

Draft

Schemation DEM Administration Console User Guide

A detailed user guide on how
to create and manage your
Database Environments using the
DEM's Administration Console.

Author: Simon Hewins

Release: 0.1

Contents

1	Introduction	1
1.1	What is Schemation DEM	1
1.2	Prerequisites.....	1
1.3	Section Overview.....	2
1.4	Getting Started / Logging into DEM Admin	2
2	Tablespace Admin	3
2.1	Overview.....	3
2.2	Create a Tablespace Mapping	3
2.2.1	Create a Tablespace Group.....	3
2.2.2	Create a Tablespace Group Member.....	4
3	User Admin	6
3.1	What is a Schemation User	6
3.2	Create Users	6
3.3	Register Existing User	7
4	Project Admin	9
4.1	What is a Schemation DEM Project	9
4.2	Creating a Project.....	9
4.3	Manage Project Privileges.....	10
4.3.1	Privileges – an overview in the context of Schemation DEM	10
5	Environment Admin	11
5.1	What is a 'Database Environment'	11
5.2	Create an Environment	12
5.3	Specify which users can manage this environment.	13
5.4	Define the Environment's Purpose / Build type (Build Config)	14
5.5	Create and Build a Cloned Environment.....	15

1 Introduction

1.1 What is Schemation DEM

Schemation is an out-of-the-box Database Environment Management (DEM) tool that helps you build and manage your applications' databases. It takes your database 'source code' (in the form of plain text files containing ddl and sql, statements, data files etc) and uses these to build and deploy multiple copies of your database.



For a comprehensive list of all features please see <http://www.schemation.co.uk/features.htm>.

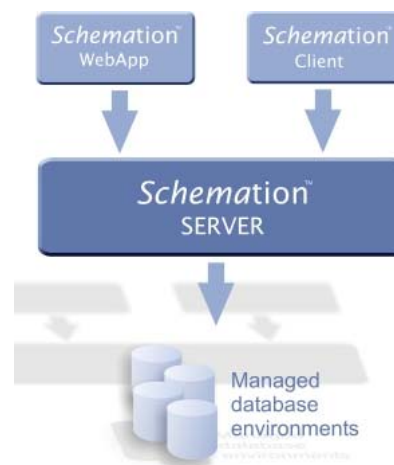
1.2 Prerequisites

This document assumes Schemation DEM's core components (The Schemation Server, Administration Console (Webapp) and Client) are already installed in your organisation. If any of these components are not installed, please see the relevant installation documents.

The Schemation DEM server performs all the database build operations.

The Administration Console (WebApp) provides users with functionality to build and manage their databases environments via a web browser.

The Schemation DEM Client provide users with the ability to create the 'master' database environments (Repositories) that other environments are cloned from and to programmatically interact with to build database environment.



1.3 Section Overview

This document is structured to match the Schemation DEM's main navigation tabs as shown below and is ordered logically to follow the areas you need to visit after installation.



Tablespace Admin (Translations)	Schemation DEM needs to know where to store the database objects that you create; this section (Section 2) tells you how to do to this.
User Admin	You, your developers, your test managers and other users will want to access Schemation DEM, this section (Section 3) provides you with instruction on how to administer users.
Project Admin	The DEM allows you to manage multiple projects that can be allocated different privileges and user access. This section (Section 4) describes the steps that you take to set up and manage different projects. Typically you have one project per application you are developing, but this is entirely up to you.
Environment Admin	The whole purpose of Schemation's DEM is to manage database environments and this section (Section 5) describes the administration steps associated with this task.
Error! Reference source not found.	As an administrator, you will want to know what is going on inside the DEM, this section (Section Error! Reference source not found.) describes the DEM's dashboard which provides visibility to its internal workings and current utilisation.

1.4 Getting Started / Logging into DEM Admin

Before you can perform any administration tasks you need to login to the DEM Administration Console as the Schemation Administrator. The person who installed your installation should provide you with the URL and login credentials. The default administrator user id and password is SCH_ADMIN/SCH_ADMIN, but this should have been changed by the person who installed the DEM in your organisation.

