Spago4Q How To

Authors Spago4QTeam

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Index

creative commons
COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Version

Version n°: **Release n°:**	1.0 2.0	**Data Version:**	March,16th 2009
Update description:	First version		

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

1 Document Goal

The purpose of this document is explain how to use and configure all the part of the platform to have a working environment. All these explanations are based on the sample data provided with the Spago4Q 2.0 release.

1.1 DEFINITIONS

Here are summarized the most important terms used inside this document:

- **KPI**: is the central element of the platform and it's definition is basically descriptive plus some associations (threshold, dataset, analytical document, …).
- **Dataset**: this element is used to perform the computation of the KPI value and need to be selected during the KPI configuration.
- **Threshold**: this is for give a KPI value an additional information based on the range of values the KPI is in. A threshold can be composed by a set of ranges.
- **Model Definition**: here is where is possible to define an abstract model (CMMi, ISO, Custom) with KPI associated.
- **Model Instance**: this is built from a Model Definition and can replicate all its nodes or a sub-tree of them. During the Model Instance definition it's possible modify the KPI associated to every node as well as the KPI configuration itself.
- **KPI Document**: it's the SpagoBI analytical document that has to be defined to represent a Model Instance in a dashboard like view.
- **KPI Engine**: is the internal SpagoBI engine that execute the computation of each KPI value using the associated Dataset.

1.2 REFERENCES

Some of the concepts of this document refer to the following documentation:
Spago4Q-2.0.0-Overview (available at http://www.spago4q.org/)

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

2 Spago4Q in more depth

The chapters of this document detail the development task needed to implement a KPI dashboard using Spago4Q.

The implementation of a KPI dashboard could be a complex work, so we need to start a specific project. In the analysis phase we define all the requirements and identify the source of the measure attributes needed to calculate the KPIs.
Here we assume that the analysis phase is completed and we have to start the development phase of the project.

Phase 1: extract and load the data into the dwh (chapter 3)
- task 1 – configure the data source (chapter 3.1)
- task 2 – configure and create the target database (chapter 3.2)
- task 3 – define the extraction and transformation process (chapter 3.3)

Phase 2: implement the dashboard
- task 1: Define a model (chapter 4)
- task 2: Define kpis and thresholds (chapter 5)
    - task 2.1: create "dataset" , the dataset implements the kpi's algorithm (chapter 7)
    - task 2.2: create "analytical document" , the "analytical document" represents the kpi such as graph or a "composed document" (refer to SpagoBI documentation)
- task 3: Create a Model Instances (it establishes relationship between: model or a subset of a model, kpis and thresholds, resources (projects, products, …) (chapter 6)
- task 4: Configure the dashboard (named "KPI document" in the following) that represent the KPI included into the model instance (chapter 8).

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

3 Extraction process

One of the main features of Spago4Q is the capability to do ETL operations.
ETL operations are:

- Extract data from outside sources
- Transforming it to fit operation needs
- Loading it into the end target

In Spago4Q you have the possibility to configure and realize this operations following this steps:

- Configure the data source
- Configure and create the target database
- Define the extraction and transformation process

In this part of this guide will be describe how to do this operation in Spago4Q.

We assume that you login as Spago4Q administrator pointing the browser to:

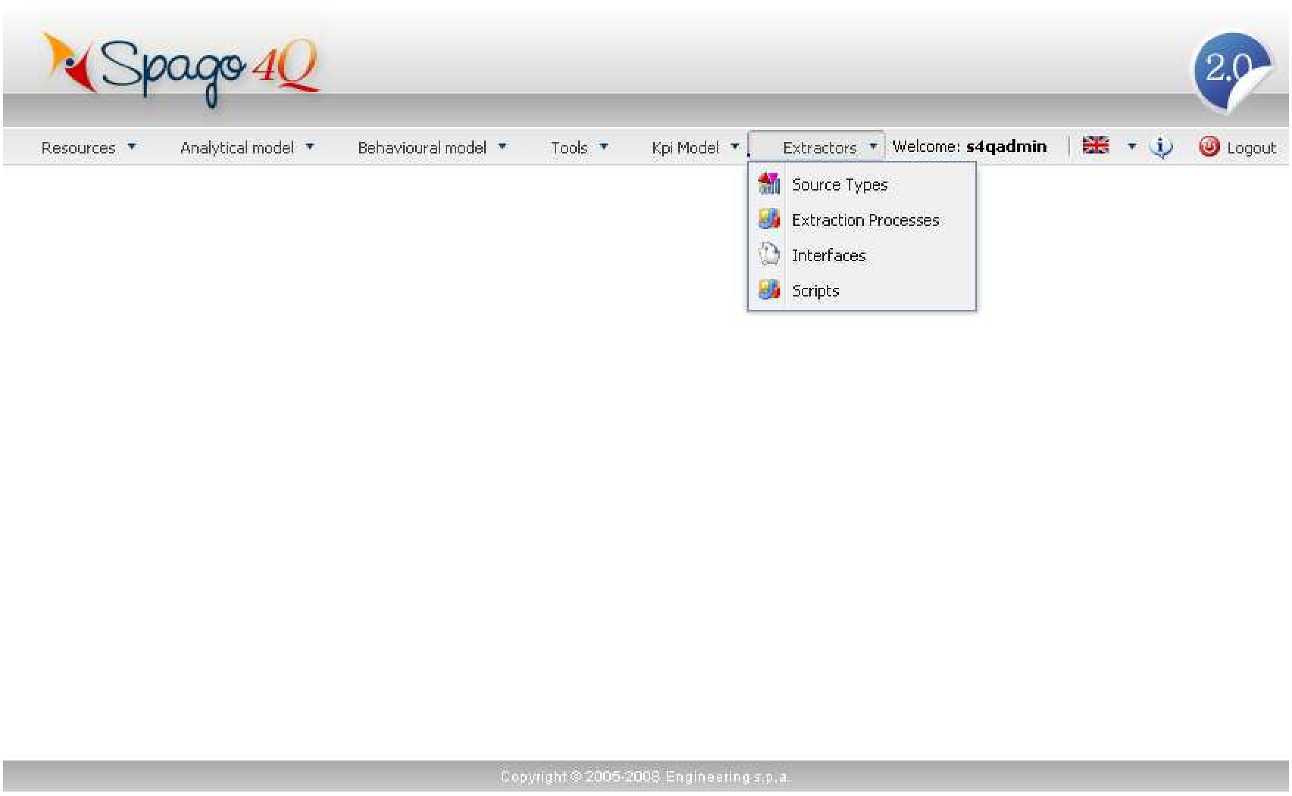
URL	http://localhost:8080/SpagoBI

Login as:

User	s4qadmin
Password	s4qadmin

This is what you see:

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/
BY:



3.1 HOW TO CONFIGURE THE DATA SOURCE

In this section it will be explained how to configure the data source. The data source is the place where will be performed the extraction. To create the data source you have to create and configure:

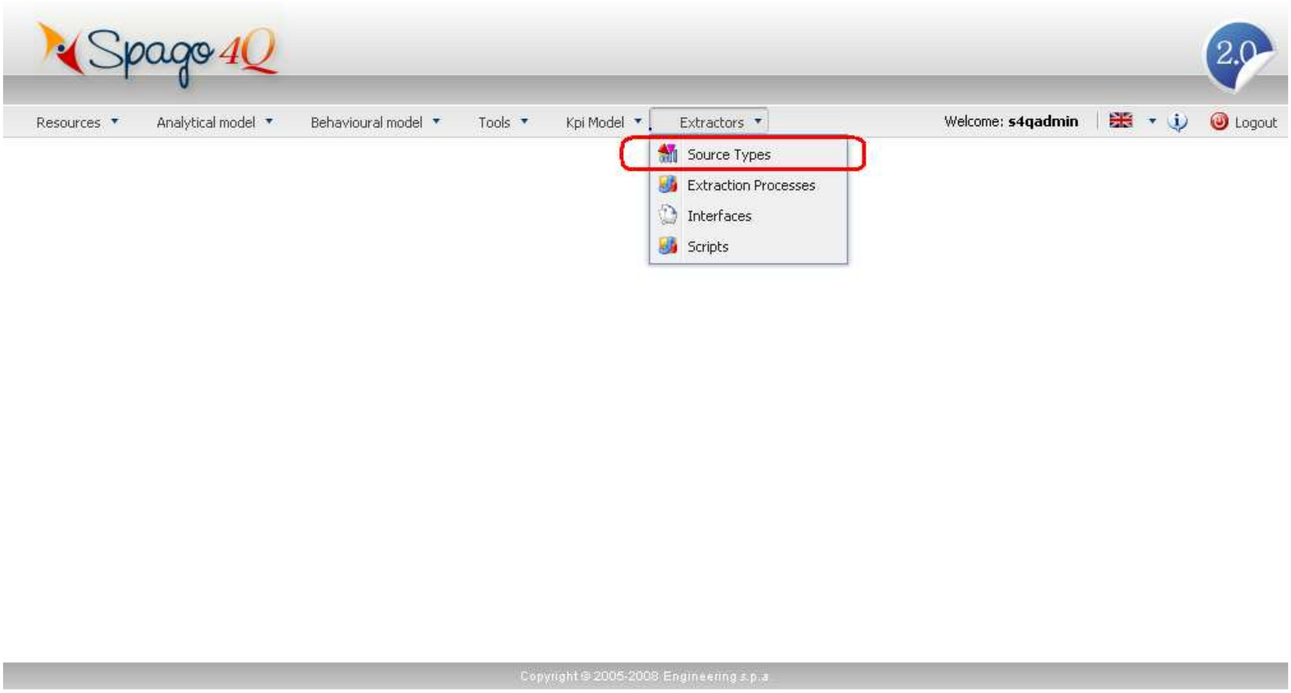
- The source type: where to specify the type of data source (XML, Data Base, etc.) and configure the engine (Java class) that extract the data.
- The data source.
- The data source parameters: specify the parameter to access the data source.

In this guide it will be created and configured a Data Base data source.

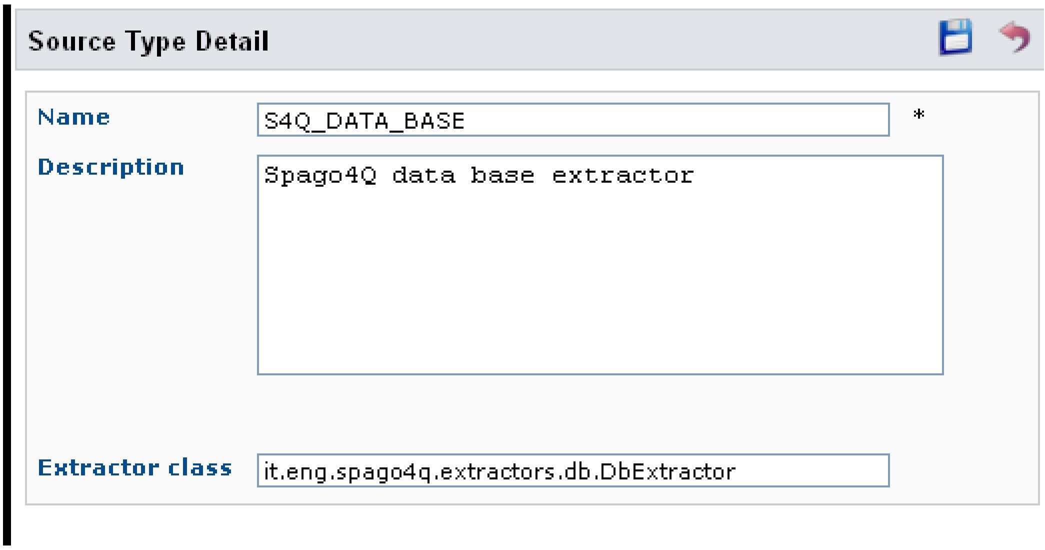
3.1.1 CREATE A SOURCE TYPE

Click on the *Source Types* menu item and you will enter into the list of existing Source Types.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/
BY:



In order to create a new Source Type click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Name: S4Q_DATA_BASE *

Description: Spago4Q data base extractor

Extractor class: it.eng.spago4q.extractors.db.DbExtractor

In this form you can insert this fields:

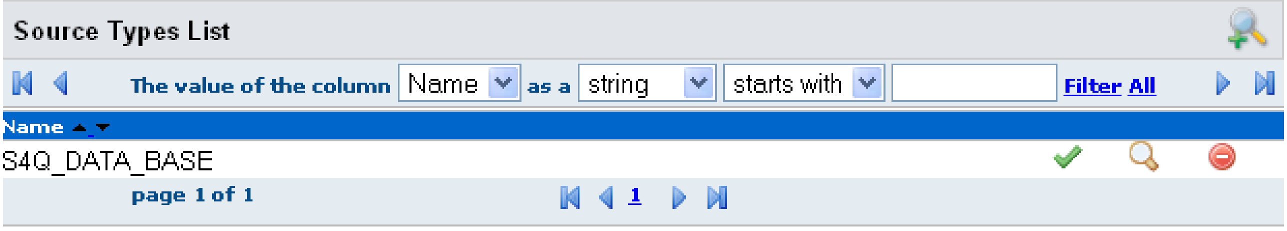
- Name (mandatory): it is the label of the Source Type and it identifies it.
- Description: Describe the Source Type
- Extractor class: it is the name of the Java class that perform the extraction. In this version you can use the following extractor classes:

Extractor class	it.eng.spago4q.extractors.db.DbExtractor	This class extracts data from a Database using a SQL-92 query

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Data Source parameters	USER	The username that has the access to the Database
	PASSWORD	The password of the user
	URL	The url that identifies the Database
	DRIVER	The driver to perform the queries
Operation Parameters	QUERY	The query to do when spago4Q execute an extraction

Enter the required information and click on the "Save" button . Than click on the "Back" button and you will see the new Source Type just inserted in Source Types list.

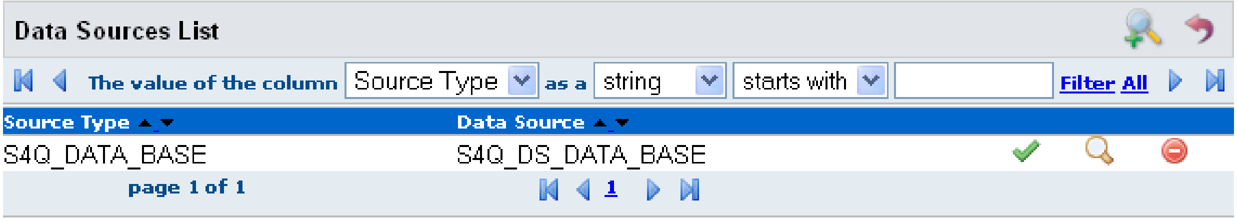


Name
S4Q_DATA_BASE

3.1.2 CREATE A DATA SOURCE

In Spago4Q a Data Source is an instance of a Source Type with its own parameter. To create a Data Source you have to perform this operations:

- Click on the menu item "Source Types" and you will enter into the list of existing Source Types.
- Click on the "View Data Sources" button ✓ and you will enter into the list of existing Data Sources that use a specific Source Type.

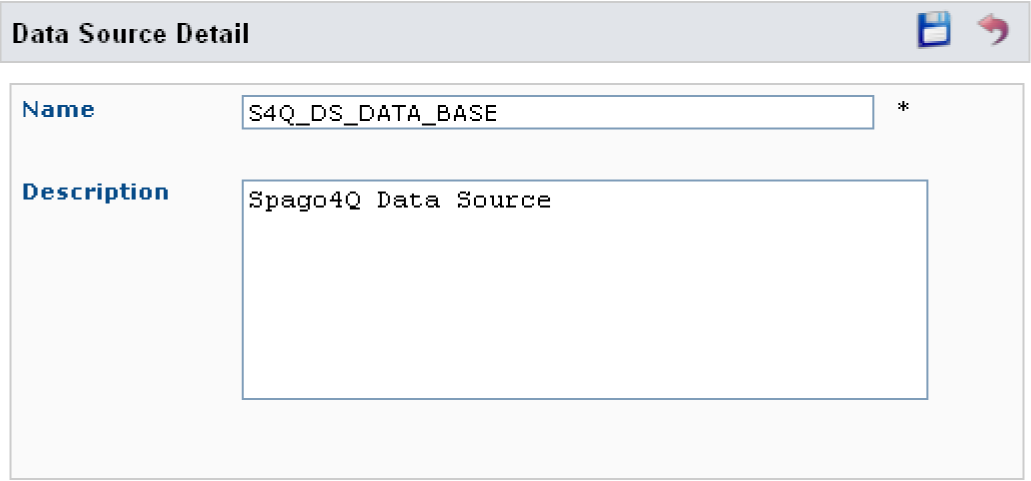


Source Type	Data Source
S4Q_DATA_BASE	S4Q_DS_DATA_BASE

In order to create a new Data Source click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Name	S4Q_DS_DATA_BASE *
Description	Spago4Q Data Source

In this form you can insert this fields:

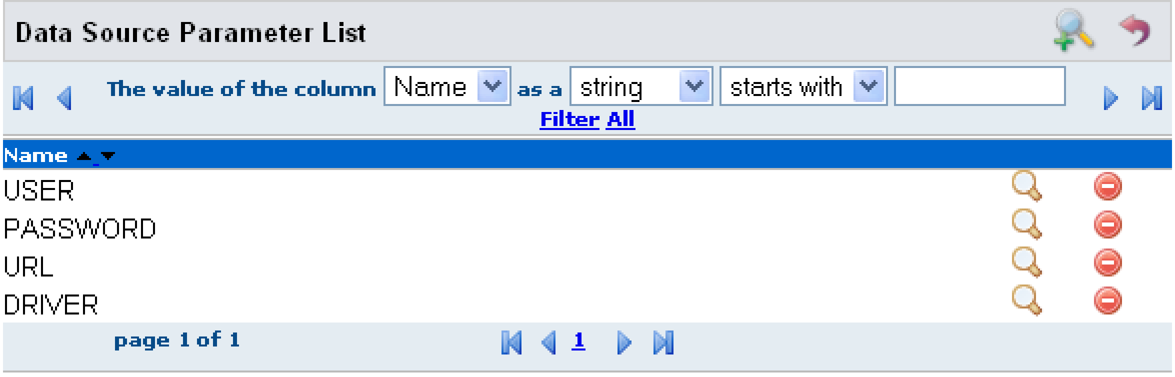
- Name (mandatory): it is the label of the Data Source and it identifies it.
- Description: Describe the Data Source.

Enter the required information and click on the "Save" button 💾. Than click the "Back" button ↶ and you will see the new Data Source just inserted in Data Sources list.

3.1.3 CREATE DATA SOURCE PARAMETERS

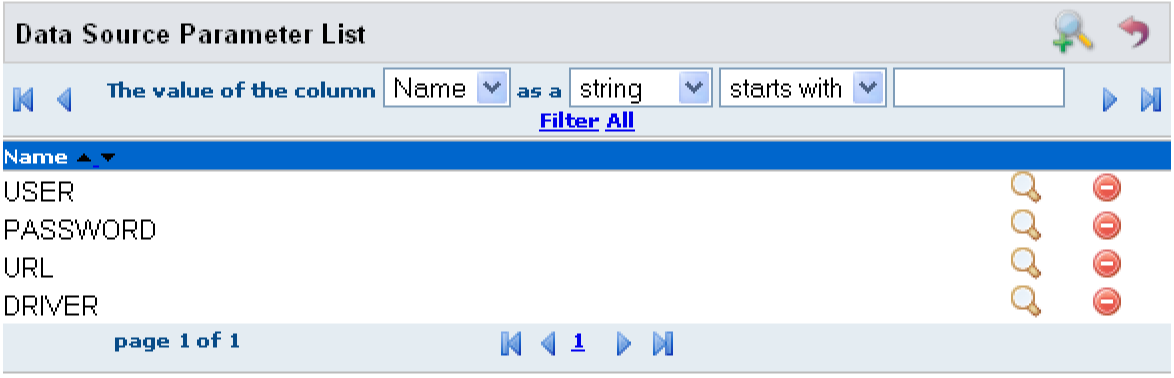
To perform an extraction the Extraction class (that is a field of a Source Type) needs same parameters that describes how the extractor can connect to the Data Source. A Data Source is an instance of a Source Type with its own parameters. To specify the parameters of a Data Source you have to do this operations:

- Click on the menu item "Source Types" and you will enter into the list of existing Source Types.
- Click on the "View Data Sources" button ✓ and you will enter into the list of existing Data Sources that use a specific Source Type.
- Click on the "View Data Sources Parameters" button ✓ and you will enter into the list of existing Data Source Parameters of a specific Data Source.



