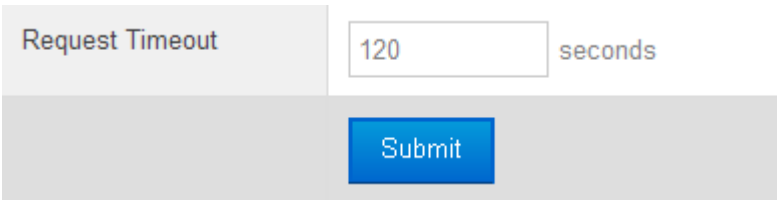


Define the job as a weekly recurring job starting at **23:30** (11:30 PM)

Enter **http://your_server[:8080]/otr/otr_friday_snapshot.cfm** as Full URL



As request Timeout set the value to **120** seconds.



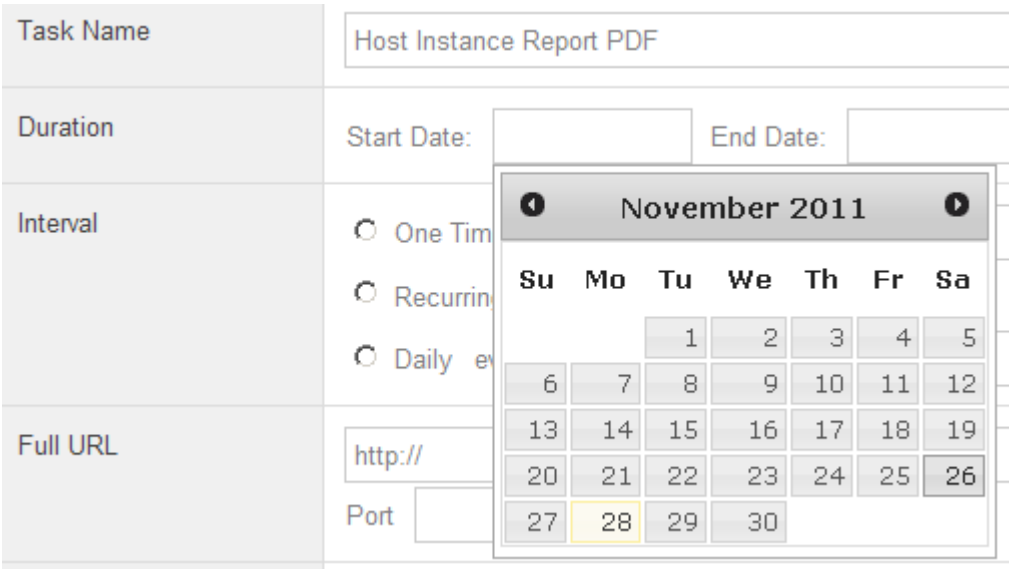
Define Job for creating Host/Instance PDF.

If you could like to have a weekly PDF generated containing info about which Instance is running on which Host, you can create this job over the OpenBD Administrator. The job should be generated on a weekly basis just as the Gather Statistics Job and be run just before or after Friday midnight, shortly after the Gather Job in Grid Control.

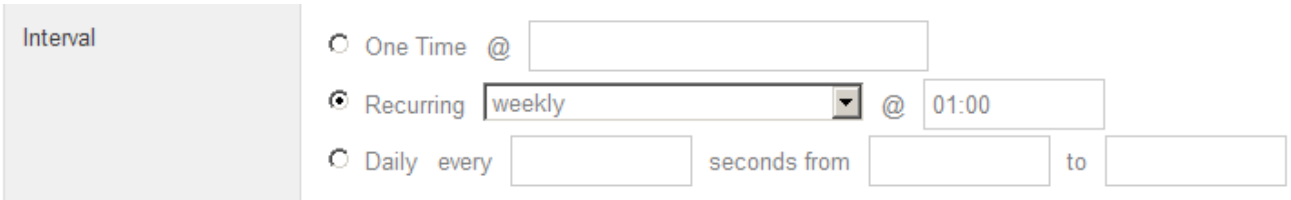
This step is not required but the information could be handy to have later on…

Define the Scheduling Task

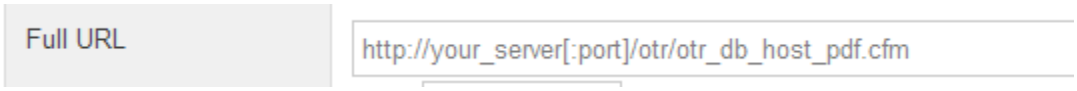
Task Name: **Host Instance Report PDF**. We'll define the Start Date to be on a Saturday.