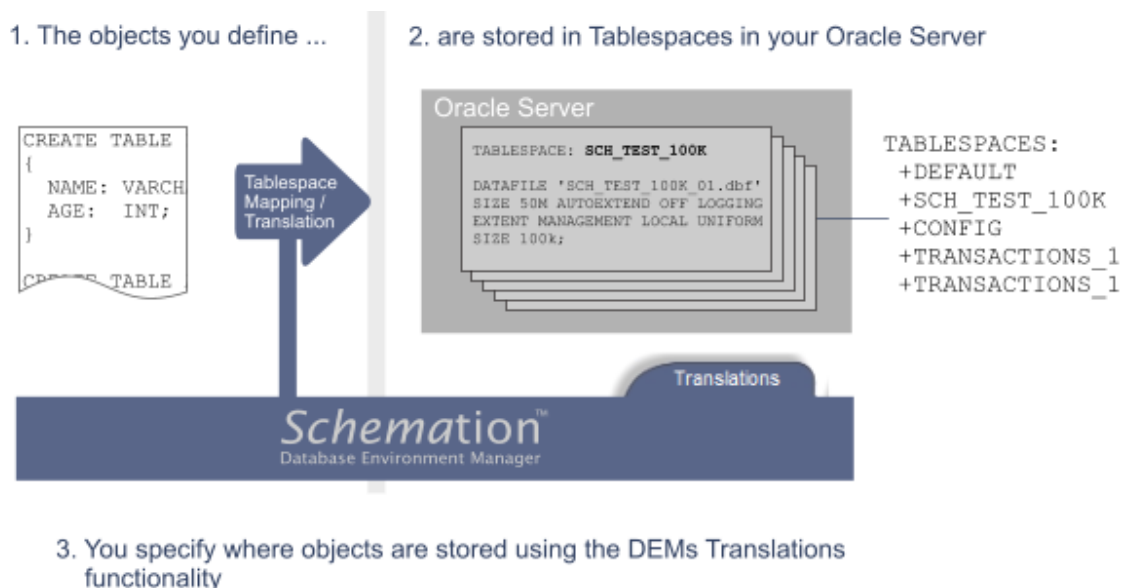
Upon log-in (and for the first time) you will notice each page selected by clicking on the tabs is empty – this is because you have not yet defined any users, projects or environments. This is what you need to do first and will be covered in this documents remaining chapters.



2 Tablespace Admin

2.1 Overview

The database environments that you create comprise of objects such as tables, stored procedures, triggers etc and these are stored physically in Tablespaces. **The DEM does not create Tablespaces but needs to reference the ones you have already created** using normal DBA procedures. When the DEM builds your database environments it uses these references to determine where to store the objects that you create.

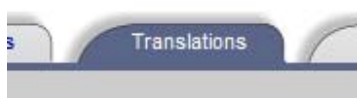


2.2 Create a Tablespace Mapping

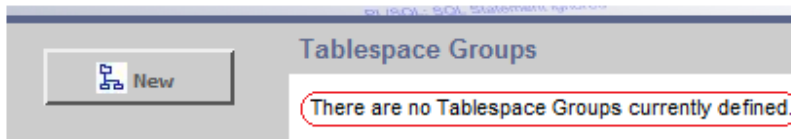
Most applications store their database objects in more than one Tablespace so, to support this, the DEM provides Tablespace Grouping which groups multiple related Tablespaces. The following paragraphs show you how to create a **Tablespace Group** and new **Tablespace Mappings** within that group.

2.2.1 Create a Tablespace Group.

1. Click on the 'Translations' tab.

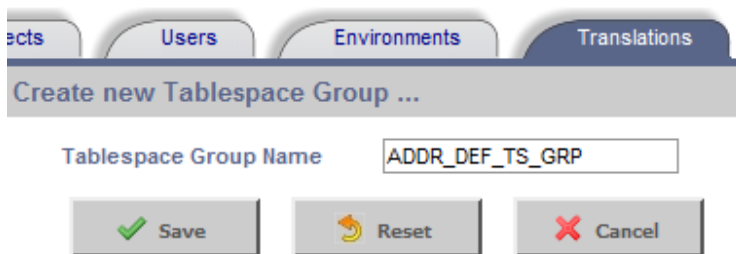


The first time you do this you will notice that there are no Tablespace Groups defined.



2. Click on 'New' and enter a name for the Tablespace Group you want to create and then click 'Save'

The name can be anything but should mean something to you and relate to Tablespaces that will be used by your project. In the example below we have chosen the name 'ADDR_DEF_TS_GRP' which means to us 'The default tablespace-group for the Address Book application'. Choose any naming-convention that works for you.



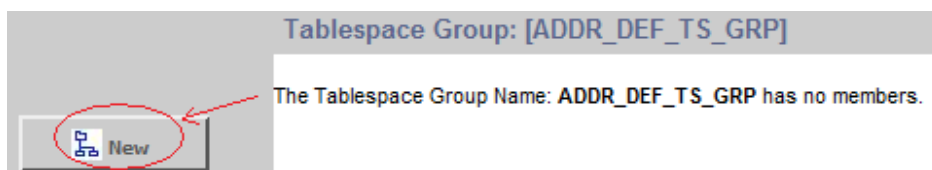
2.2.2 Create a Tablespace Group Member.