Name
USER
PASSWORD
URL
DRIVER

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Data Source Parameter List

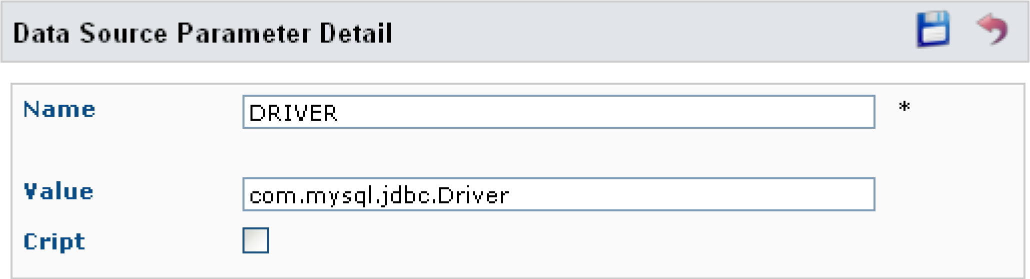
The value of the column Name as a string starts with

Filter All

Name
USER
PASSWORD
URL
DRIVER

page 1 of 1 — 1

In order to create a new Data Source Parameter click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Data Source Parameter Detail

Name	DRIVER	*
Value	com.mysql.jdbc.Driver	
Cript	☐	

In this form you can insert this fields:

- Name (mandatory): it is the label of the Data Source Parameter and it identifies it.
- Value: is the value of the field.
- Cript: if it is select, when you click on the "Save" button, the value will be stored encoded in Spago4Q platform.

Enter the required information and click on the "Save" button. Than click the "Back" button and you will see the new Data Source Parameter just inserted in Data Source Parameters list.

To configure a Data Source that implements a Source Type that has it.eng.spago4q.extractors.db.DbExtractor as extractor class you have to insert the following parameters:

- USER
- PASSWORD
- URL
- DRIVER

3.2 HOW TO CONFIGURE AND CREATE THE TARGET DATABASE

In this section will be explained how to configure the target database. In Spago4Q the target database is defined by an Interface type that describes a fact table where to put the data extracted from the extraction process. To create the fact table you have to do this step:

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

- Create an Interface type where you define the name of the fact table.
- Create same Interface field. An interface field represents a column of a fact table.

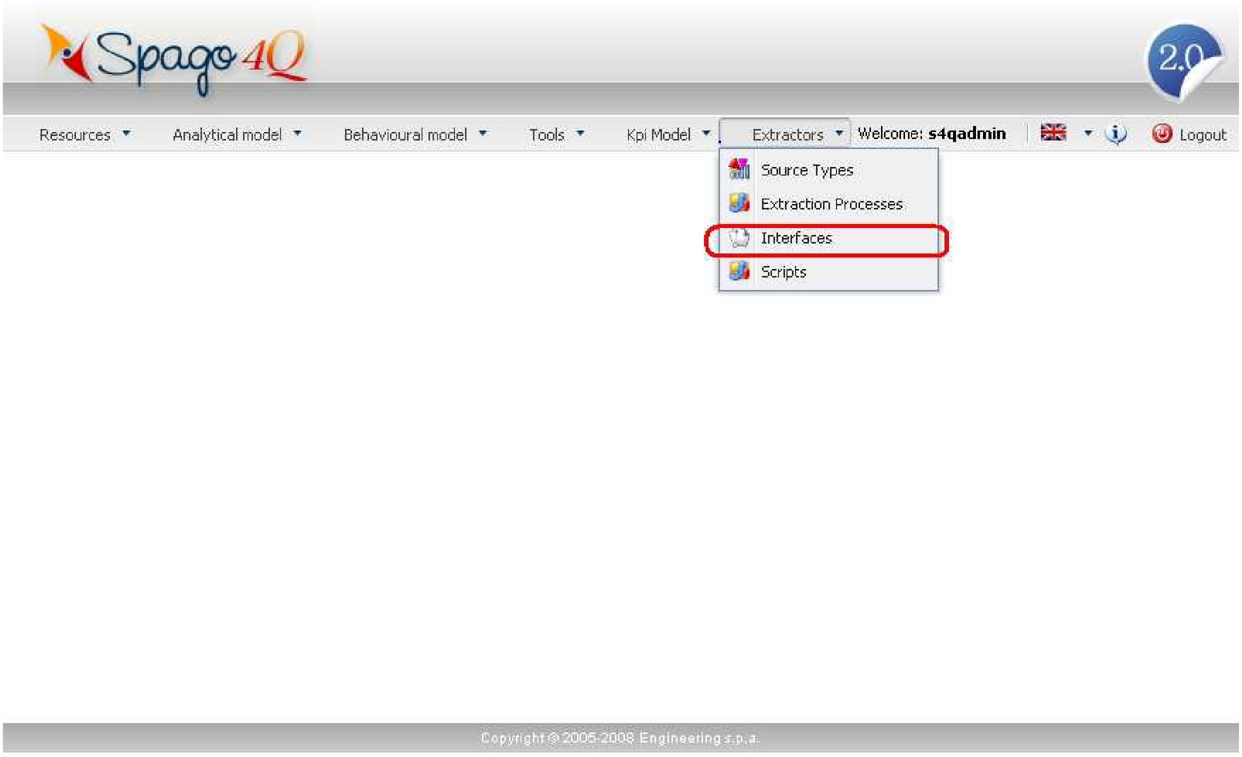
In this guide it will be created an Interface type that represent this table:

FT_PROJECT_MAN		
Field	Type	Key
PRJ	VARCHAR	YES
AC	DOUBLE	NO
PV	DOUBLE	NO
EV	DOUBLE	NO

3.2.1 CREATE AN INTERFACE TYPE

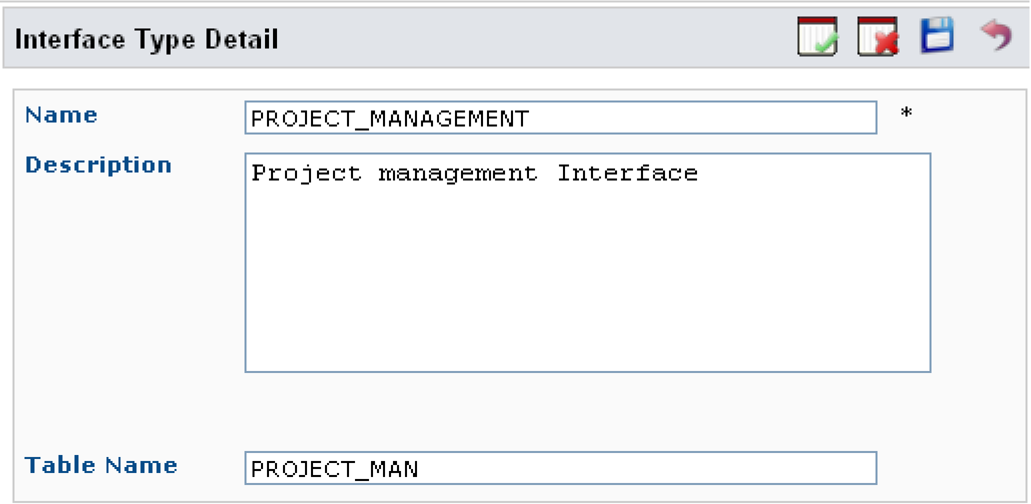
An Interface Type represent a fact table that will be create in the database. This table will be populated with the data extracted from the extraction process. To create an Interface Type you have to do this operations:

- Click on the menu item "Interfaces" and you will enter into the list of existing Interface Types.



In order to create a new Interface Type click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Interface Type Detail

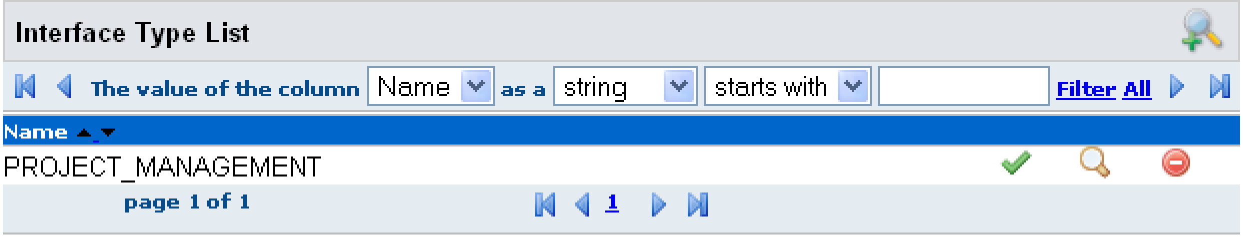
Name	PROJECT_MANAGEMENT *
Description	Project management Interface
Table Name	PROJECT_MAN

In this form you can insert this fields:

- Name (mandatory): it is the label of the Interface Type and it identifies it.
- Description: describe the interface type
- Table Name: it is the name of the fact table to create. When you create a fact table, To this vale will be automatically added a prefix like "FT_".

In the example explained in the figure will be create a fact table named "FT_PROJECT_MAN".

Enter the required information and click on the "Save" button 💾. Than click the "Back" button ↶ and you will see the new Interface Type just inserted in Interface Type list.



Interface Type List

The value of the column Name as a string starts with Filter All

Name
PROJECT_MANAGEMENT

page 1 of 1

3.2.2 CREATE INTERFACE FIELDS

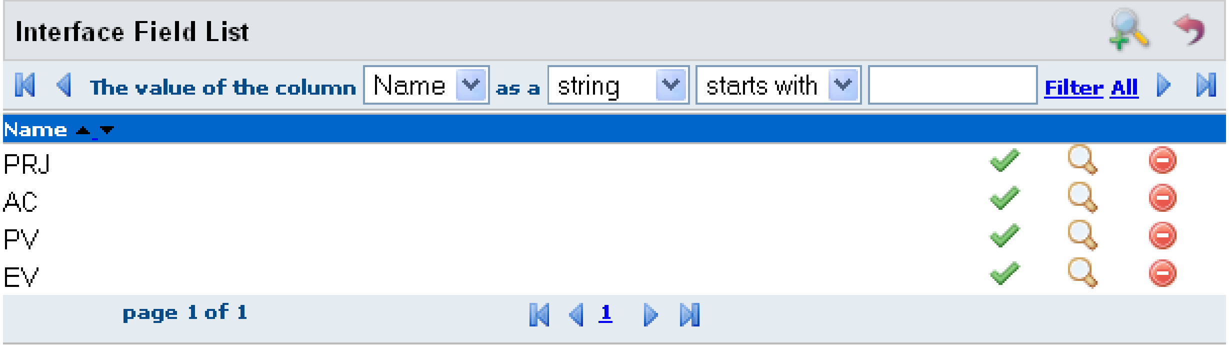
To replicate the table explained before in the example you have to add to the Interface Type this field:

PRJ	STRING
AC	DOUBLE
PV	DOUBLE
EV	DOUBLE

To Associate same field to an Interface type you have to do this operations:

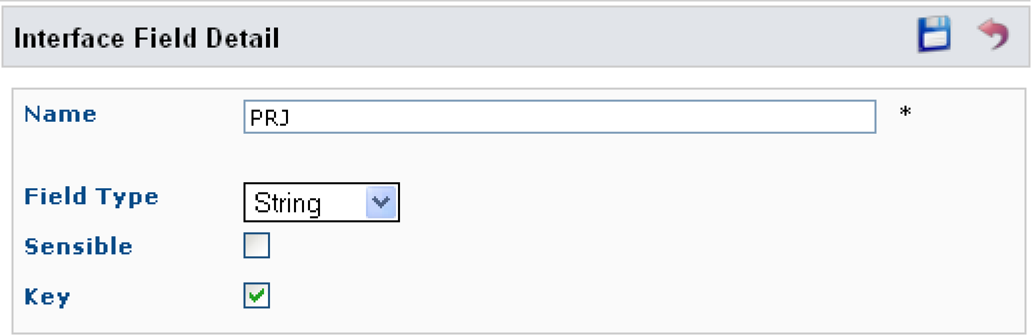
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

- Click on the menu item "Interfaces" and you will enter into the list of existing Interface Types.
- Select the interface type that you want to add the new fields (in the example select "PROJECT_MANAGMENT") and click the "View Interface Fields" button ✓ and you will enter into the list of Interface Fields of a specific Interface Type.



Name
PRJ
AC
PV
EV

In order to create a new Interface field click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Name	PRJ *
Field Type	String
Sensible	☐
Key	☑

In this form you can insert this fields:

- **Name** (mandatory): it is the label of the Interface Field and it identifies it. It represents the name of a column of the fact table specify by the Interface Type.
- **Field Type**: it is the type of the column that will be created in the fact table. The type that you can select are:
    - String
    - Integer
    - Double
    - Boolean (0,1)
    - Date (Specify in the format of the Database where the fact table will be created)
- **Key**: specify if this column identify a row of the Data base
- **Sensible**: If it is checked if an extraction produce two rows with the same values as key and different values as sensible field produce the insertion of two rows. If an extraction of two rows with the same values as key and same value as sensible field produce the insertion of only one row.

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Enter the required information and click on the "Save" button . Than click the "Back" button and you will see the new Interface field just inserted in Interface fields list.

If you want to replicate the example explained before you have to insert the following interface field:

Name	Field Type	Sensible	Key
PRJ	STRING	NO	YES
AC	DOUBLE	YES	NO
PV	DOUBLE	YES	NO
EV	DOUBLE	YES	NO

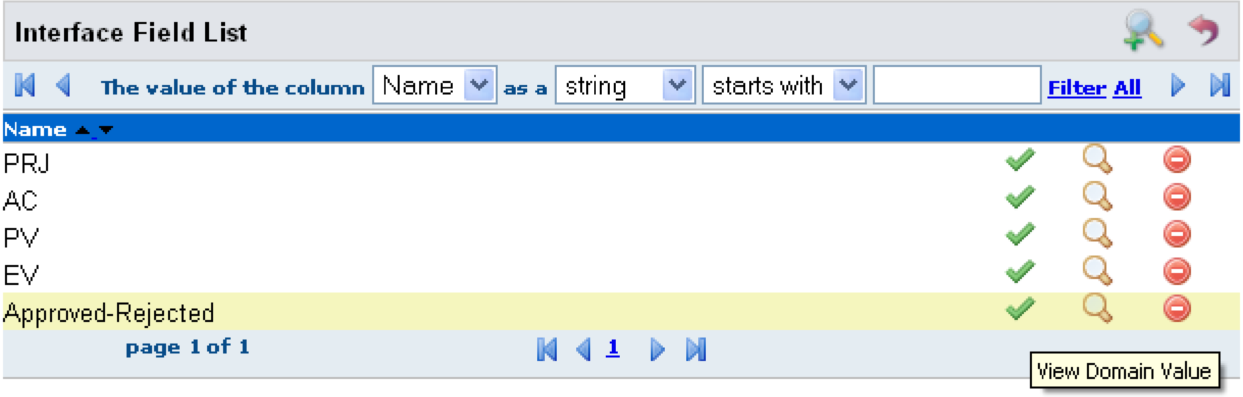
3.2.3 DOMAINS VALUE

Sometime an extraction can produce a column of a finite set of values. In the following example the extraction produced this data:

ID	proposal	approved/rejected
1	proposal1	rejected
2	proposal2	approved
3	proposal3	approved

The column "approved/rejected" can have only this set of value : {approved, rejected}.
To optimize the quantity of data that will store from an extraction, Spago4Q has the capability to define a Set of value that can be repeated in an extraction. When an extraction will be performed Spago4Q replace this value with the identifier of the value that has be specified in the Domain Value.
To associate a Domain Value to an interface field you have to do this operations:

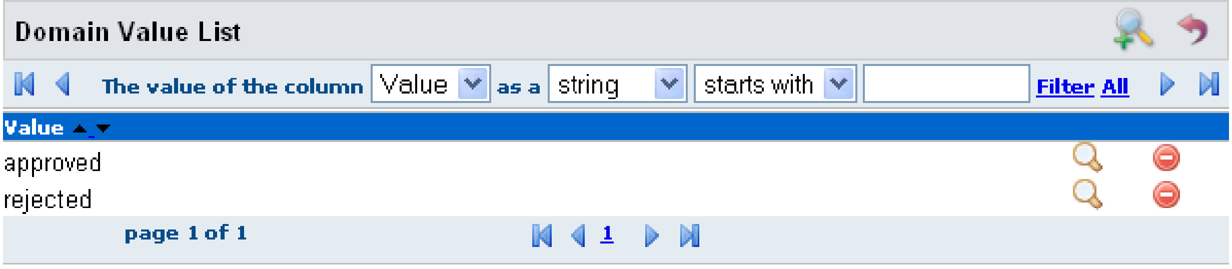
- Click on the menu item "Interfaces" and you will enter into the list of existing Interface Types.
- Select the interface type and click the "View Interface Fields" button and you will enter into the list of Interface Fields of a specific Interface Type.



PRJ
AC
PV
EV
Approved-Rejected

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Select the interface field that you want to associated a Domain Value and click the "View Interface Fields" button ✓ and you will enter into the list of Interface Fields of a specific Interface Type.



Domain Value List

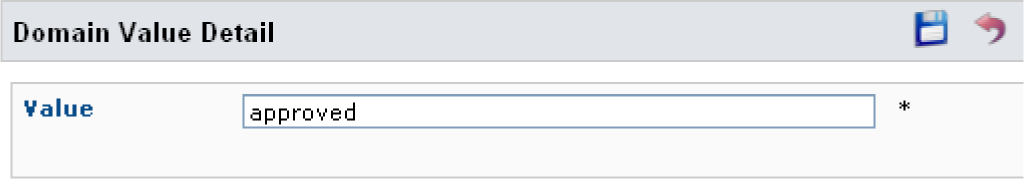
The value of the column Value as a string starts with

Value
approved
rejected

page 1 of 1

In order to create a new Domain Value click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:

- In this form you can insert this fields:
  - **Value**: identify one of the value in the set.



Domain Value Detail

Value: approved *

Enter the required information and click on the "Save" button. Than click the "Back" button and you will see the new Domain Value just inserted in Interface fields list. To complete the example insert two Domain Value:

- approved
- rejected

When you execute an extraction that produce the value show before, the column "approved/rejected", if it is associated with an interface field that has the domain value just inserted will be produce this result:

ID	proposal	approved/rejected
1	proposal1	{ID Domain Value rejected}
2	proposal2	{ID Domain Value rejected}
3	proposal3	{ID Domain Value rejected}

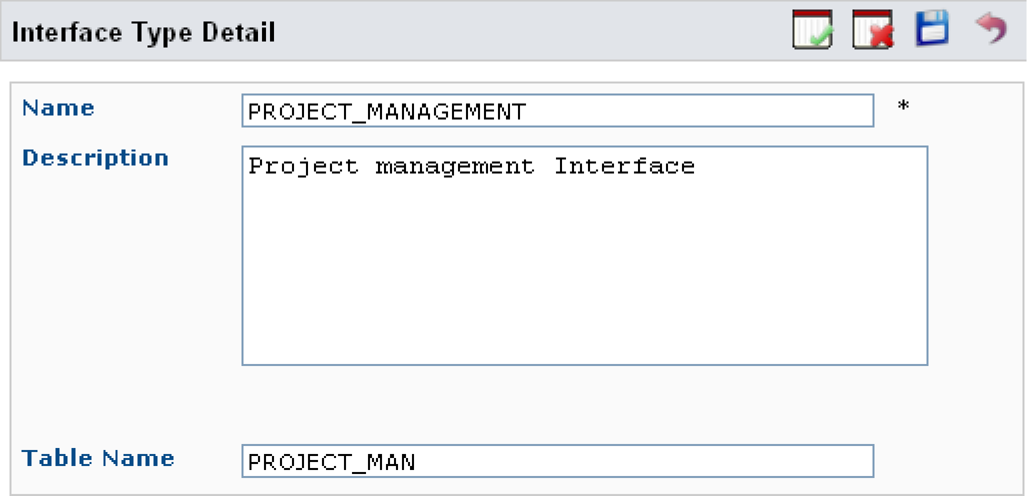
Where {ID Domain Value rejected} will be the identifier of the data set with the value rejected.