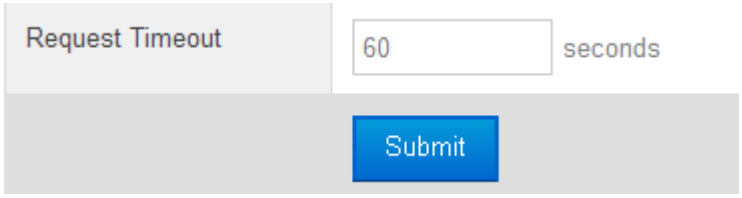
Set the Interval to **Weekly** at **01:00** which means that the job will be run on Saturday morning at 01:00 AM



The URL is: **http://your_server[:*port*]/otr/otr_db_host_pdf.cfm**



Set the Request Timeout to **60** sec and click the Submit button.

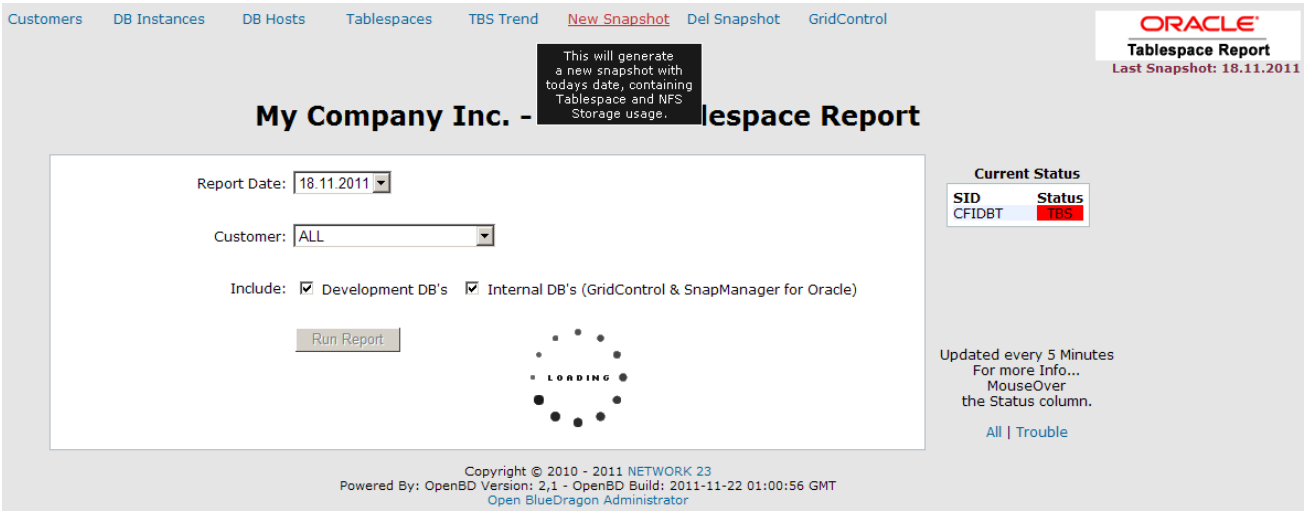


Your job is now defined. You could test the job but since we don't have any statistical data collected yet it won't generate any PDF.

Test your Setup.

As long as you're not testing your setup on a Friday you could now create your first Manual Snapshot. Required is of course that the relationship between Customer/Instance and Tablespace is done so the snapshot will have something to collect.

In the Web GUI of OTR select the menu option New Snapshot.