Now you have created a Tablespace Group you need to add Tablespace mappings to this group; to do this you need to:

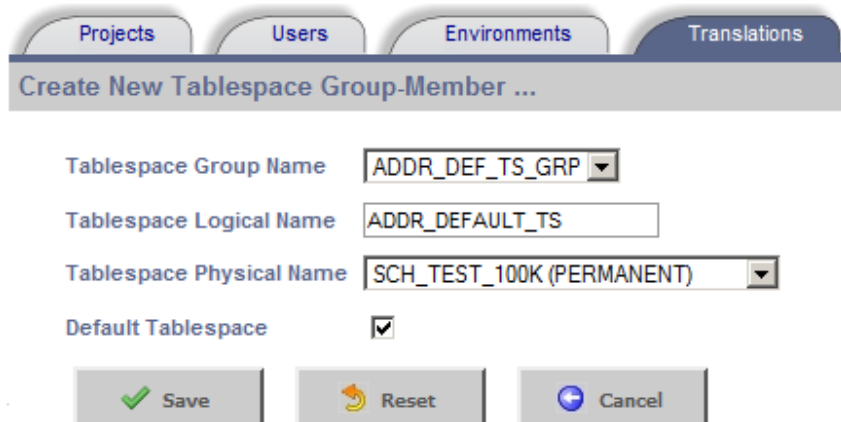
1. Open the Tablespace Group for which you wish to add new members. (Click on the name of the Tablespace Group you have just created)



3. Create a new Tablespace group member:



4. Define Tablespace member properties:



Projects Users Environments Translations

Create New Tablespace Group-Member ...

Tablespace Group Name ADDR_DEF_TS_GRP

Tablespace Logical Name ADDR_DEFAULT_TS

Tablespace Physical Name SCH_TEST_100K (PERMANENT)

Default Tablespace

Save Reset Cancel

- a. **Tablespace Group Name:** Select which group the member will belong to (since you navigated to this page from a link it will be pre-populated with the correct group name that you just created so leave this as is)
- b. **Tablespace Logical Name:** Chose a Logical name that you may optionally reference in your DDL code in your code to allocate specific objects to specific Tablespaces.
- c. **Tablespace Physical Name:** Chose which actual specific Tablespace this logical Tablespace will map to when it is the build process is performed
- d. **Default Tablespace:** 'Check' or 'Uncheck' the 'Default Tablespace' checkbox to indicate whether this tablespace should be used as a default if your DDL does not specify where objects should be stored. Note you can only have one default Tablespace in a Tablespace group.

3 User Admin

3.1 What is a Schemation User

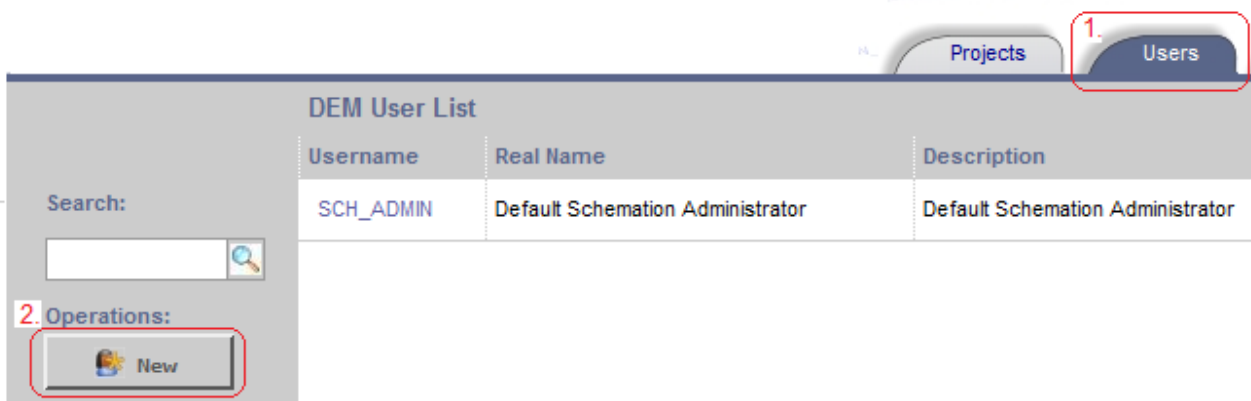
A Schemation DEM User is a regular Oracle User that the DEM has granted specific permissions to create and manage environments within the context and control of the Schemation DEM. A Schemation user can be created by the DEM User Admin page or created by any other DBA means and then registered as a Schemation user using the DEM's Register Existing User functionality.