3.2.4 CREATE THE FACT TABLE

After you defined the Interface Type and the Interface Fields that represent the Fact Table you can create the fact table in the Data Base. To create the Fact Table in the Data Base you have to do this operations:

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

- Click on the menu item "Interfaces" and you will enter into the list of existing Interface Types.
- Click the "Select" button of the interface type that represents the fact table that you want to create and you will enter in a form like the following one:


Interface Type Detail

Name	PROJECT_MANAGEMENT *
Description	Project management Interface
Table Name	PROJECT_MAN

To create the fact table click the "Create Fact Table" button . If you check in the database you will find a table named as "FT_"+ the name specify in the "Table Name" field of the Interface Type with the field specify in the Interface Fields associated of this Interface Type.

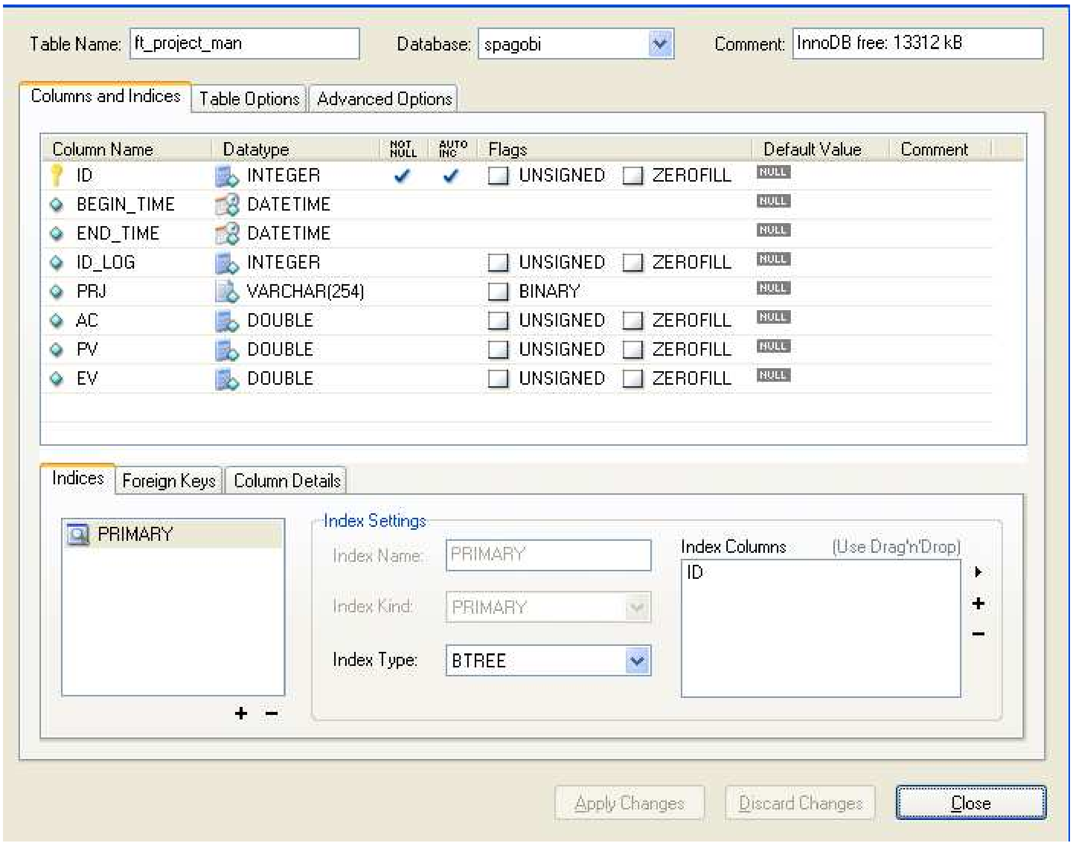


Table Name: ft_project_man
Database: spagobi
Comment: InnoDB free: 13312 kB

Column Name	Datatype	NOT NULL	AUTO INC	Flags		Default Value	Comment
ID	INTEGER	✓	✓	☐ UNSIGNED	☐ ZEROFILL	NULL	
BEGIN_TIME	DATETIME					NULL	
END_TIME	DATETIME					NULL	
ID_LOG	INTEGER			☐ UNSIGNED	☐ ZEROFILL	NULL	
PRJ	VARCHAR(254)			☐ BINARY		NULL	
AC	DOUBLE			☐ UNSIGNED	☐ ZEROFILL	NULL	
PV	DOUBLE			☐ UNSIGNED	☐ ZEROFILL	NULL	
EV	DOUBLE			☐ UNSIGNED	☐ ZEROFILL	NULL	

Index Name: PRIMARY
Index Kind: PRIMARY
Index Type: BTREE
Index Columns (Use Drag'n'Drop): ID

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

If you want to add or modify the fields of an interface type you have to delete the fact table and recreate it. To Delete a fact table click the "Drop Fact Table" button ☒ (all the data in the fact table will be erased).

3.3 HOW TO DEFINE THE EXTRACTION AND TRANSFORMATION PROCESS

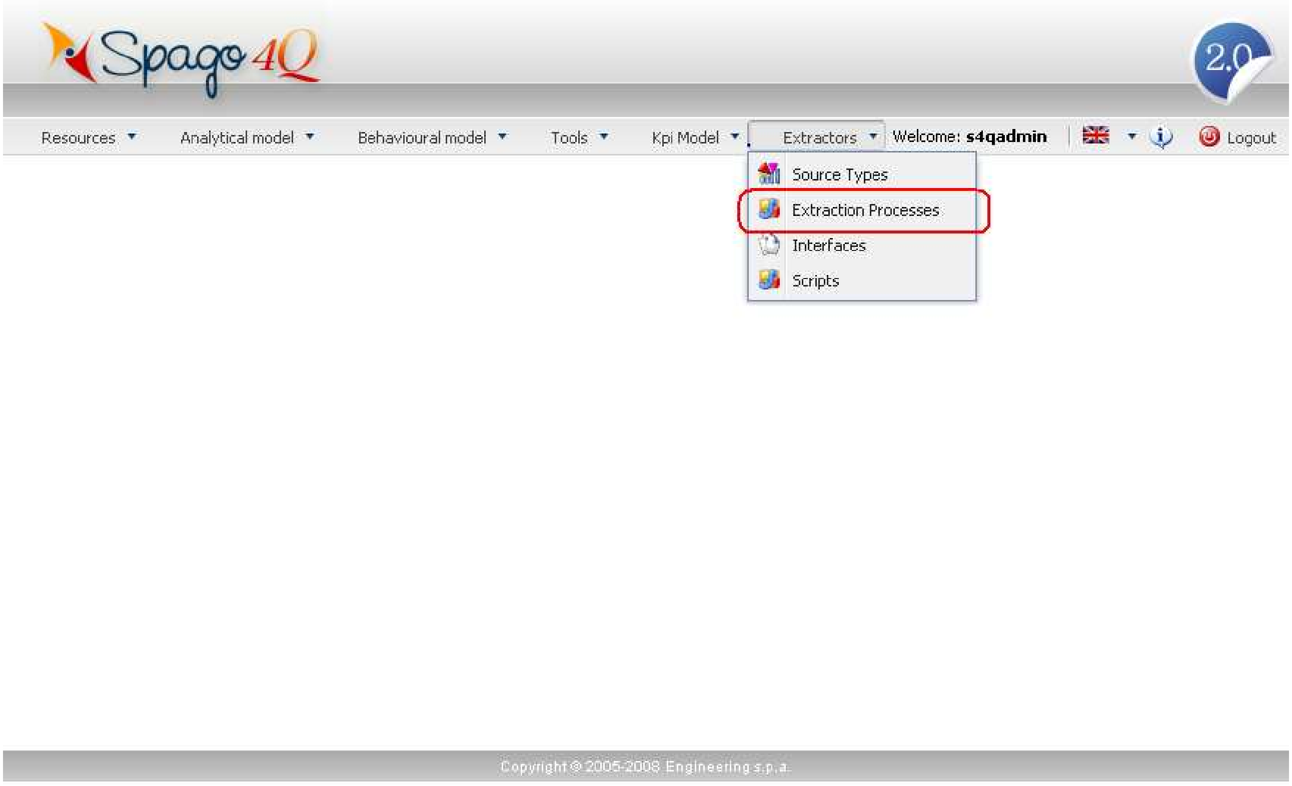
An extraction process is a set of operations. Every operation extracts, transforms, and loads data from a data source to a fact table. To define an extraction process you have to do this operations:

- Define an extraction process.
- Define and associate same operations to the extraction process.
- Define and associate same parameter to an operation.
- Map the data extracted from the data source to the column of the fact table.
- Execute the extraction process.

3.3.1 DEFINE AN EXTRACTION PROCESS

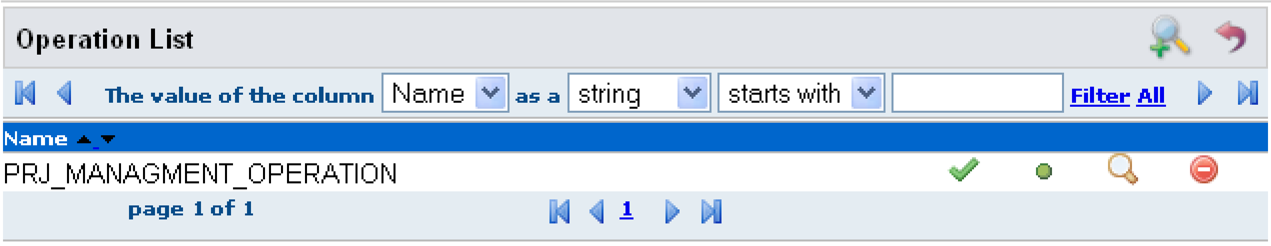
To create an Extraction Process you have to do this operations:

- Click on the menu item "Extraction Process" and you will enter into the list of existing extraction processes.



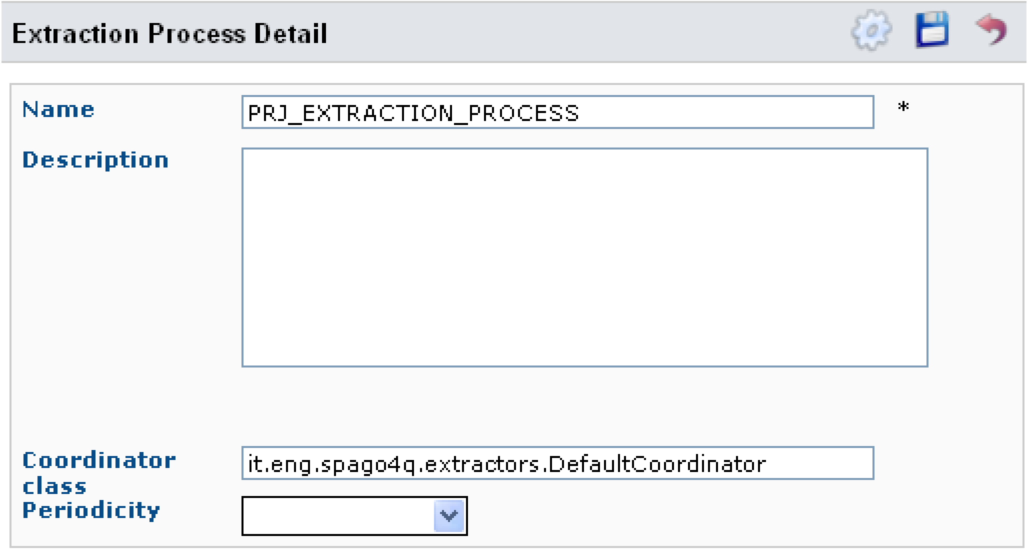
Click on the "View Operations" button ✓ and you will enter into the list of existing Operations associated to a specific Extraction Process:

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



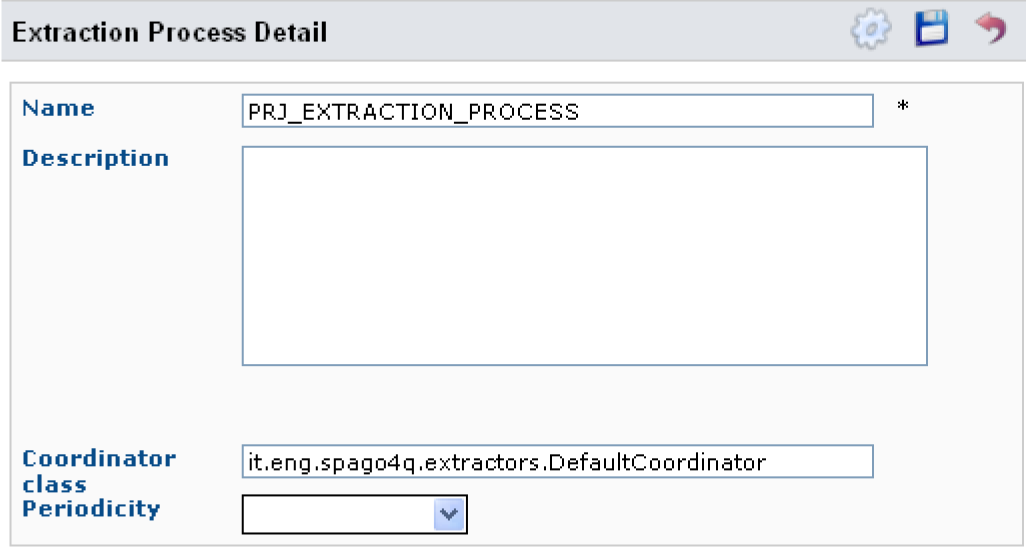
PRJ_MANAGMENT_OPERATION

In order to create a new Extraction Process click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Extraction Process Detail

Name	PRJ_EXTRACTION_PROCESS *
Description	
Coordinator class	it.eng.spago4q.extractors.DefaultCoordinator
Periodicity	



Extraction Process Detail

Name	PRJ_EXTRACTION_PROCESS *
Description	
Coordinator class	it.eng.spago4q.extractors.DefaultCoordinator
Periodicity	

In this form you can insert this fields:

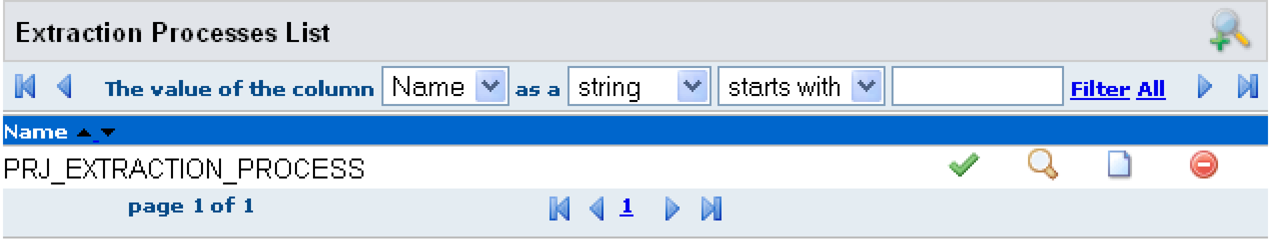
- **Name** (mandatory): it is the label of the Extraction Process and it identifies it.

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

- **Description**: describe the Extraction Process
- **Coordinator class**: Specify the Java Class that performs the extraction operations. By default is it.eng.spago4q.extractors.DefaultCoordinator.
- **Periodicity**: In this field you can select the frequency when the extraction process will be fired. If you select a periodicity automatically the extraction process will be scheduled. If you want to remove the extraction process from the scheduler set this field to empty.

Enter the required information and click on the “Save” button. Than click the “Back” button and you will see the new Extraction Process just inserted in Extraction Processes list.



PRJ_EXTRACTION_PROCESS

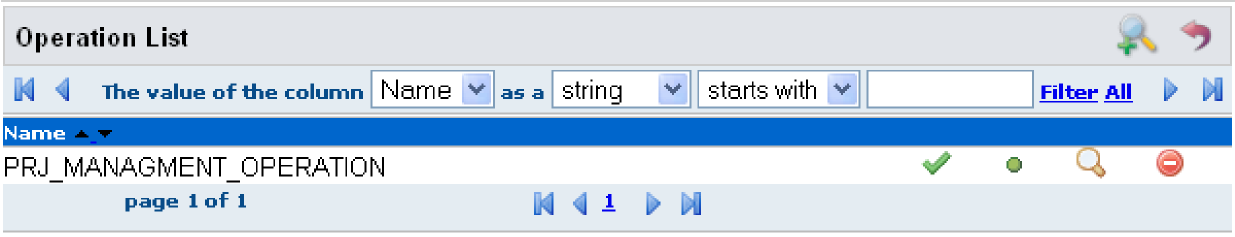
3.3.2 DEFINE AND ASSOCIATE SAME OPERATIONS TO THE EXTRACTION PROCESS

An Extraction Process will be executed same operations. In an operation you can specify:

- What you can extract from a Data source
- How to map and transform the data extracted from a Data Source in an Interface Type (Fact table)
-

In this guide will be created an Operation that perform a query in a Data Base (defined in a Data Source) and map the value extracted to the fact table (defined in a Interface Type). The first thing to do is create and associate an operation to a Extraction process. To create and associate an Operation to an extraction process you have to follow this steps:

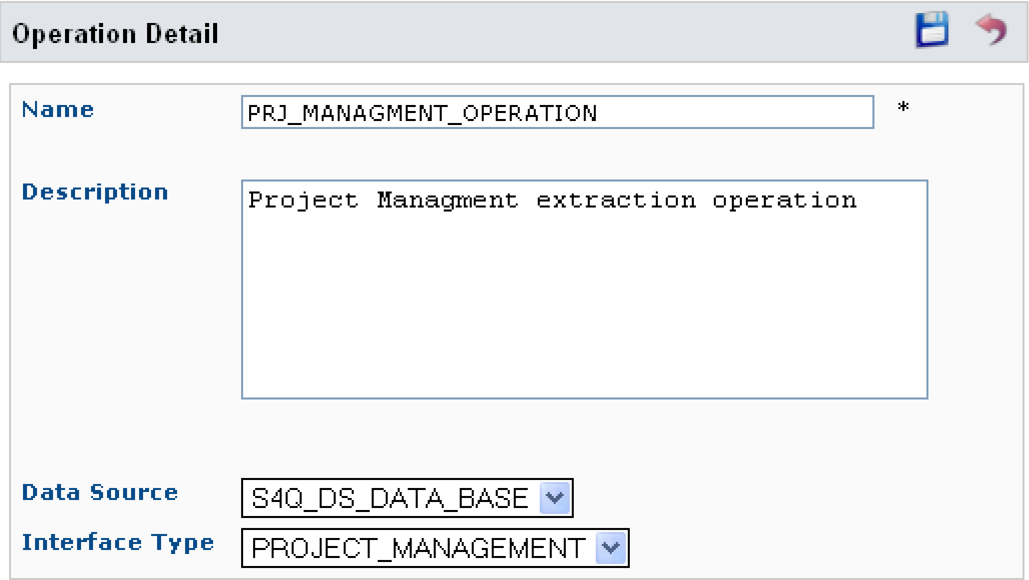
- click on the menu item “Extraction Process” and enter into the list of existing extraction processes.
- click on the ✓ to enter the operations list for that extraction process



PRJ_MANAGMENT_OPERATION

In order to create a new operation click on the “Insert” button in the top right corner of the page and you will enter in a form like the following one:

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Name: PRJ_MANAGMENT_OPERATION *

Description: Project Managment extraction operation

Data Source: S4Q_DS_DATA_BASE

Interface Type: PROJECT_MANAGEMENT

In this form you can insert this fields:

- **Name** (mandatory): it is the label of the Operation and it identifies it.
- **Description**: describe the operation.
- **Data Source**(mandatory): Using this combo you can associate a Data Source to the Operation.
- **Interface Type** (mandatory): Using this combo you can associate an Interface type to the Operation.