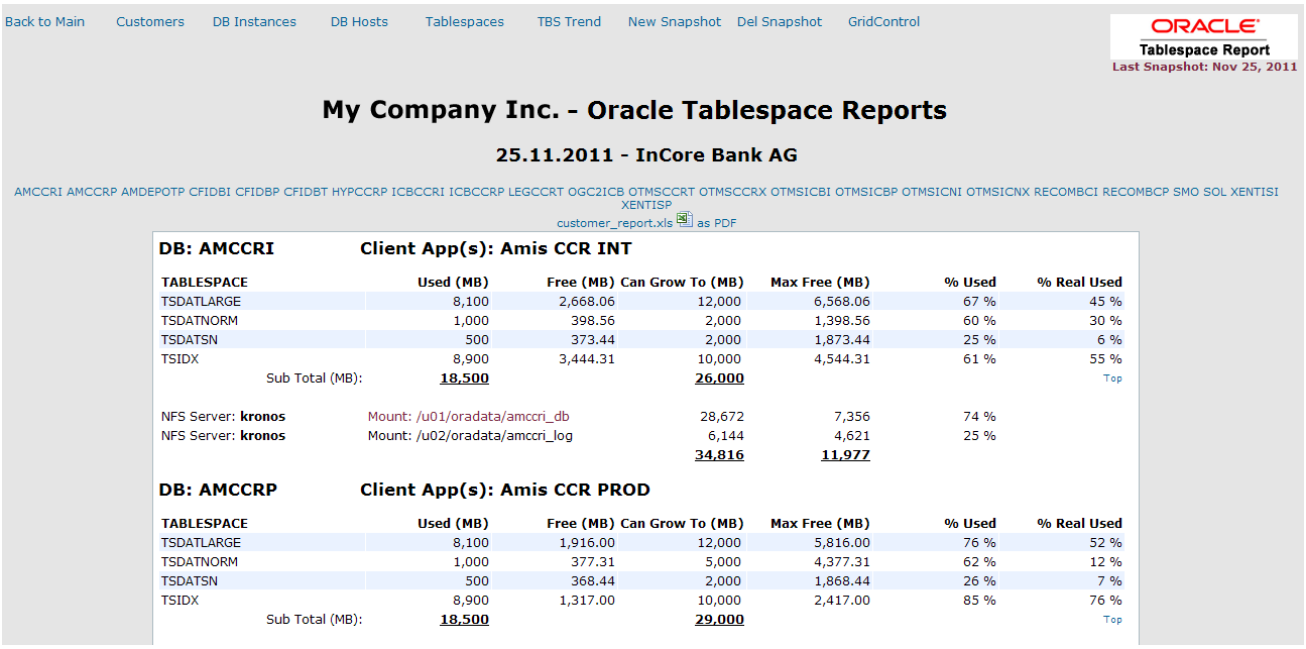


Note: If no snapshots has been made you can't generate any usage reports!!!

If at least 1 Snapshot exists it's possible to run a Report.

First select the Report Date and for which Customer. It can be for All customers or for 1 specific customer. Also select to include (or not) Development DB's and/or Internal DB's like the Grid Control or some other internal type of DB (SMO, RMAN etc.)

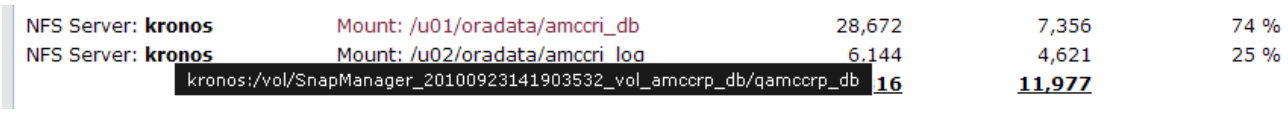
The report output will contain info about Instance, Tablespace name, Used MB, Free MB, Can Grow To MB, Max Free MB, % Used and % Real used which reflects the Can grow to space.



The report also contains NFS space usage in MB. How much space an NFS Volume has and how much free space is still available in MB. It also displays which NFS Server or Storage system is used.

One special feature for volumes created with NetApp's SnapManager for Oracle. The names of these volumes are usually not following your regular volume definition. If such a volume is used, it will be displayed with a dark red color. With a mouse-over on such a Mount name the real name of the volume will be displayed.

For example:



This concludes the description of the basic Setup and usage of OTR.

Feel free to add functionality to OTR. Get the source code from Google code and join in on the development.