The following paragraph shows you how to **'Create Users'** or **'Register Existing Users'** using the DEM.

3.2 Create Users

To create users:

1. Click on the 'User' tab and press the 'New' button



2. Populate the new user fields:

- a. **Username:** Chose a short username for the person (or system account) that will be using the DEM to build and manage environments.
- b. **Password:** Specify a password for this user.
- c. **Real Name:** Specify this persons real name (or account name/description if a system account)
- d. **Role / Job Description:** Should you need a further means of identifying this individual you can specify a role or job description or any other description.
- e. **Telephone No:** If appropriate or relevant, you can specify the person's telephone number that may help you as an administrator should you need to contact them for any reason
- f. **Super User:** If you want this person to be able to manage all environments in a given project (e.g. they are a lead developer or project manager) you can set this flag to 'Y' otherwise leave as 'N' and they will only be able to control their own environments.

- g. **User Type:** If you want this user to be an administrator and perform the functions as you are doing now then choose 'ADMIN' from the drop down, otherwise leave as 'USER' and they will only gain the limited privileges that you grant them.
- h. **Default Profile:** This functionality is not yet enabled so can be ignored
- i. **Report Type:** When a user logs in to the DEM Admin console you (or they) can choose which environments they can see in the summary page: They can see (1) All Environments that are being managed by the DEM, (2) All Environments in projects which they are associated with or (3) just the environments for which they are controllers of.

The screenshot shows the 'Create New User' form in the Schemation Database Environment Manager. The form is titled 'Create New User ...' and has two tabs: 'Projects' and 'Users'. The 'Users' tab is selected. The form contains the following fields and values:

- Username: smithj
- Password: [Redacted]
- Password Conf: [Redacted]
- Real Name: John Smith
- Role / Job Description: Developer on the 'Address Book' project
- Telephone: 0123 456 7890
- SuperUser: N
- User Type: USER
- Default Profile: STANDARD
- Report Type: All my envs

At the bottom of the form are three buttons: 'Save' (with a green checkmark icon), 'Reset' (with a yellow circular arrow icon), and 'Cancel' (with a red X icon).

At this stage we recommend that you create your own administrator account for your DEM installation and delete the default SCH_ADMIN one.

3.3 Register Existing User

In rare (but supported) circumstances you may wish to nominate an existing Oracle User as a Schemation Controller (User). To do this you need to use the 'Register Existing User' functionality by:

1. Click on the 'User' tab and press the 'Register' button

Projects Users

DEM User List

Operations:

New

Register

Username	Real Name	Description
SCH_ADMIN	Default Schemation Administrator	Default Schemation Administrator
SMITHJ	John Smith	Developer on the Address Book project

2. For Username chose one of the existing Oracle users from the drop down list them specify the remaining values as you would for a new user described above.

Projects Users

Register an Existing Account...

Username

Real Name

Password

Password Conf

Role / Job Description

Telephone

Is a Super User

Default Scheme Name

Register

ADDR_SIMON_DEMO

ADDR_SIMON_DEMO

ADDR_TEST_RO

ANDY

AURORA\$JIS\$UTILITY\$

AURORA\$ORB\$UNAUTHENT

BOATHK

DAYLEJ

DB_READONLY

DDD

FFF

FUSER

HANSELM

KEITH

LISA

MISERYK

NAZRUL

NISLAM

OSE\$HTTP\$ADMIN

ROR

4 Project Admin

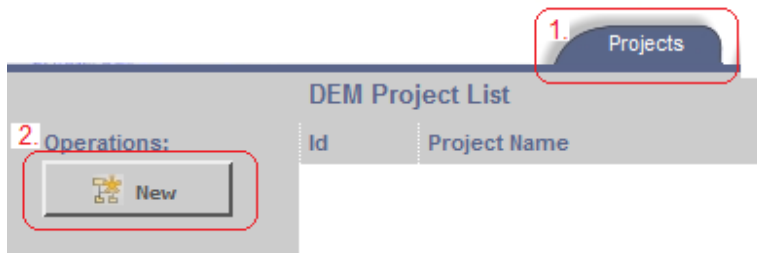
4.1 What is a Schemation DEM Project

A Schemation DEM project is of grouping all environments, users, tablespace, privileges and other configuration together in one manageable chunk. Typically you would create one project per application development you are working on but this is entirely up to you.

4.2 Creating a Project

To create a project:

1. Select the 'Project' administration tab and click on the 'New' button



2. Populate the new user fields:
 - a. **Project Id:** Choose and 4 alphanumeric characters as an ID for your project; this ID will be used programmatically to uniquely identify all aspects of you project and environment with it. In the example below we have chosen ADDR as is help us to identify this project is the 'Address Book' project.
 - b. **Project Name:** Choose any name for your project which will be displayed on several pages in the administration console to help you and your colleagues manage your environments.
 - c. **Project Description:** (Optionally) type a description for your project which will be displayed on several pages in the administration console to help you and your colleagues identify which project you are working on at any given time.