In this example the operation named PRJ_MANAGMENT_OPERATION will extract from the S4Q_DS_DATA_BASE and put the data to the fact table described from the PROJECT_MANAGMENT Interface Type.

Enter the required information and click on the "Save" button . Than click the "Back" button and you will see the new Operation just inserted in Operations list.

3.3.3 DEFINE AND ASSOCIATE SAME PARAMETERS TO AN OPERATION

To define what the operation has to extract you have to define same parameters that describe what the extractor class (Defined in Source Type and execute from the operation) has to return. The extractor class described in this guide in section X needs this operation parameters:

Extractor class	it.eng.spago4q.extractors.db.DbExtractor	This class extracts data from a Database using a SQL-92 query
Operation Parameters	QUERY	The query to do when spago4Q execute an extraction

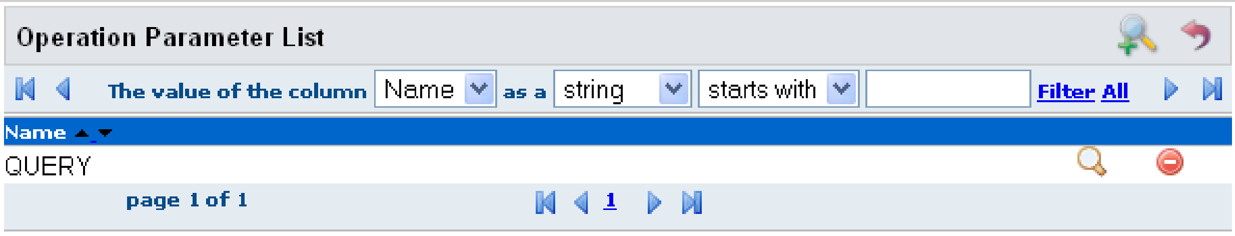
To create an Operation Parameter you have to do this operations:

1. Click on the menu item "Extraction Process" and enter into the list of existing extraction processes.

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

2. Click on the "View Operations" button ✓ and you will enter into the list of existing Operations associated to a specific Extraction Process.

Click on the "View Operation Parameters" button ✓ and you will enter into the list of existing Operation Parameters associated to a specific Operation.

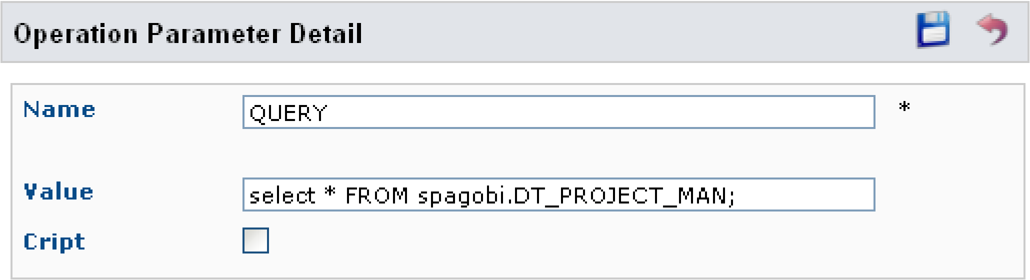

Operation Parameter List

The value of the column Name as a string starts with Filter All

Name
QUERY

page 1 of 1

In order to create a new operation parameter click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:


Operation Parameter Detail

Name	QUERY *
Value	select * FROM spagobi.DT_PROJECT_MAN;
Cript	☐

In this form you can insert this fields:

- **Name** (mandatory): it is the label of the Operation Parameter and it identifies it. This name will be defined by the documentation of the extractor class. (in this example the name will be QUERY)
- **Value**: specify the value of the operation parameter.
- **Cript**: if it is checked the data will be saved cripted in the Data Base.

In this example the parameter define a query that select all the field from a table named DT_PROJECT_MAN in the Database defined by the Data Source named S4Q_DS_DATA_BASE.

Enter the required information and click on the "Save" button . Than click the "Back" button and you will see the new Operation Parameter just inserted in Operation Parameters list.

3.3.4 MAP THE DATA EXTRACTED FROM THE DATA SOURCE TO THE COLUMN OF THE FACT TABLE

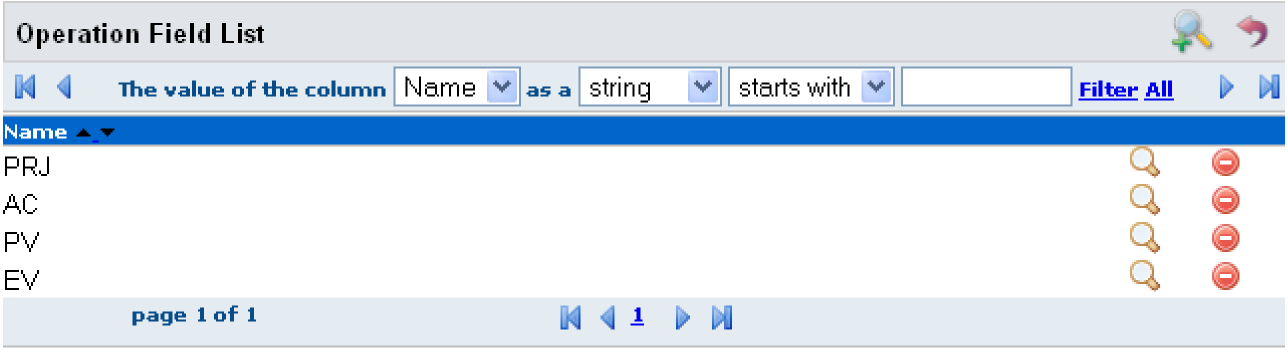
When an operation is executed it produced same data that will be insert in the fact table defined in the Interface Type. To Map the data extracted to the field of an Interface Type you have to do this operations:

3. Click on the menu item "Extraction Process" and enter into the list of existing extraction processes.
4. Click on the "View Operations" button ✓ and you will enter into the list of existing Operations associated to a specific Extraction Process.

Click on the "View Operation Field" button ● and you will enter into the list of existing Operation Field associated to a specific Operation.

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Operation Field List

The value of the column Name as a string starts with

Name

PRJ

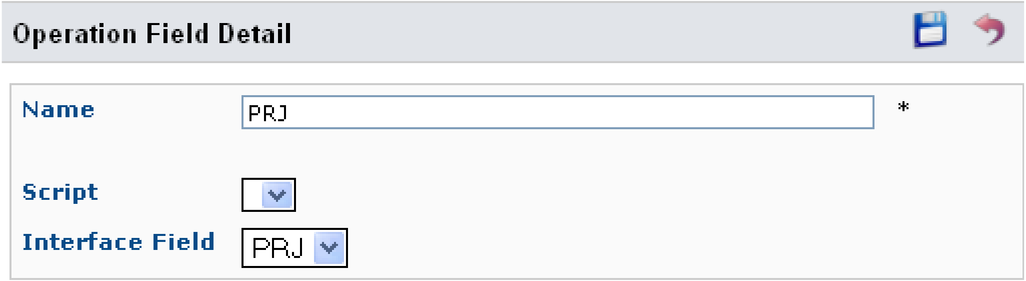
AC

PV

EV

page 1 of 1

In order to create a new operation field click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Operation Field Detail

Name: PRJ *

Script:

Interface Field: PRJ

In this form you can insert this fields:

- **Name** (mandatory): it is the label of the field extracted.
- **Script**: This combo define a script that has the ability to transform a data extract from the data source before the insertion in the fact table.
- **Interface Field**: it is the field of the Interface Type (the column of a Fact Table) where to put the data extracted from the operation.

Enter the required information and click on the "Save" button. Than click the "Back" button and you will see the new Operation Field just inserted in Operation Parameters list.

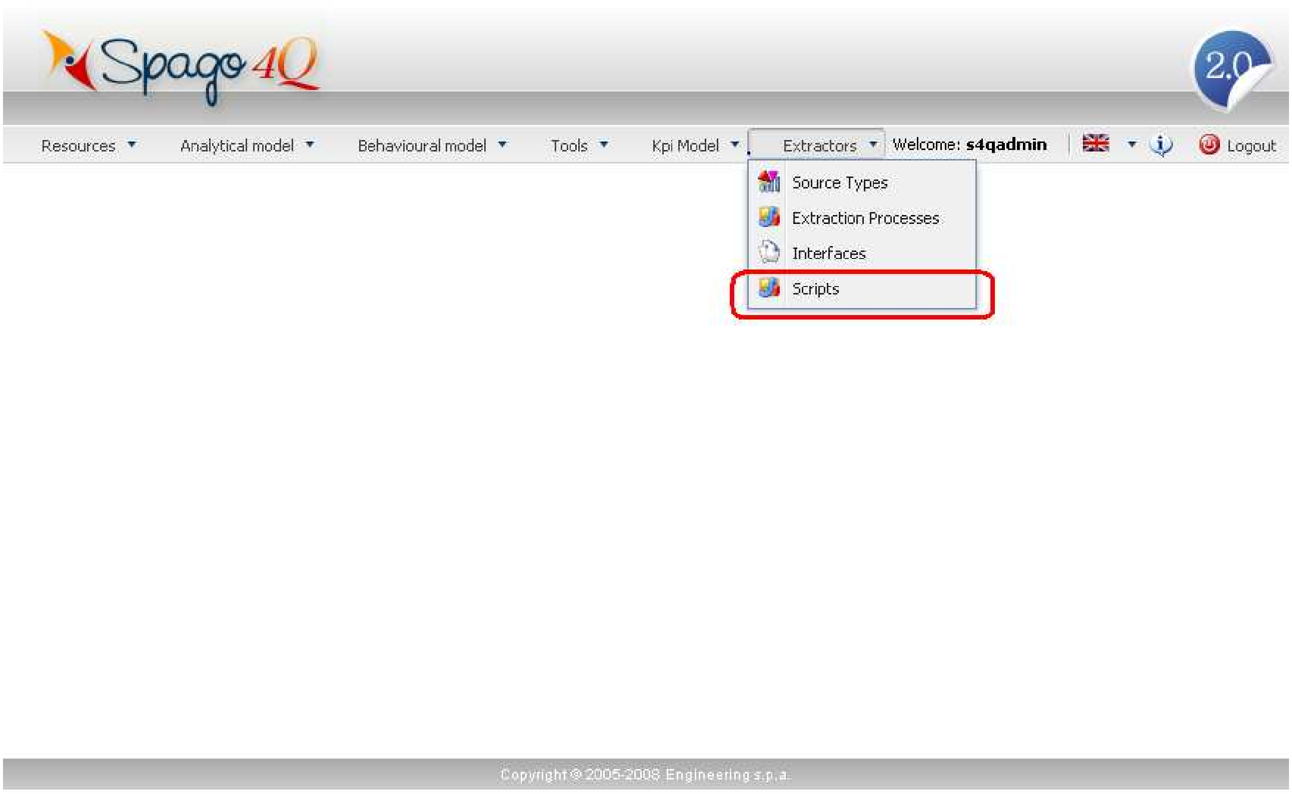
3.3.5 CREATE A TRANSFORMATION SCRIPT

Sometime the data produced from the extraction process are incompatible with the format of the fact table. Sometime the data produce from the extraction process has to be transform before the insertion in the fact table. To perform this operations Spago4q has the capability to define a script and associated it to an operations field (as explained before). To create a script you have to do this operations:

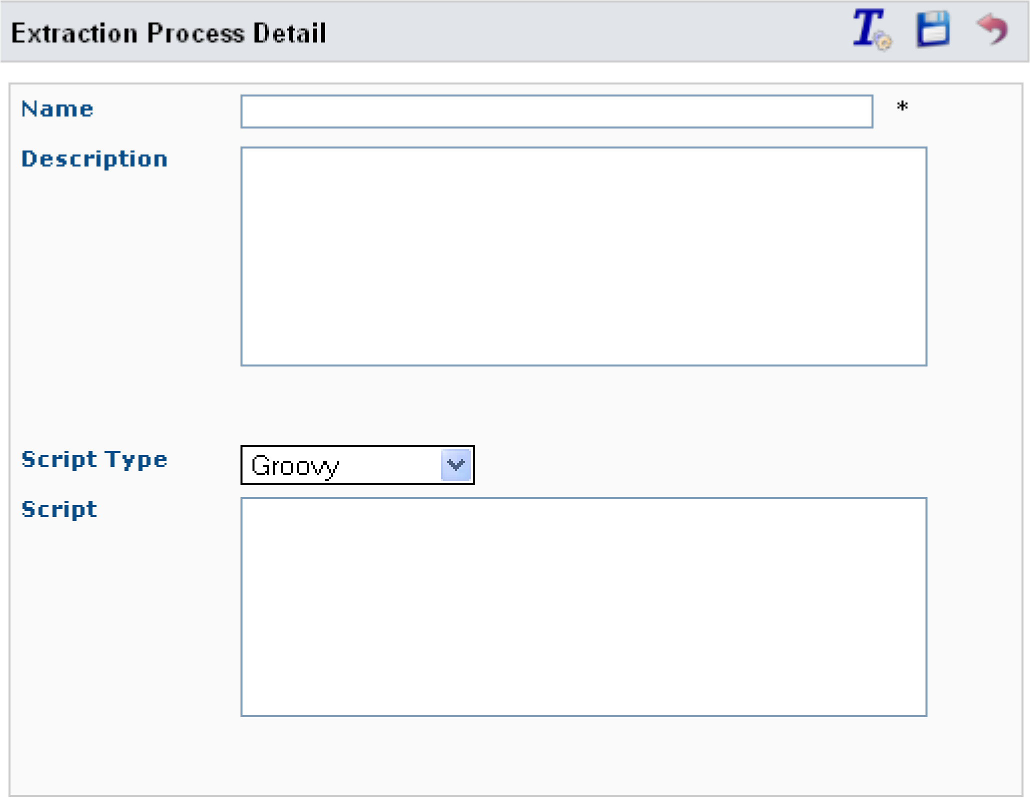
Click on the menu item "Script" and enter into the list of existing Script.

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



In order to create a new Script click on the "Insert" button in the top right corner of the page and you will enter in a form like the following one:



Name	*
Description	
Script Type	Groovy
Script	

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

In this form you can insert this fields:

- **Name** (mandatory): it is the label of the Script.
- **Description**: it describes the script
- **Script Type**: This combo is defined the type of the script:
    - **Groovy**: if you select this value you will insert a groovy script in the "Script" field. The groovy script will be implemented the "`it.eng.spago4q.extractors.script.IMappingScript`" interface.
    - **Date Converter**: If you select this value you will insert in the "Script" field the date format of the value extracted. This type of script is used to convert a date in the format compatible with the one used by the fact table.

For example if you want to convert a boolean value to 1 if the value is true and 0 otherwise you have to insert this value in the form:

name	BooleanConverter
description	Convert true to 1 0 oderwise
Script type	Groovy

Script

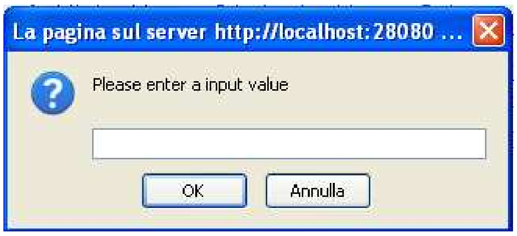
```
import it.eng.spago4q.extractors.script.IMappingScript;

public class Tester implements IMappingScript {

public String execute(String value) {
String toReturn = "0";
  if (value == "true")
    toReturn = "1";
  return toReturn;
  }
}

Object o = new Tester();
```

If you want to test the script you can click on "Save and Test" button . When you click this button should be appear a pop up like this

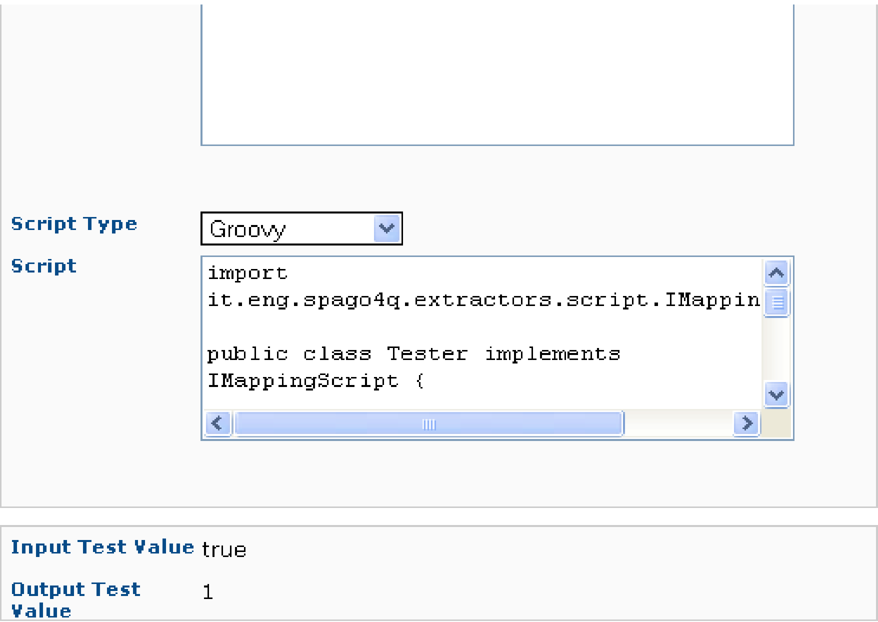


La pagina sul server http://localhost:28080 ...

Please enter a input value

Insert an input value (like "true") and click on the "OK" button. You should see at the bottom of the form this information:

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Script Type: Groovy

Script:

```
import
it.eng.spago4q.extractors.script.IMappin

public class Tester implements
IMappingScript {
```

Input Test Value: true

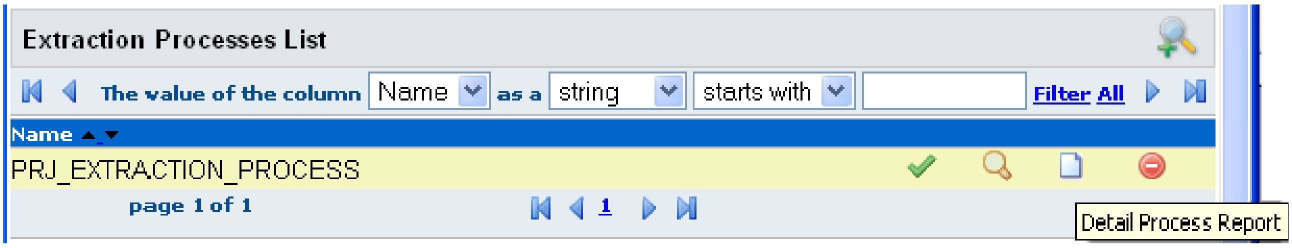
Output Test Value: 1

If the script produces the result that you expected click on the "Save" button . Than click the "Back" button and you will see the list of the scripts.

3.3.6 EXECUTE THE EXTRACTION PROCESS

When you've finished the configuration of an extraction process (Data Sources, Interface Type and Operations) you can execute it. Before the execution of the extraction process you can check the configuration using a report that summarize all the configuration explain this guide. To generate this report you have to do this operations:

Click on the menu item "Extraction Process" and enter into the list of existing extraction processes.