4.3 Manage Project Privileges

4.3.1 Privileges – an overview in the context of Schemation DEM

As a DBA (and outside the world of Schemation DEM) when you create user accounts (schemas) you have to grant privileges to the account you create for them to be useful to the people or applications that use them. The privileges that you grant are based on what the application needs and what your security policy allows; ideally you only assign privileges that are needed and no more. Thus you may grant an account the privilege 'CREATE TABLE' but you may not grant the privilege 'DROP USER'

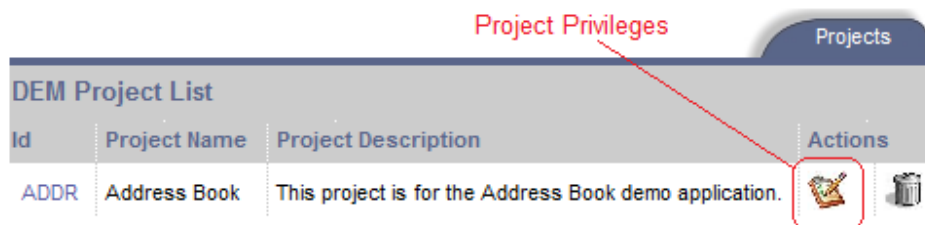
In Schemation's DEM you delegate high powered functions that allow users to drop and create schema to your end users but the DEMs own internal security policy restricts the user so that they can only exercise these power on schemas that have been created by themselves using the DEM. Thus a Schemation DEM user cannot drop or alter any other schema within a given database that has not been created by the DEM itself.

When a DEM user creates a database environment (i.e one or more schemas/account) they will also want to grant the accounts they create with privileges. The DEM allows them to do this but you as a DBA control this defining what privileges that can grant. To do this:

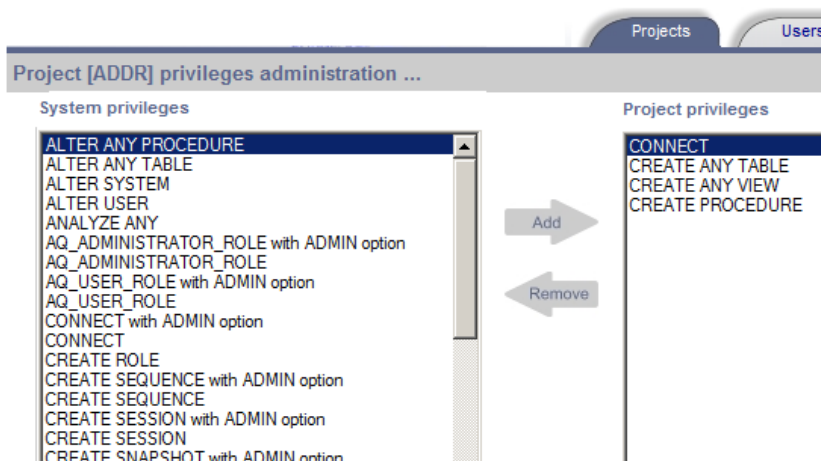
1. Navigate to the 'Projects' tab in the DEM Administration Console



2. For the project you wish to grant privileges to select the 'Manage Privileges' icon as shown below



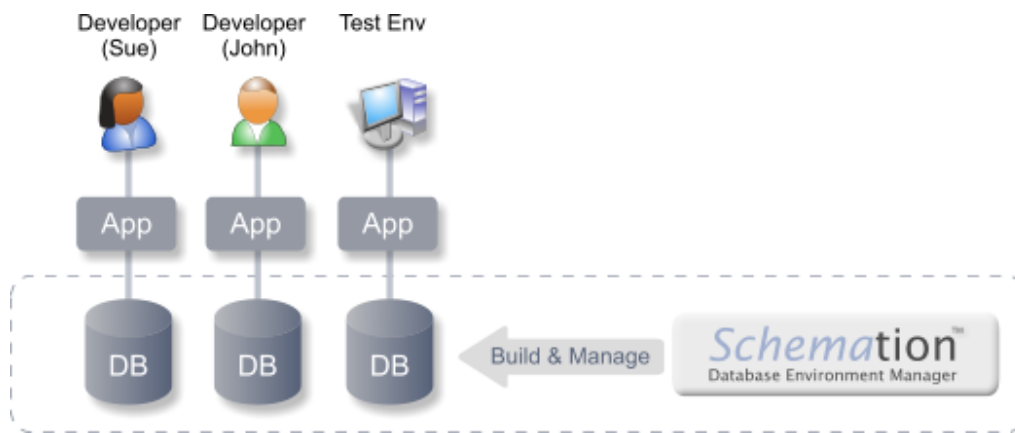
3. Select which privileges you wish to grant to the project in the left panel and click on the 'Add' button.



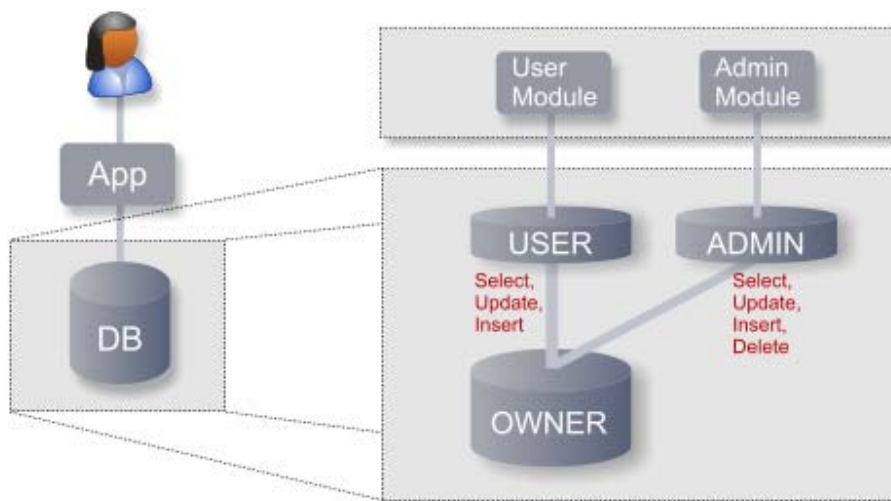
5 Environment Admin

5.1 What is a 'Database Environment'

In the context of Schemation DEM, a Database Environment is an Oracle Schema (or set of Schema's as we shall see later) that your Application uses to store and retrieve its data. Simplistically, we have illustrated three Database Environments (the 'DB' icons) below; each is build by Schemation's DEM from common source code but each developer and test environment gets a personal copy of the database to use exclusively.



In the real world, a typical and well architected Database Environment will consist of several schemas which include 'Accessor' and 'Owner' schemas where 'Owner' schemas own objects and 'Accessor' schema's have limited access to them by way of synonyms. The application then connects to the 'Accessor' schemas and only has access to objects that it needs to perform it function and no more. To illustrate this concept, the 'DB Icon' which represents a Database environment has been expanded to show 'USER', 'ADMIN' and 'OWNER' schemas. The 'USER' and 'ADMIN' schema would not contain any objects but can access the objects in the 'OWNER' schema by way of synonyms.



5.2 Create an Environment

1. Navigate to the 'Environments' tab and click on the 'Environment' or 'Action' columns.



2. Populate the new user fields and click on the 'Create Environment' button to create the environment.
 - a. **Project:** Choose the Project that the Environment will be associated with by selecting the appropriate Project ID from the drop down menu. In the example below we have chosen ADDR as we are creating a new environment for the 'Address Book' project.
 - b. **Environment ID:** Choose any short alpha-numeric ID that will identify this project; for developer's environments it often helps to use their login name to help identify the environment purpose (e.g. SMITHJ, for John Smith's Development environment).
 - c. **Environment Name:** Enter a name for this environment that provides a little more information than the ID alone. This is displayed on various pages in the console like the Environment Summary page.
 - d. **Environment Description:** (Optionally) enter a description for this environment. This field is only presented in the console when you drill down to find out more about the environment's purpose. For example you might to what to use this field to document the different purposes of two UAT environments.
 - e. **Tablespace Group:** Specify which Tablespace Group you would like this environment to. This tells the DEM which Tablespaces it can use to store this environments objects in. (See Section 2 – Tablespace Admin for more information.)
 - f. **Enable Auto Rebuild:** If you later designate this environment to be a 'clone' of another environment you can instruct the DEM to automatically rebuild this environment when there is an update to the 'master' environment. For now, unless you know you need this then leave this check box unchecked.

Projects Users **Environments**

Create New Database Environment ...