Extraction Processes List

The value of the column: Name, as a: string, starts with

Name
PRJ_EXTRACTION_PROCESS

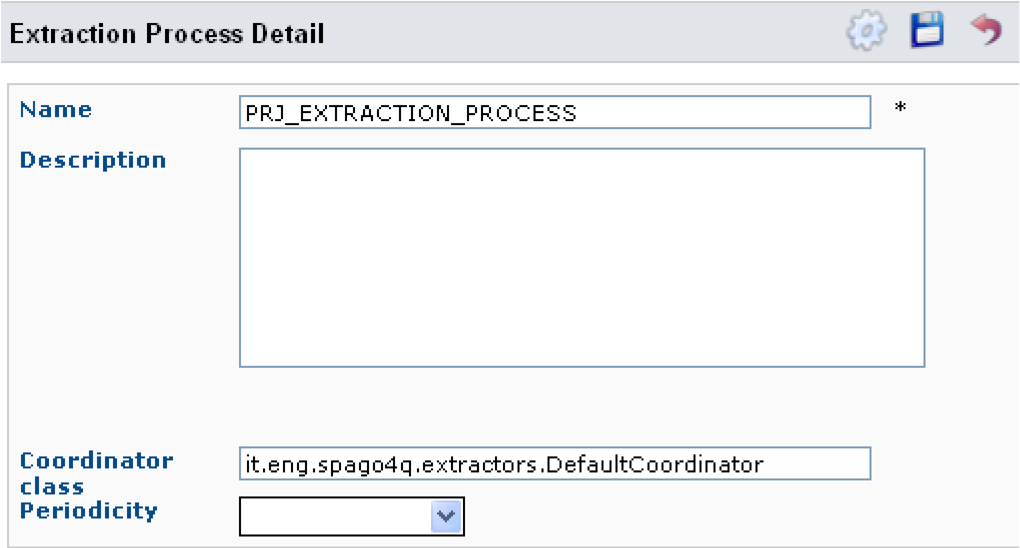
Click on the "Detail Process Report" button and you will sow a pdf report that summarize all the configuration of the extraction process.

If you want to execute the extraction process you have to do this operations:

Click the "Select" button of the Extraction Process that you want to execute an you will see the following form

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Extraction Process Detail

Name	PRJ_EXTRACTION_PROCESS *
Description	
Coordinator class	it.eng.spago4q.extractors.DefaultCoordinator
Periodicity	

Click on the "Save and Execute" button and the process will be execute. If the extraction process doesn't produce error you will see the data extracted in the fact tables associated to the operations of the extraction process.

If you want to schedule the process you have to do this operations:

5. Open the process detail by clicking the "Select" button of the Extraction Process that you want to execute.
6. Select a periodicity and the extraction process will be added to the scheduler.

3.3.7 CUSTOM EXTRACTOR CLASS

This part of the guide shows you how to develop your custom extractor based on the characteristics of the tool you want to extract data from. The explanation will follow the provided Database extractor. The code we are going to refer to is the one on the class:

```
it.eng.spago4q.extractors.db.DbExtractor
```

To develop a custom extractor you have to:

- Extend the abstract class `it.eng.spago4q.extractors.AbstractExtractor`
- Implement the methods (optional):
  - `protected void setUp();`
  - `protected void tearDown();`
- Implement the method (mandatory)
  - `protected List<GenericItemInterface> extract() throws EMFUserError;`

The methods setUp() and tearDown() are called before and after the extract() respectively, and they are useful to manage operations that are not strictly connected with the extraction itself but are needed to prepare and close the whole extraction operation. For the database extractor you can see that only the setUp() has been used to get the database connection parameters.

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Here, in the setUp() method, you can see the usage of a service method

String **readDataSourceParameterValue**(String key)

that allow you to read the data source parameters by Name (key). These parameters are the ones you can insert through the user interface for a specific data source configuration. For the Database extractor this is used to retrieve user, password, url and driver.

On the same setUp() method there is also an example of another service method

String **readOperationParameterValue**(String key)

that let you get the operation parameters values. These parameters are the ones you can insert through the user interface for a specific operation configuration. For the Database extractor this is used to retrieve the query that will be executed for the data extraction.

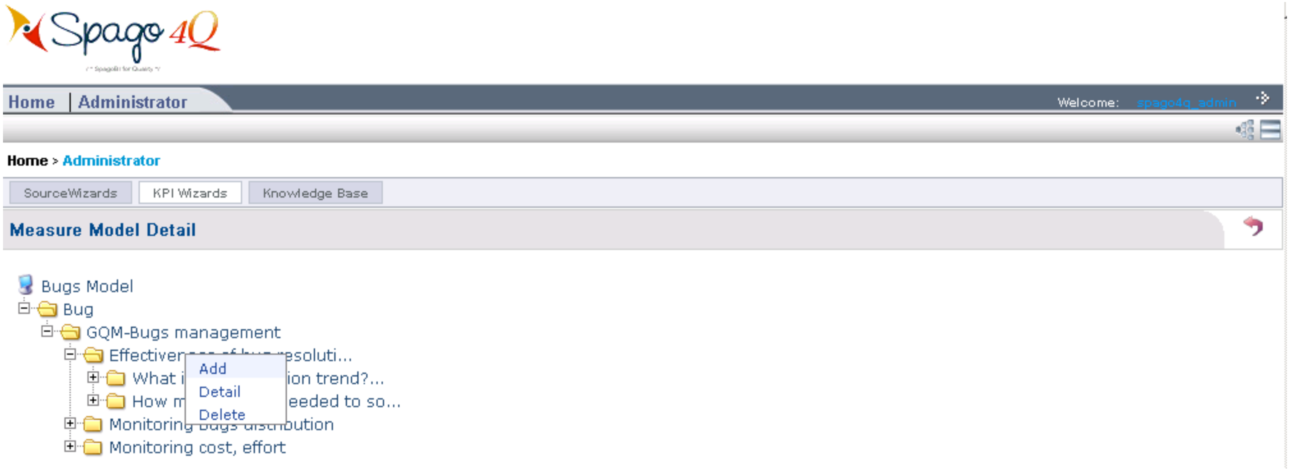
The main method to implement is the **extract()** where the main task is to build up a list of items that are the information extracted from the tool. Every single item of this list is a **GenericItem** Java Object that is essentially an HashMap with both name (key) and value of String type.
Once the list is built inside the extract() method the work is done, from the extraction side. At this point the list extracted will be managed by the operation coordinator that is in charge to map all the GenericItems into the items of the specific interface (**InterfaceItem**), used for the extraction operation. This list of InterfaceItems will be loaded into the appropriate FT_ table.

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

4 How to define models

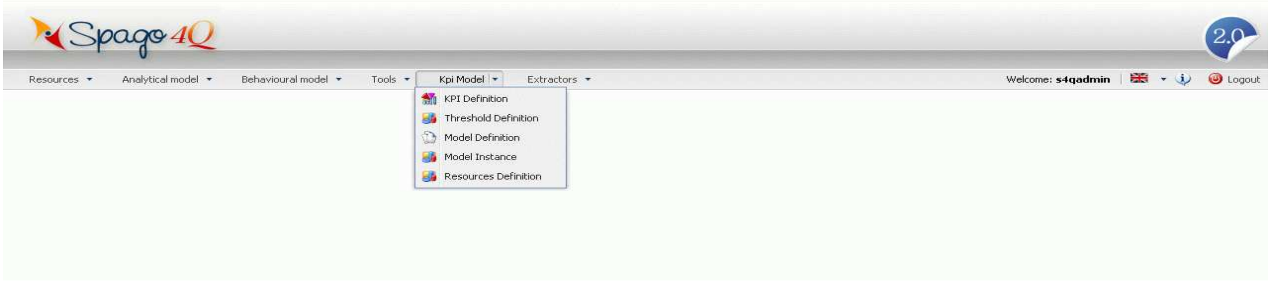
“Model Definition” wizards lets you define a generic model composed by a hierarchy of levels and by elements associated to each level: an element defined at an i level is associated to only one element at level i-1 and zero or more elements at the i+1 level.

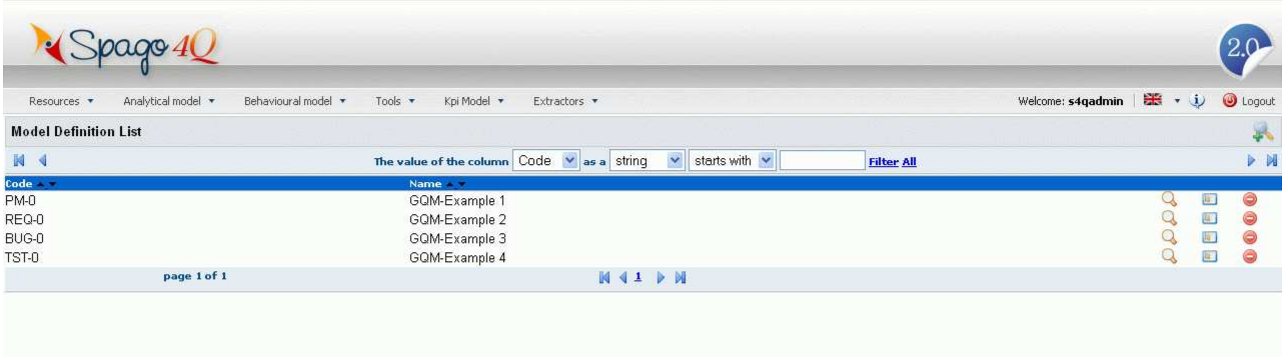
Using this feature you can create a GQM model as showed in next figure



In the same way you can create an Assessment Models, based on CMMI or ISO9001-2000 or any other assessment framework.

To create a new model click on the menu item “KPI Model” and on “Model definition”, you enter into the list of existing model.



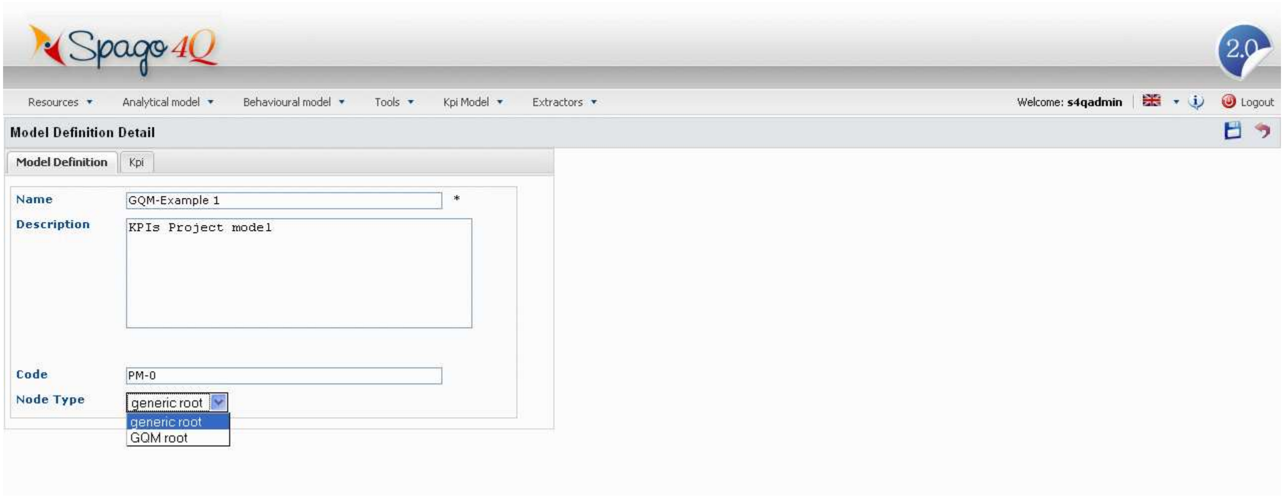


Model Definition List

Code	Name
PM-0	GQM-Example 1
REQ-0	GQM-Example 2
BUG-0	GQM-Example 3
TST-0	GQM-Example 4

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Click on to insert a new model, you will enter in a form like the following one:



Name: GQM-Example 1 *

Description: KPIs Project model

Code: PM-0

Node Type: generic root

- generic root
- GQM root

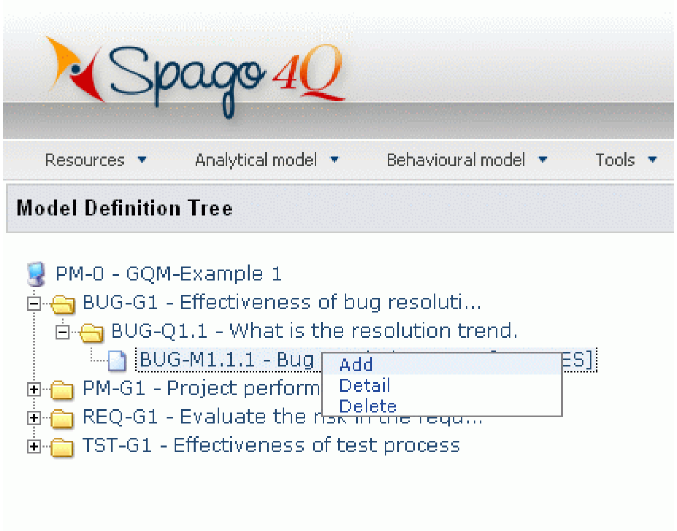
In this form you can insert these fields:

- **Name**: it's the label of the model
- **Description**: it describes the model
- **Code**: you can assign a mnemonic code that it will be displayed in the tree view.
- Node Type: The first node is created assigning a node type "root".

Here you can also link a kpi to the node of the model, or you can perform this relationship when you create a model instance.
Click on to "kpi" tab to link a kpi to the node of the model. You can select ones from the list of existing kpi or you have to create ones before using "Kpi Definition" wizard.

Click on to save the model.

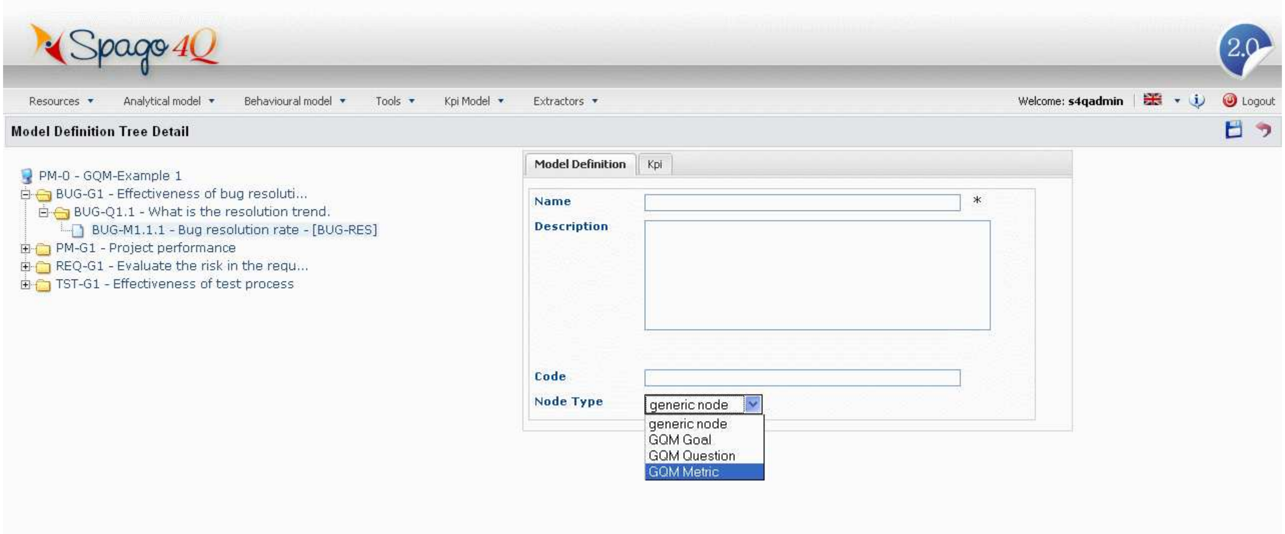
Click on to go back to the list and click on to switch from the list view to the tree view.


Spago4Q
Resources
Analytical model
Behavioural model
Tools
Model Definition Tree
PM-0 - GQM-Example 1
BUG-G1 - Effectiveness of bug resoluti...
BUG-Q1.1 - What is the resolution trend.
BUG-M1.1.1 - Bug
ES]
PM-G1 - Project perform
REQ-G1 - Evaluate the risk in the requ...
TST-G1 - Effectiveness of test process
Add
Detail
Delete

creative commons
COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Click on a node and click "Add" to create a new node, you enter in a form as the following:


Spago4Q
2.0
Resources
Analytical model
Behavioural model
Tools
Kpi Model
Extractors
Welcome: s4qadmin
Logout
Model Definition Tree Detail
PM-0 - GQM-Example 1
BUG-G1 - Effectiveness of bug resoluti...
BUG-Q1.1 - What is the resolution trend.
BUG-M1.1.1 - Bug resolution rate - [BUG-RES]
PM-G1 - Project performance
REQ-G1 - Evaluate the risk in the requ...
TST-G1 - Effectiveness of test process
Model Definition
Kpi
Name
*
Description
Code
Node Type
generic node
generic node
GQM Goal
GQM Question
GQM Metric

In this form you can insert these fields:

- **Name**: it's the label of the node
- **Description**: it describes the node
- **Code**: you can assign a mnemonic code that it will be displayed in the tree view.
- **Node Type**: you can select a type node following for example a GQM schema.

Here you can also link a kpi to the node of the node, or you can perform this relationship when you create a model instance.
Click on to "kpi" tab to link a kpi to the node of the model. You can select ones from the list of existing kpi or you have to create ones before using "Kpi Definition" wizard.

Click on to save the model.

Click on to go back or click on a node of the tree to add new nodes.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

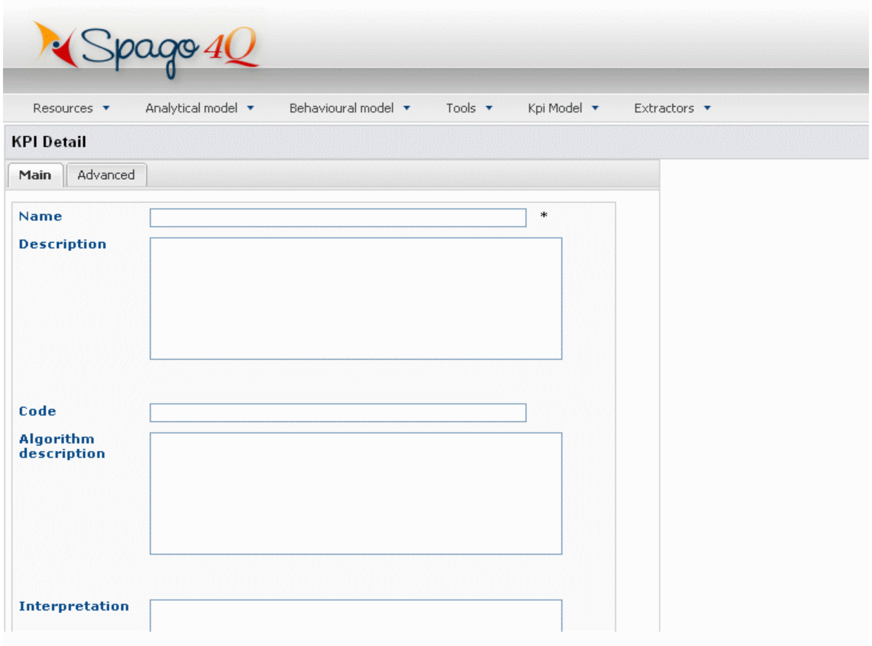
5 How to define KPIs and thresholds

KPI and threshold wizard lets you detail all the parameters and values to calculate and represent the metric.

“**KPI Definition”** wizards lets you define a new KPI.

To create a new KPI click on the menu item “KPI Model” and on “KPI definition”, you enter into the list of existing KPI.