Project * ADDR ▾

Environment Id NIGHTLY

Environment Name * Nightly Build Environment

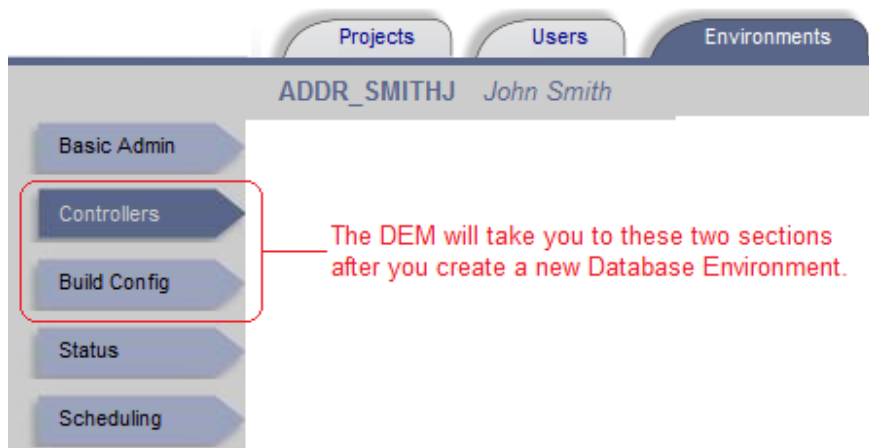
Description
 This environment will be build by Autosys each night and gives to give the Dev team a snapshot / reference build each day

Tablespace Group ADDR_DEF_TS_GRP ▾

Enable Auto Rebuild

Create Environment

After creating an environment the DEM will take you through two simple configuration steps: Controllers' – To specify who can manage this environment and 'Build Config' which allows you to specify what type of environment the will be (i.e a Master (Repository) environment, A clone environment or an independent build.



5.3 Specify which users can manage this environment.

Only users who have been registered as a 'Controller' of an environment can manage it. This prevents 'John' from accidentally rebuilding Paul's environment to do this:

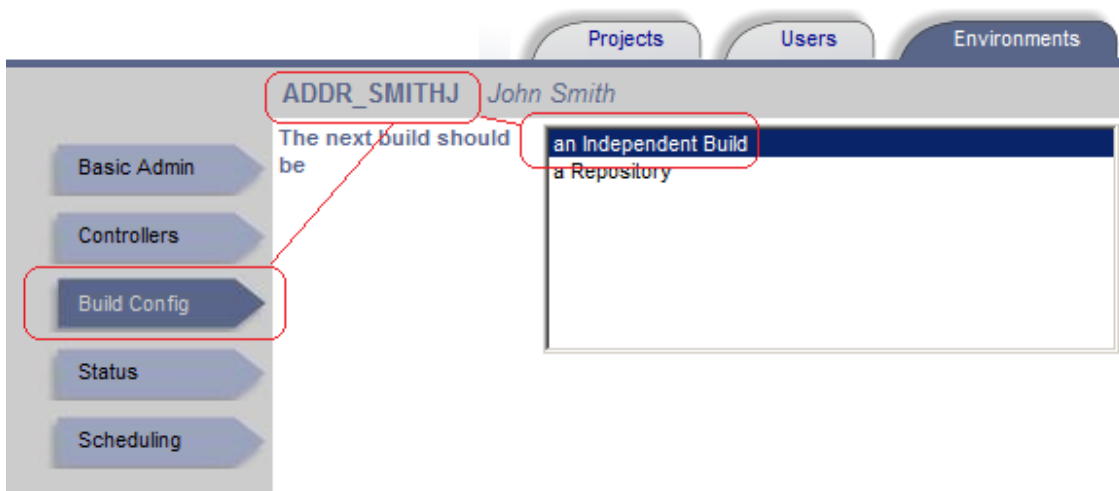
1. Navigate to the Environment/Controllers page (if the DEM has not already taken you there).
2. Choose which user(s) you want to Control this environment in the left hand pane and add them to the right hand page by clicking the 'Add' button in the middle.
3. In the example below we are assigning the developer John Smith to be a controller of the environment 'ADDR_SMITHJ' an environment that was created for him.



5.4 Define the Environment's Purpose / Build type (Build Config)

In the context of Schemation DEM there are three types of Database Environment: **Repository**, **Clone** and **Independent**. After creating an Environment you will need to inform the DEM what its intended, to do this:

1. Navigate to the 'Environments' tab and click on the 'Build Config' 2nd level navigation tab if you have not already been taken there.

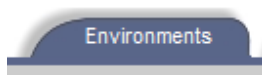


2. Chose which type of Environment you wish to create and click on the save button.


- a. **Repository Environments:** A Repository environment is one that is used as a 'Master' database environment that other environments are cloned from.
- b. **Cloned Environment:** As indicated above these are copies in all or part of Repository Environments. The dropdown list will not give you this option if you have not yet created (and built a valid Repository Environment).
- c. **Independent Environments:** These are stand alone Environments not linked to any others and cannot be cloned. Typically these are set up for developers who wish to do a significant piece of development in a database and do not want to receive updates from the team's shared source.

3. View your environment in the summary page

Click on the 'Environments' tab






In the Environments Summary page you see the environment you have just created and may notice a 'Warning' icon beside it. This indicates that the environment has not yet been built. The environment exists within the context of the DEM but has not yet been built in the target Oracle environment – i.e. no Oracle schemas exist yet. At this stage this is all you can do – create empty environments - ready for them to be build using the DEM Client. Once a Repository Environment has been built you can then create and build/deploy clone environments from this. This is described in the next section. Prior to progressing to the next section though you may wish to consider reading the 'DEM Client User Guide' document for instruction on how to build-out the Repository Environment you have just created.

Environment		Start time	Elapsed time	Operations		
		End time	Status	Total	Pass	Fail
	ADDR_NIGHTLY Nightly Build Environment		Not yet built	0	0	0
			VIRGIN			

5.5 Create and Build a Cloned Environment




Before you can create a Clone environment you need to have a Built Repository Environment from which you clone. To determine if you have this, click on the Environments Summary page and observe at least one environment with a Golden DB icon as shown below.

Environment	Start time End time	Elapsed time Status	Operations			Data	
			Total	Pass	Fail	Load	Rej
 ADDR_NIGHTLY Nightly Build Environment	Mon 18-01-10 12:19 Mon 18-01-10 12:20	1m 23s built	38	5	33	0	
 ADDR Simon							
 ADDR John Smith							

The Golden DB icon indicates this is a Repository Environment from which others can be cloned.

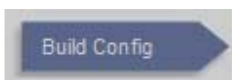
Any cloned environment will contain upto 5 objects depending on your clone criteria

Once you have a 'built' Repository Environment you can make any other environment a clone of it. To do this, choose the Environment you want to become a clone and click on the Environment name as shown below.

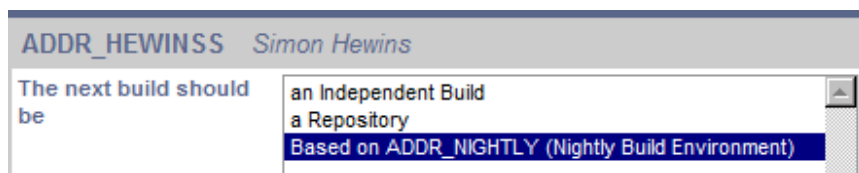
Environment	Start time End time	Elapsed time Status	Operations	
			Total	Pass
 ADDR_NIGHTLY Nightly Build Environment	Tue 19-01-10 00:09 Tue 19-01-10 00:10	20s COMPLETE	531	9
 ADDR_HEWINSS Simon Hewins	Not yet built	VIRGIN	0	0
 ADDR_SMITHJ John Smith		IRGIN	0	0

Click on the Environment you wish to make a clone of another environment.

This will take you to the Environment Details pages where you need to click on the 'Build Config' sub-tab.



From here you can specify what type of environment this will be (Repository, Clone or Independent) as described earlier. To create a Clone Environment, click on the 'Based on XXXX' option in the drop down.



Depending on how the Repository that you are cloning from has been built a list will then appear with all the module that made up that environment. Select which components you want to include in your cloned environment (typically all of them) and the click 'Save'

You will then see a 'Build' button appear.



Click on this build button to initiate the build of the Cloned Environment

Projects Users **Environments**

ADDR_HEWINSS *Simon Hewins*

Basic Admin

Controllers

Build Config

Status

Scheduling

The next build should be

an Independent Build
a Repository
Based on ADDR_NIGHTLY (Nightly Build Environment)

The next build components

Select All
De-select All

CODE audit

base_schema

partitioning

stored_procs

DATA main_config

projects_10000

The audit sub-system

The basic schema objects required by all environments.

The production set of partitioning statements.

The main schema that owns the objects

This is the main configuration data required for all environments.

A set of 10,000 test projects to test data load.

✓ Save

↺ Reset

To check the status of your build, go back to the 'Environments Summary' page and look for your environment. In the example below we have created a clone of the 'ADDR_NIGHTLY' environment that is called ADDR_HEWINSS.

Projects Users

Environment	Start time End time	Elapsed time Status
ADDR_NIGHTLY Nightly Build Environment	Tue 19-01-10 00:09 Tue 19-01-10 00:10	20s COMPLETE
ADDR_HEWINSS Simon Hewins	Tue 19-01-10 00:41 Tue 19-01-10 00:41	7s COMPLETE
ADDR_SMITH John Smith		GIN

This icon indicates that it is a Cloned Environment (i.e. ADDR_HEWINSS will be a clone of ADDR_NIGHTLY)