Click on to insert a new KPI, you will enter in a form like the following one:



In this form – Main tab - you can insert these fields:

- **Name**: it’s the label of the kpi
- **Description**: it describes the kpi
- **Code**: you can assign a mnemonic code. **Name** and **Code** will be displayed where you have to select a KPI relation. Code is also displayed in the tree view of the models.
- **Algorithm**: it’s a free textual description. It describes the algorithm needed to the computation of the KPI. This algorithm will be implemented into the “Dataset”.
- **Interpretation**: it describes the KPI interpretation criteria
- **Weight default value:** You can assign a weight to manage kpi aggregation. You could create a kpi for a node i as weighted media of the kpi of the nodes i-1.
- **Threshold Name:** You can select ones from the list of existing thresholds or you have to create ones before using “thresholds Definition” wizard.
- **Document Label** : you can define an analytical document, for the KPI, that can be accessed directly from the KPI dashboard.
- **Calculation Rule (“Data Set”):** you can define the proper computation element to use for the KPI.

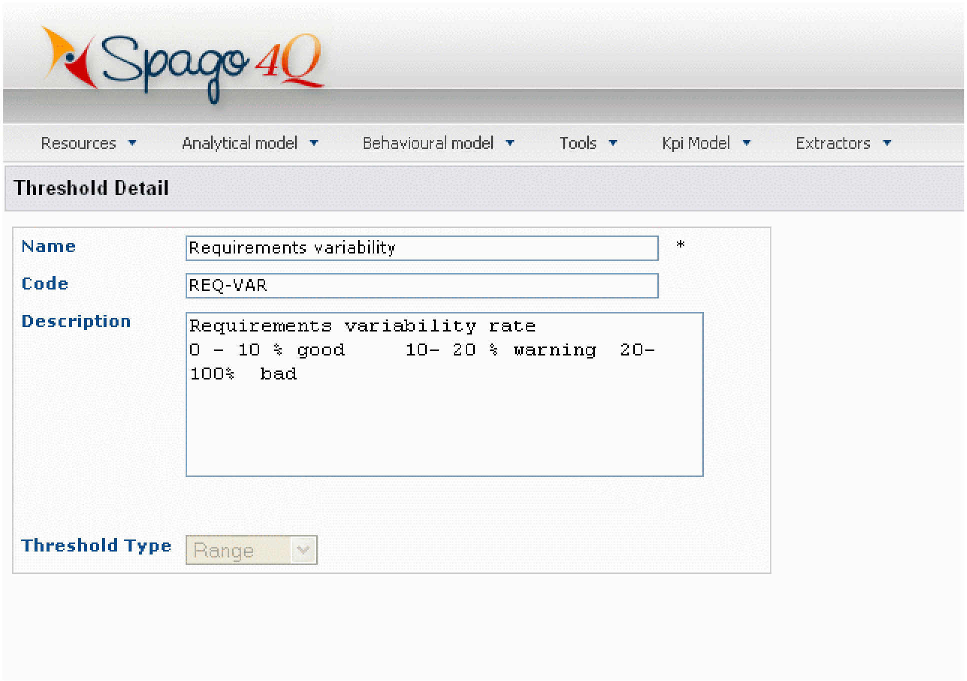
creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/
BY:

The **Advanced** tab let you define more structured information for better describe the KPI under definition.

“Thresholds Definition” wizards lets you define a new KPI.

To create a new Threshold click on the menu item “KPI Model” and on “Threshold definition”, you enter into the list of existing Threshold. The list lets the user insert new ones or update or delete the current ones. Thresholds that are referred from at least one KPI cannot be deleted.

Click on to insert a new Threshold, you will enter in a form like the following one:



Threshold Detail

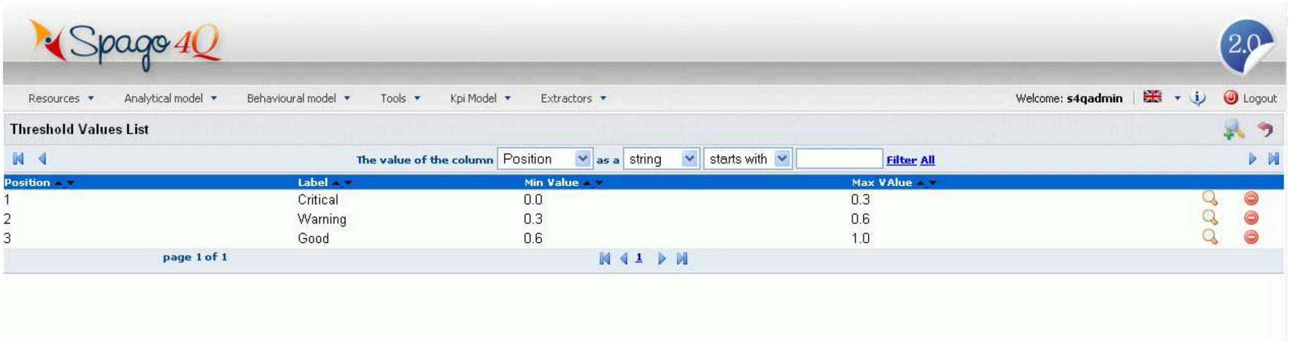
Name	Requirements variability	*
Code	REQ-VAR	
Description	Requirements variability rate 0 - 10 % good 10- 20 % warning 20- 100% bad	
Threshold Type	Range	

In this form you can insert these fields:

- **Name**: it’s the label of the threshold
- **Code**: you can assign a mnemonic code. **Name** and **Code** will be displayed where you have to select a threshold relation. Code is also displayed in the tree view of the models.
- **Description**: it describes the threshold
- **Threshold type**: it define the type of the values associated.

From the list of thresholds you can select a threshold and view the values associated, add new values and update or delete the current ones. A threshold value is characterised by a minimum and a maximum value that bound its range interval, a label and a colour.
The values associated to a specific threshold are an ordered list, in particular they are treated as a LIFO list, so that the user can delete only the last value inserted; this choice has been made to preserve a strict ordination among values.

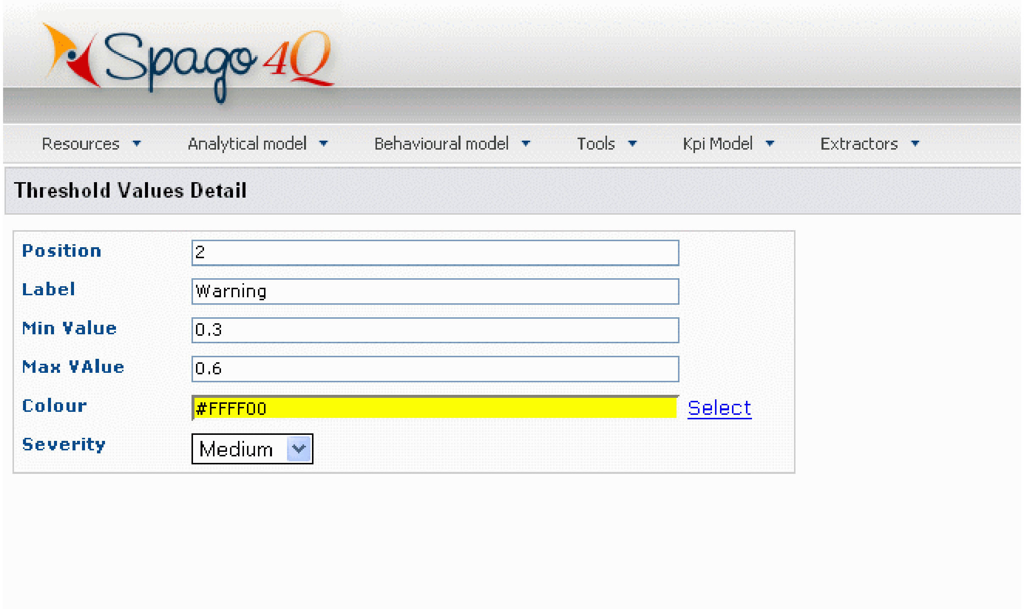
creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Threshold Values List

Position	Label	Min Value	Max VAlue
1	Critical	0.0	0.3
2	Warning	0.3	0.6
3	Good	0.6	1.0

Click on 🔍 to insert new values or click on 🔍 to modify the values.



Threshold Values Detail

Position	2
Label	Warning
Min Value	0.3
Max VAlue	0.6
Colour	#FFFF00
Severity	Medium

In this form you can insert these fields:

- **Position**: it defines the sequence
- **Label**: it's the label of the range of value
- **Min Value**: it defines the minimum value of the range
- **Max value**: it defines the maximum value of the range
- **Colour**: it defines the colour of this range in the dashboard graphical representation
- **Severity**: it's used to manage alarms.

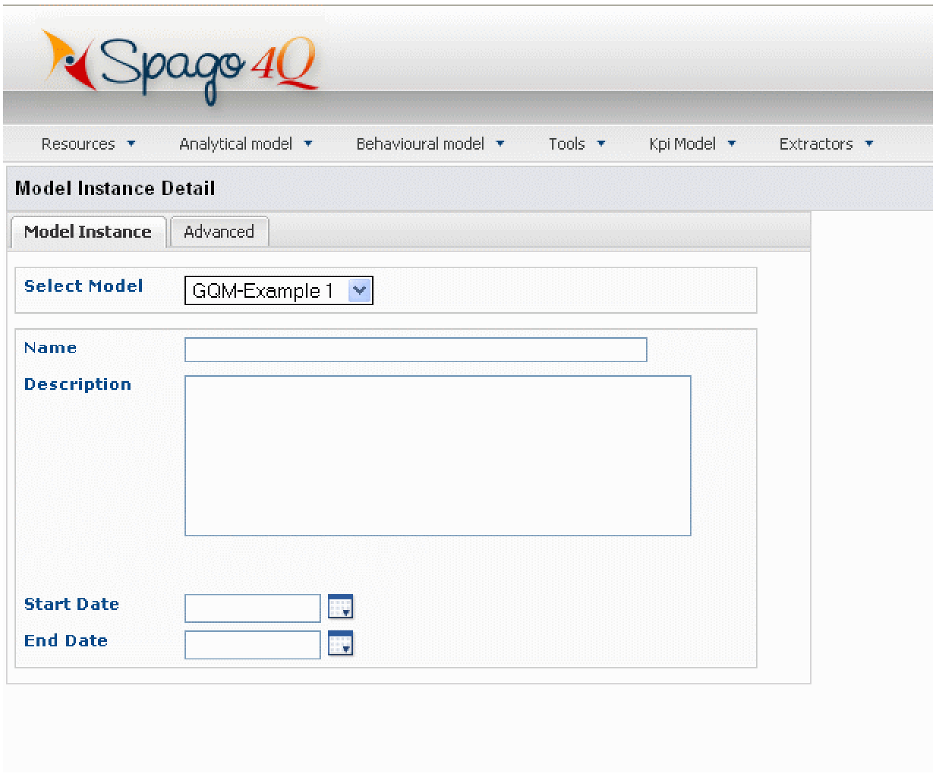
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

6 How to define model instances

“Model Instances” wizards lets you define a relationship between a source model - defined using “model definition wizards”- and kpi, thresholds and resources as, for example, projects or products. The kpi dashboard represents the kpi values computed using the measure attributes collected from the specific project or product repositories.

To create a new model instance click on the menu item “KPI Model” and on “Model instance”, you enter into the list of existing model instances.

Click on to insert a new model, you will enter in a form like the following one:



Select Model: GQM-Example 1

Name

Description

Start Date

End Date

In this form you can select the source model and insert these fields:

- **Name**: it’s the label of the model instance
- **Description**: it describes the model instance
- **Start and end Date**: they define the validity period of the model instance.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

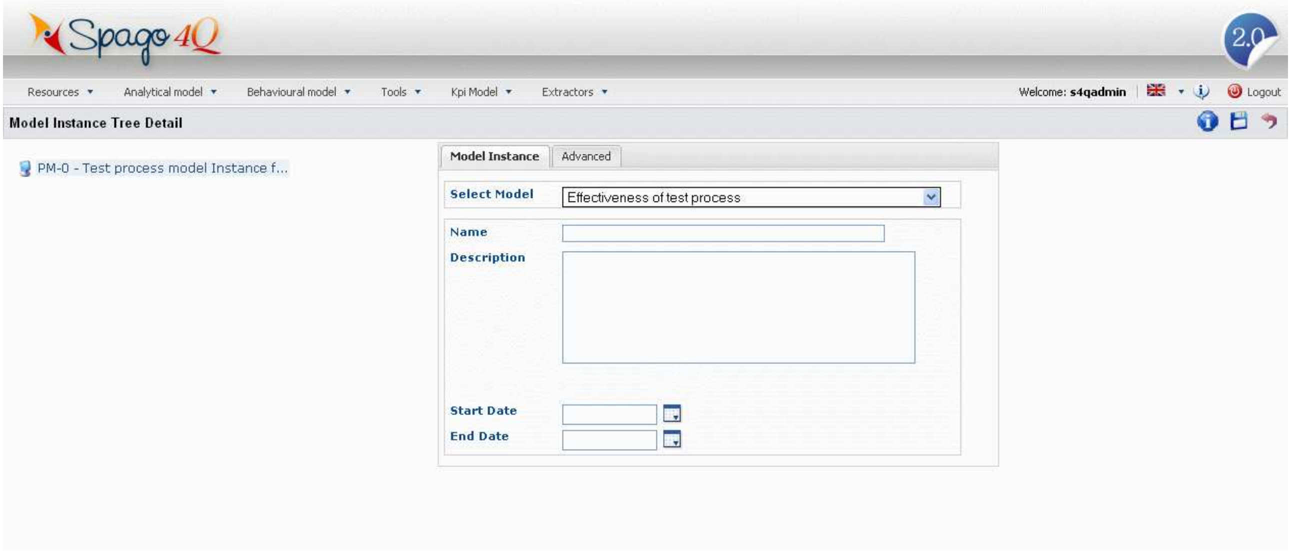
Click on to save the model instance.

Click on to go back to the list and click on to switch from the list view to the tree view.

By selecting a node at *i* level on the tree and by left-clicking on it you can:

1) Add a node at the level i+1 among the available elements, that are the elements associated to the selected node (for example the question node of a specific goal node). If there are no more elements available, for example because all have been already associated, the system displays an information message.
2) View the detail
3) Delete the node and the sub-tree of which it is the root.

When you click "Add" to create a new node, you enter in a form as the following:



PM-0 - Test process model Instance f...

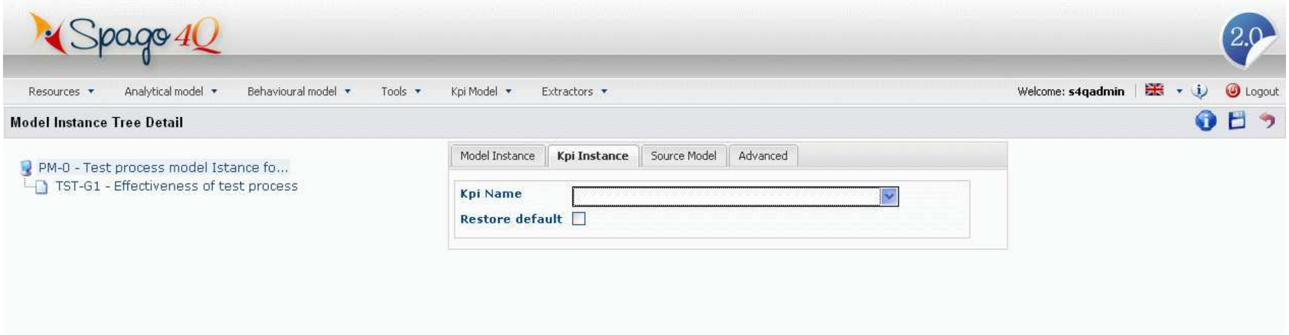
Select Model: Effectiveness of test process

Select model: you can select nodes from the source model
When you add a new node you have to select which one to add from the combo box. If you leave the **Name** filed blank, the name of the source node will be used.

The - Advanced tab - automatically provide the "Model label Instance". You need this label to create the "KPI document" (chapter 10).

Click on to update the model instance.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



PM-0 - Test process model Istance fo...
TST-G1 - Effectiveness of test process

Kpi Name

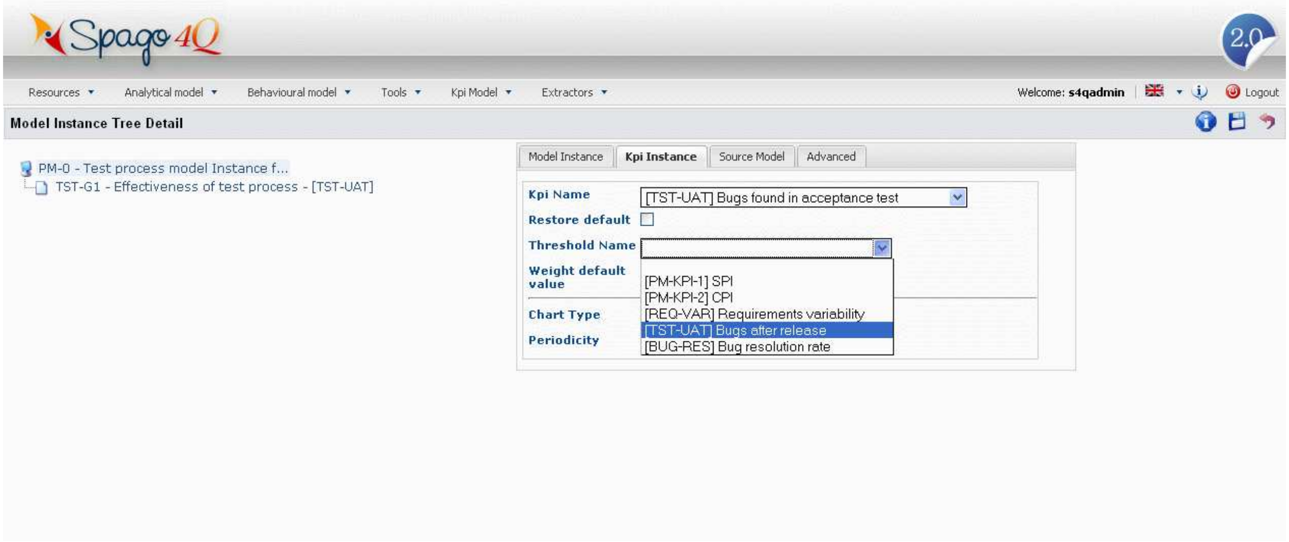
Restore default ☐

Node Detail

Open the node detail using the contextual menu from the tree. All the information are grouped in tabs. Go to the KPI tab to update the configuration for this KPI instance.

Kpi Name: You can select a kpi or you can select **Restore Default** to resume the kpi already linked to the source model.

Click on to update the model instance, you will enter in a form like the following one:



PM-0 - Test process model Instance f...
TST-G1 - Effectiveness of test process - [TST-UAT]

Kpi Name [TST-UAT] Bugs found in acceptance test

Restore default ☐

Threshold Name

Weight default value

Chart Type

Periodicity

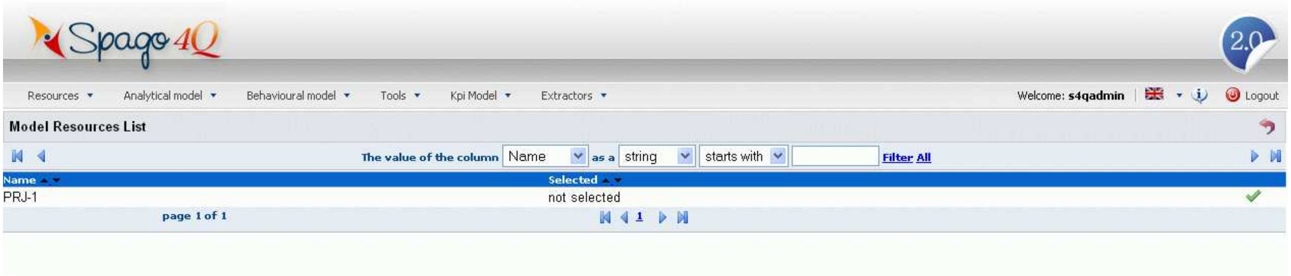
[PM-KPI-1] SPI
[PM-KPI-2] CPI
[REQ-VAR] Requirements variability
[TST-UAT] Bugs after release
[BUG-RES] Bug resolution rate

You can complete the kpi instance updating the resumed values or selecting a new threshold.

Repeat the previous steps to complete the model instance.

Click on to go back to the list of model instance then click on the "Star" button to create a relationship between the model instance and the resources (as projects or products).
In this form you can select or deselect the resources.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/
BY:



Name	Selected
PRJ-1	not selected

page 1 of 1

To create a new resource click on the menu item "KPI Model" and on "Resource definition".

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

7 How to create a Dataset for KPI computation

Every KPI defined uses a Dataset to execute the computation of its value. This execution could depend on a couple of parameters where one id based on the generic resource, that the KPI is computed on, and the other is the date on which the computation is performed.
The possible values returned by the Dataset will have to have the following aliases:

VALUE	The value of the KPI	Mandatory
DESC	The description of the KPI Value	Optional
END_DATE	The validity end date that override the one calculated form the periodicity of the KPI Format: dd/MM/yyyy hh:mm:ss	Optional

Moreover, you have the possibility to insert some parameters in your DataSet. The standard parameters that you can add are:

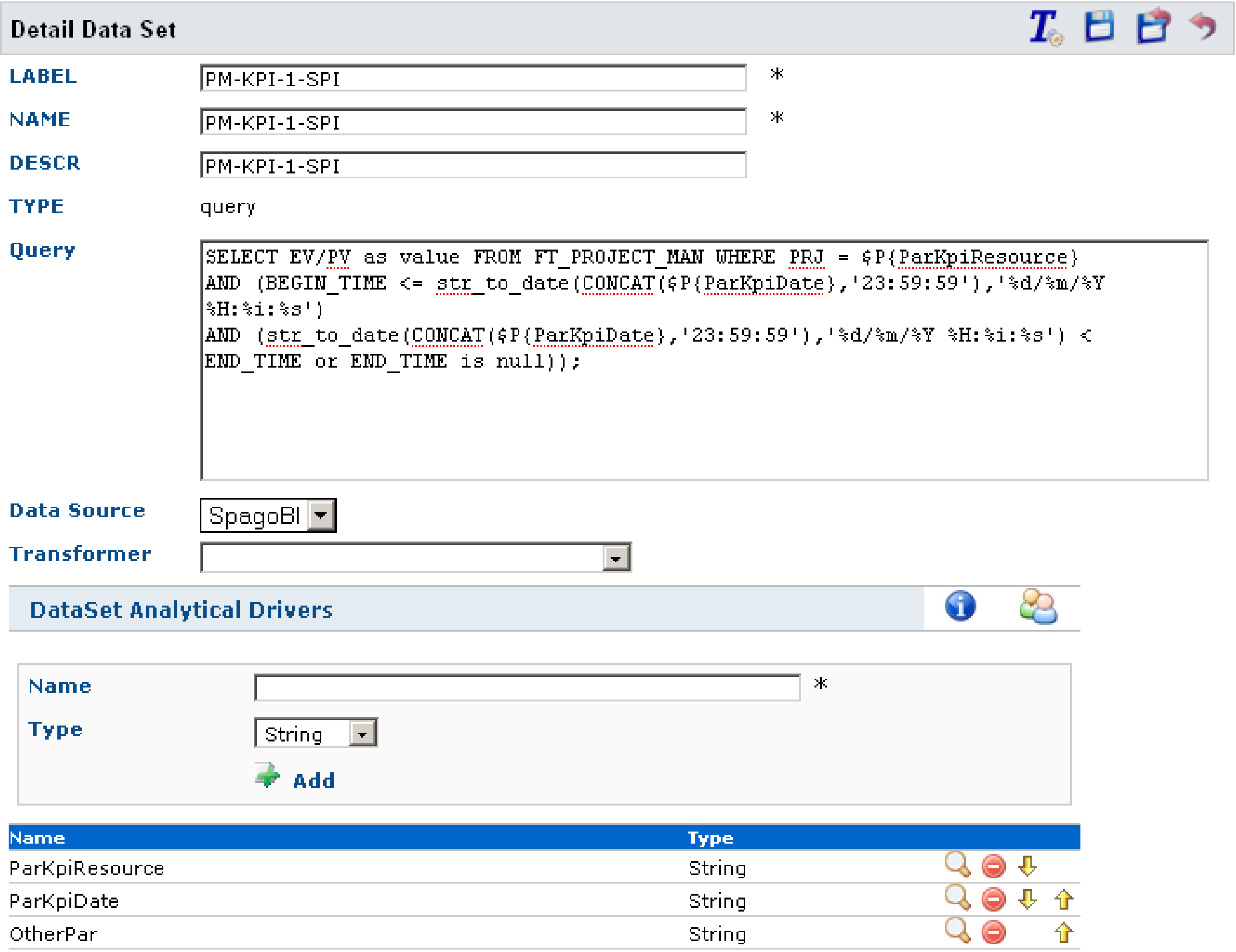
$P{ParKpiDate}	The reference date to use for the KPI value computation. The expected format is he one defined in the *spagobi.xml* configuration file in the tag *DATE-FORMAT-SERVER* ex: <DATE-FORMAT-SERVER format="dd/MM/yyyy"/>
$P{ParKpiResource}	The resource name (from the *SBI_RESOURCES* tables) that is used for the KPI value computation.

These parameters don't need to be added in the Kpi Document. If not added they will get the default values, explained below. If you want to add other parameters that are not the standard ones you will have to add them mandatorily in the Kpi Document or your kpi value will not be calculated. An example of a custom parameter is in the next screenshot:

$P{OtherPar}	Define the OtherPar in the Dataset Analytical Drivers as shown and use it inside the Data Set computation rule: for the query: $P{OtherPar}.

The next screenshot shows a query type Dataset.

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



LABEL	PM-KPI-1-SPI *
NAME	PM-KPI-1-SPI *
DESCR	PM-KPI-1-SPI
TYPE	query

Query

```
SELECT EV/PV as value FROM FT_PROJECT_MAN WHERE PRJ = $P{ParKpiResource}
AND (BEGIN_TIME <= str_to_date(CONCAT($P{ParKpiDate},'23:59:59'),'%d/%m/%Y
%H:%i:%s')
AND (str_to_date(CONCAT($P{ParKpiDate},'23:59:59'),'%d/%m/%Y %H:%i:%s') <
END_TIME or END_TIME is null));
```

Data Source: SpagoBI

Transformer:

Name	Type
ParKpiResource	String
ParKpiDate	String
OtherPar	String

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

8 How to create a KPI Document

In the following section it will be explained how to create a Kpi Document. First you need to define the parameters and LOVs (paragraphs 8.1 and 8.2) secondly the document template and after you have to configure the new KPI document on the platform.

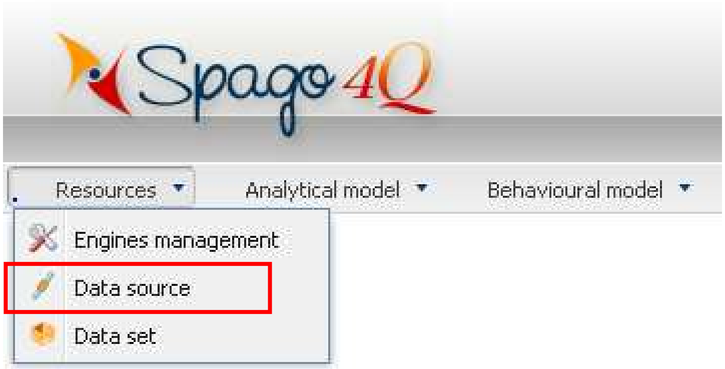
8.1 SPAGOBI PARAMETERS AND LOVS

In the following paragraphs it will first be explained how to define a new Data Source, a new List Of Values and a new Parameter. Afterwards you will be guided through the definition of the LOVs and Parameters that you could need for your Kpi Document. This is just an overview, for all other information just take a look at the SpagoBI Quick Start documentation. Go to www.spagobi.org download section and you will find all about LOVs, Parameters and Data Sources.

8.1.1 DATA SOURCE DEFINITION

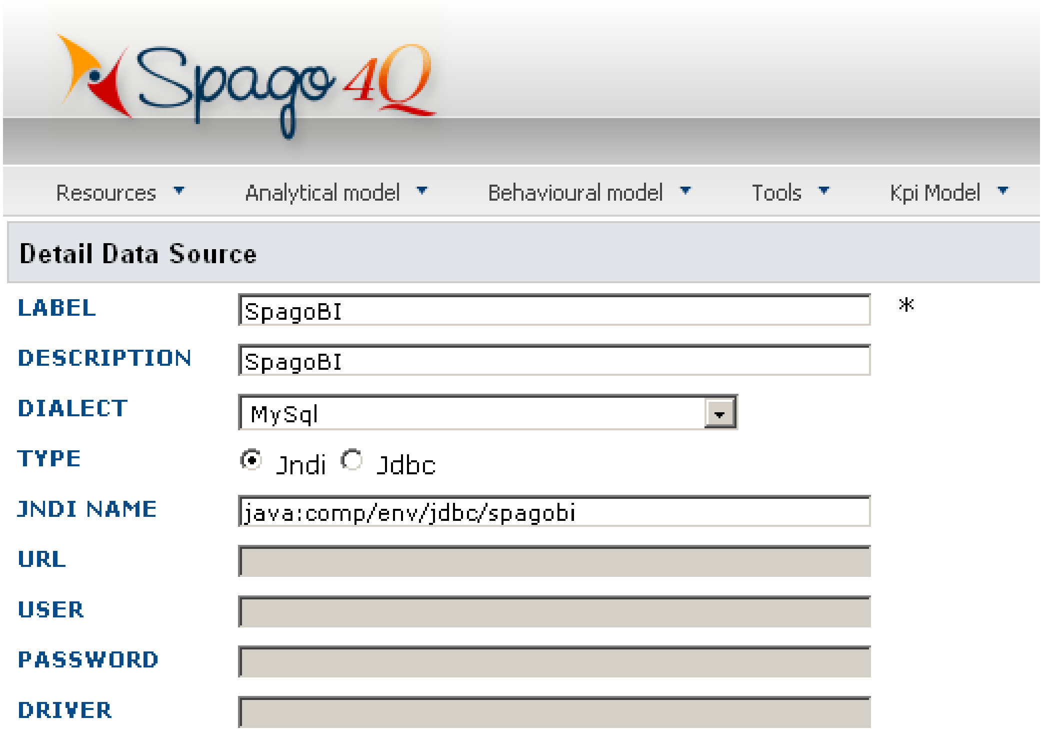
In order to create a Kpi document you will first need to create a Data Source that will permit you to interrogate the database on which you want to calculate your KPIs.

Click on the *Data Source* menu item and you will enter into the list of existing Data Sources.



In order to create a new Data Source click on the "Add" button on the top right corner of the page and you will enter in a form like the following one:

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Detail Data Source

LABEL: SpagoBI *
DESCRIPTION: SpagoBI
DIALECT: MySql
TYPE: ● Jndi ○ Jdbc
JNDI NAME: java:comp/env/jdbc/spagobi
URL:
USER:
PASSWORD:
DRIVER:

Fill all the blank spaces with the credentials that you need to access your database. The Label and the Description are up to you. If you have already defined your Data Source in the file *server.xml* in *apache_home/conf* then you can use the type *Jndi*. Differently, use the type *Jdbc*.
The Data Source in the file *server.xml* should look like this:

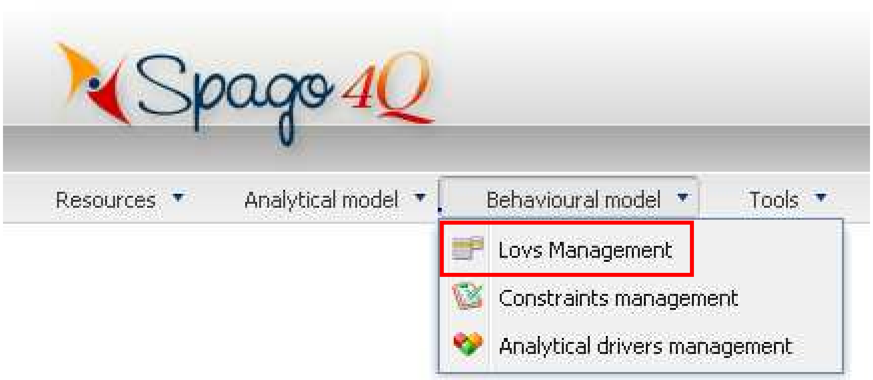
```
<Resource name="jdbc/spagobi" auth="Container"
        type="javax.sql.DataSource"
driverClassName="com.mysql.jdbc.Driver"
        url="jdbc:mysql://localhost/foodmart"
        username="root"        password="admin"        maxActive="20"
maxIdle="10"
        maxWait="-1"/>
```

It's possible to test the Data Source with the *Test* button on the top right corner of the page. If ok, a pop up window will come up saying "OK". When done, save the Data Source configuration with the save icon.

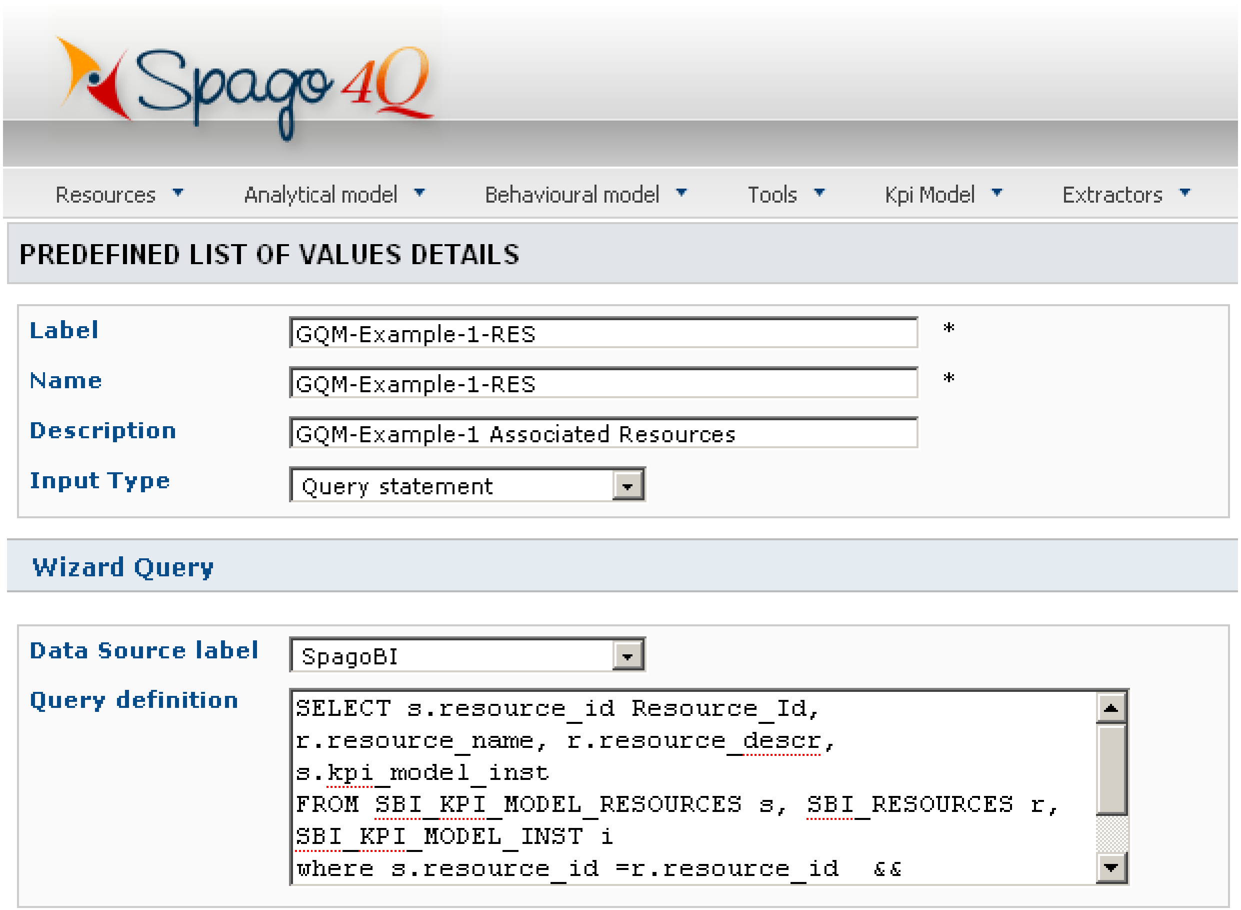
8.1.2 LOVs DEFINITION

Click on the *Lovs Management* menu item and you will enter into the list of existing List Of Values.

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



In order to create a new LOV click on the "Add" button on the top right corner of the page and you will enter in a form like the following one:



PREDEFINED LIST OF VALUES DETAILS

Label	GQM-Example-1-RES *
Name	GQM-Example-1-RES *
Description	GQM-Example-1 Associated Resources
Input Type	Query statement

Wizard Query

Data Source label SpagoBI

Query definition

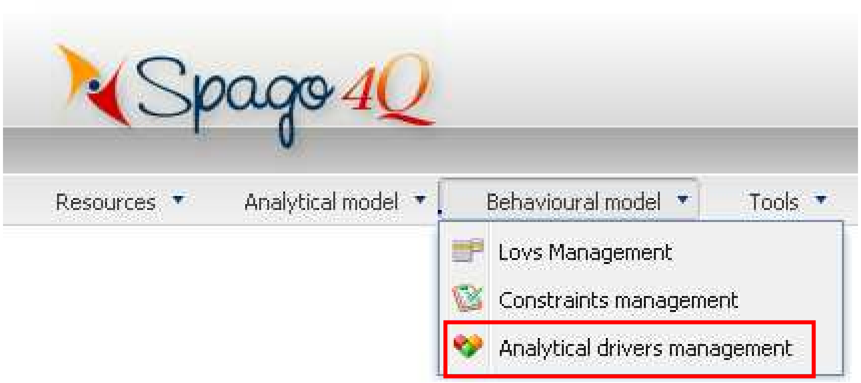
```
SELECT s.resource_id Resource_Id,
r.resource_name, r.resource_descr,
s.kpi_model_inst
FROM SBI_KPI_MODEL_RESOURCES s, SBI_RESOURCES r,
SBI_KPI_MODEL_INST i
where s.resource_id =r.resource_id  &&
```

Choose the type of LOV you want, for example a *Query*, write the query in the *Query Definition Box*. Test it and save it.

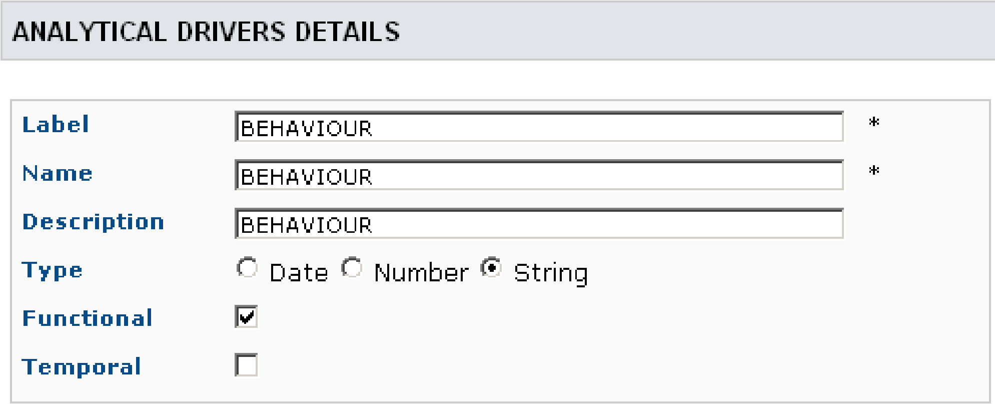
8.1.3 PARAMETERS DEFINITION

Click on the *Analytical drivers management* menu item and you will enter into the list of existing Parameters.

creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



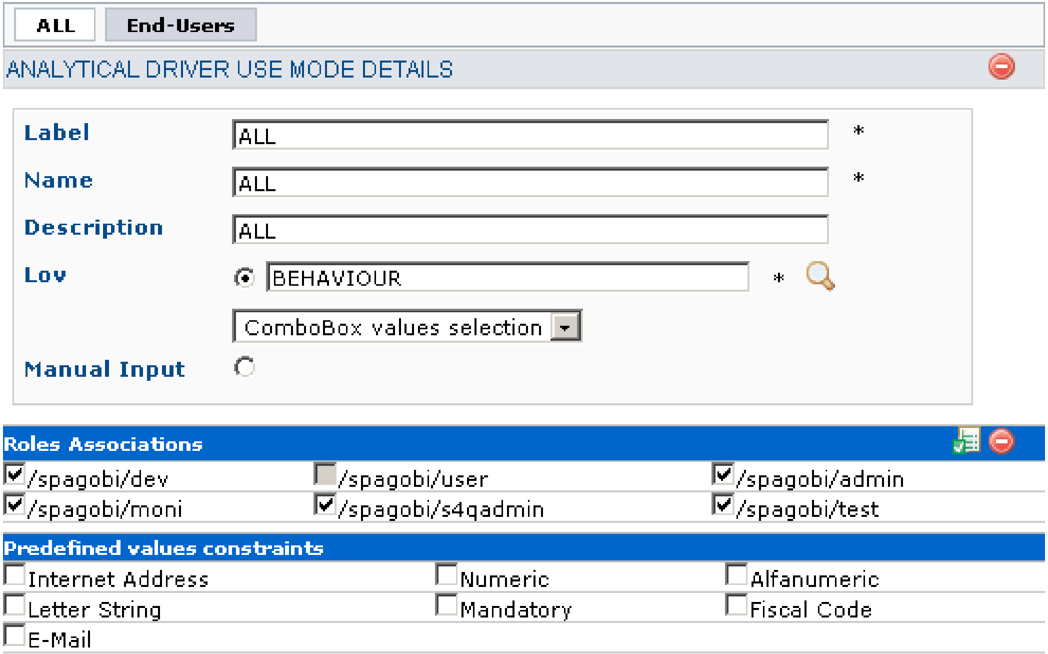
In order to create a new *Analytical Driver* click on the “Add” button on the top right corner of the page and you will enter in a form like the following one:



ANALYTICAL DRIVERS DETAILS

Label BEHAVIOUR *
Name BEHAVIOUR *
Description BEHAVIOUR
Type ○ Date ○ Number ● String
Functional ☑
Temporal ☐

Enter the required information and click on the “Save” button. A new form like the following one will appear:



ANALYTICAL DRIVER USE MODE DETAILS

Label ALL *
Name ALL *
Description ALL
Lov ● BEHAVIOUR *
ComboBox values selection
Manual Input ○

Roles Associations

☑/spagobi/dev ☐/spagobi/user ☑/spagobi/admin
☑/spagobi/moni ☑/spagobi/s4qadmin ☑/spagobi/test

Predefined values constraints

☐Internet Address ☐Numeric ☐Alfanumeric
☐Letter String ☐Mandatory ☐Fiscal Code
☐E-Mail

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

Fill all the blank spaces and click on the magnifying glass to select the LOV you want to use. A pop up window with the list of existing LOVs will appear. Select the one you want. When finished click on the "Save" button.

8.2 LOVS AND PARAMETERS FOR KPI DOCUMENTS

In the following section it will be explained the meaning and how to create the standard LOVS and Parameters for a KPI Document.

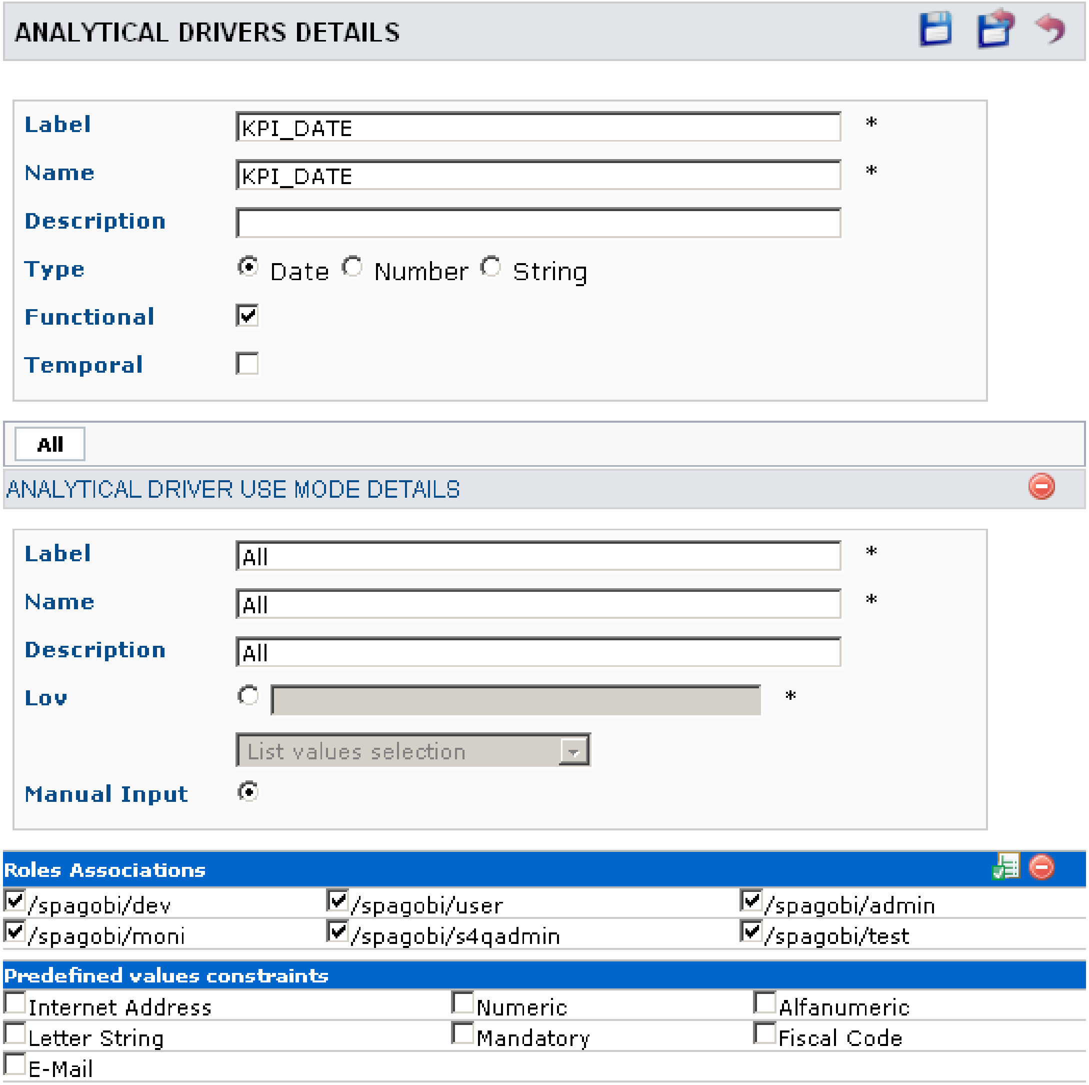
8.2.1 "DATE" PARAMETER

This parameter allow the end user to select a Date at the moment he executes the KPI document. By selecting a date different from current date, the KPIs values will be presented at the moment of the Date selected.

DEFAULT: If this parameter is not present in the KPI document execution form, or it is present but the user decides to leave it blank, the **current date** and **time** will be taken.

To create the parameter you have to click on the menu item "Analytical drivers management" and on the "Add" button on the top right corner of the page. Once you are on an empty detail, fill the form with the data shown in the next screenshot and save:

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



ANALYTICAL DRIVERS DETAILS

Label	KPI_DATE *
Name	KPI_DATE *
Description	
Type	● Date ○ Number ○ String
Functional	☑
Temporal	☐

All

ANALYTICAL DRIVER USE MODE DETAILS

Label	All *
Name	All *
Description	All
Lov	○ *
	List values selection
Manual Input	●

Roles Associations

☑ /spagobi/dev	☑ /spagobi/user	☑ /spagobi/admin
☑ /spagobi/moni	☑ /spagobi/s4qadmin	☑ /spagobi/test

Predefined values constraints

☐ Internet Address	☐ Numeric	☐ Alfanumeric
☐ Letter String	☐ Mandatory	☐ Fiscal Code
☐ E-Mail		

8.2.2 “BEHAVIOUR” PARAMETER

This parameter allow the end user to select the *Behaviour* type, for the computation rules, at the moment he executes the KPI document. The behaviour define when the computation engine has to update the KPI values. When a KPI value is computed it will be stored in the database with two dates: a *begin date* and an *end date* that defines the range of validity of the KPI value itself. This range is obtained from the periodicity attribute of the KPI instance, if defined, otherwise an *infinite* end time will be used. (the convention for infinite is December 31th, 9999).

The possible *Behaviour* values are the following ones:

- **Display**: The Kpi Engine will not calculate new KPI values but it will only show the ones already available in the database. Accordingly with the date parameter required, the engine will show the more recent kpi value (kpi begin date <= req. date).
- **Default**: For each KPI associated to the model instance, the KPI Engine, will check if at the requested date it’s possible to find an already calculated value (kpi begin

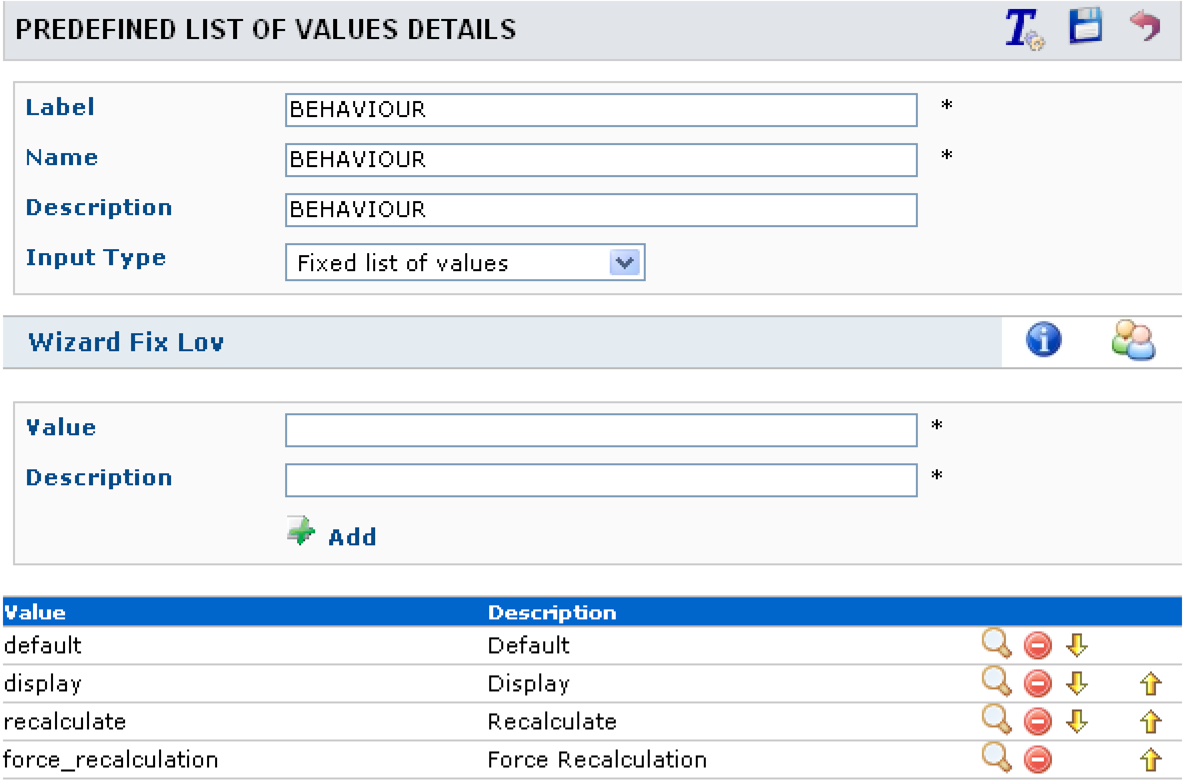
creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

date <= req. date < kpi end date). If no valid value exists, a new value will be calculated at the requested date. If it find more than one value it will act like in the Display behaviour and it will show the more recent one from the requested date (kpi begin date <= req. date).

- **Recalculate**: The behaviour is the same as the *Default* one, except that all the KPIs that don't have a defined *periodicity* attribute will be recalculated anyway even if a valid value already exist.
- **Force Calculation**: All KPIs are recalculated at the requested date even if valid values already exist.

DEFAULT: If this parameter is not present in the KPI document execution form or it is present but the user decides to leave it blank, the **Default** behaviour will be applied.

Before to proceed with the parameter definition the LOV has to be created. First, click on the menu item "LOVs management" and you will enter into the list of existing LOVs, and than click on the "Add" button on the top right corner of the page to enter the detail form. Fill all the blanks with the data shown in the figure and save:



PREDEFINED LIST OF VALUES DETAILS

Label	BEHAVIOUR *
Name	BEHAVIOUR *
Description	BEHAVIOUR
Input Type	Fixed list of values

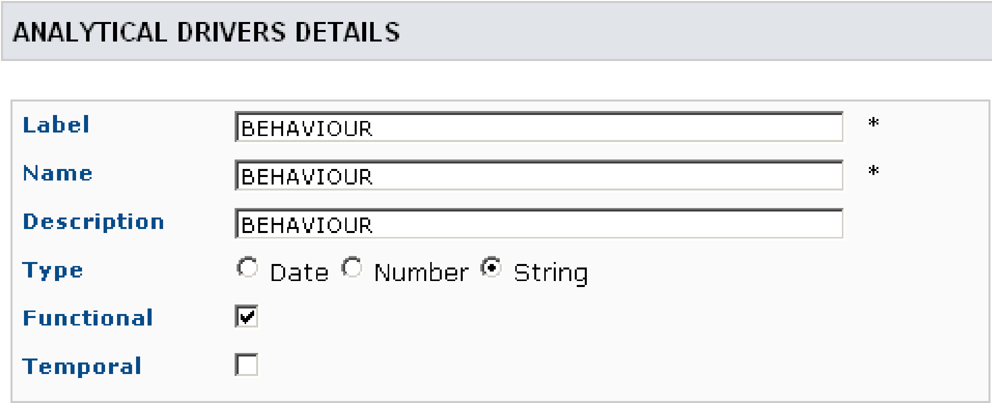
Wizard Fix Lov

Value	*
Description	*

Value	Description
default	Default
display	Display
recalculate	Recalculate
force_recalculation	Force Recalculation

If you look at the Sample configuration you can find a LOV called BEHAVIOUR-DISPLAY used for the end user in order to avoid them to execute KPI calculation but allow them only to see pre-calculated values.
The Behaviour parameter configuration is:

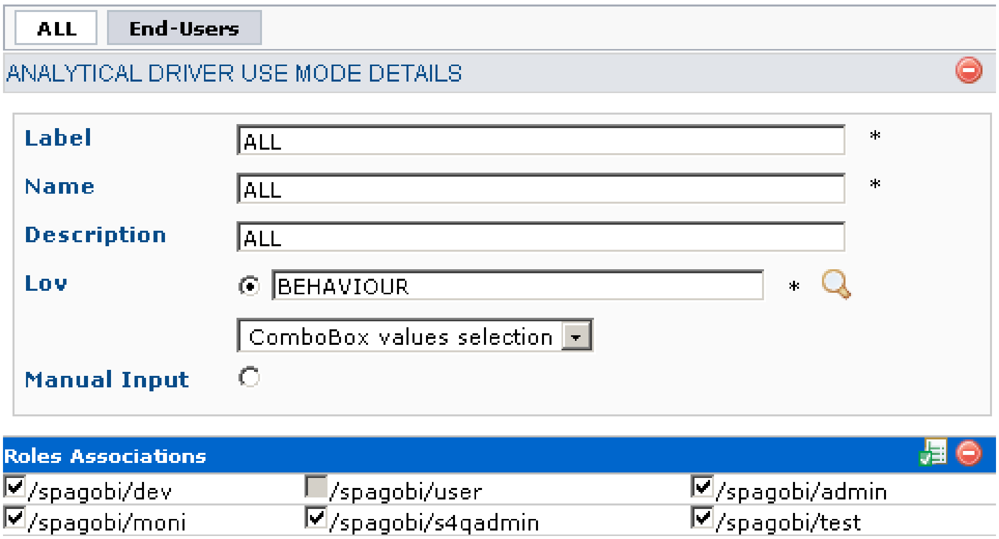
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



ANALYTICAL DRIVERS DETAILS

Label	BEHAVIOUR *
Name	BEHAVIOUR *
Description	BEHAVIOUR
Type	○ Date ○ Number ● String
Functional	☑
Temporal	☐

And the two usage modes are:

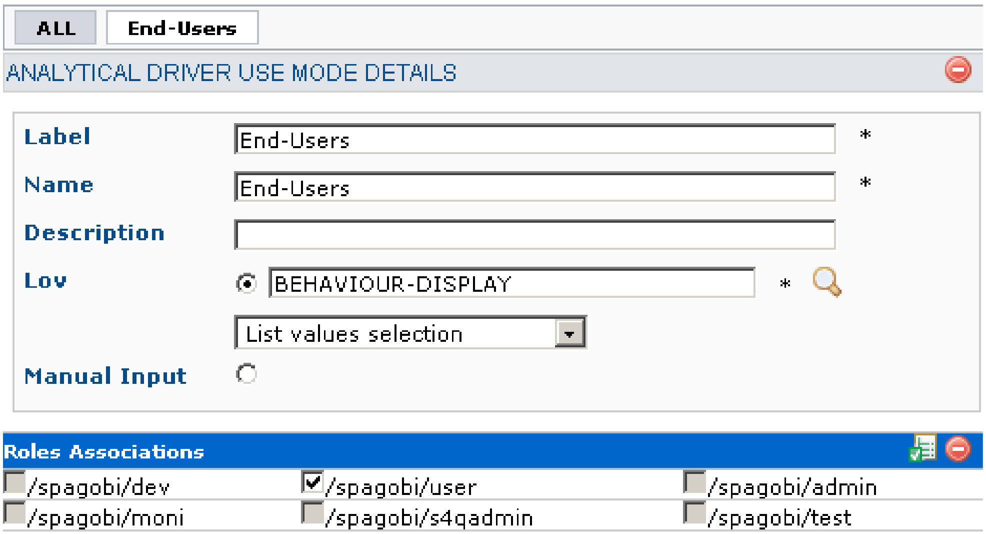


ANALYTICAL DRIVER USE MODE DETAILS

Label	ALL *
Name	ALL *
Description	ALL
Lov	● BEHAVIOUR * ComboBox values selection
Manual Input	○

Roles Associations

☑ /spagobi/dev	☐ /spagobi/user	☑ /spagobi/admin
☑ /spagobi/moni	☑ /spagobi/s4qadmin	☑ /spagobi/test



ANALYTICAL DRIVER USE MODE DETAILS

Label	End-Users *
Name	End-Users *
Description	
Lov	● BEHAVIOUR-DISPLAY * List values selection
Manual Input	○

Roles Associations

☐ /spagobi/dev	☑ /spagobi/user	☐ /spagobi/admin
☐ /spagobi/moni	☐ /spagobi/s4qadmin	☐ /spagobi/test

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

8.2.3 "REGISTER VALUES" PARAMETER

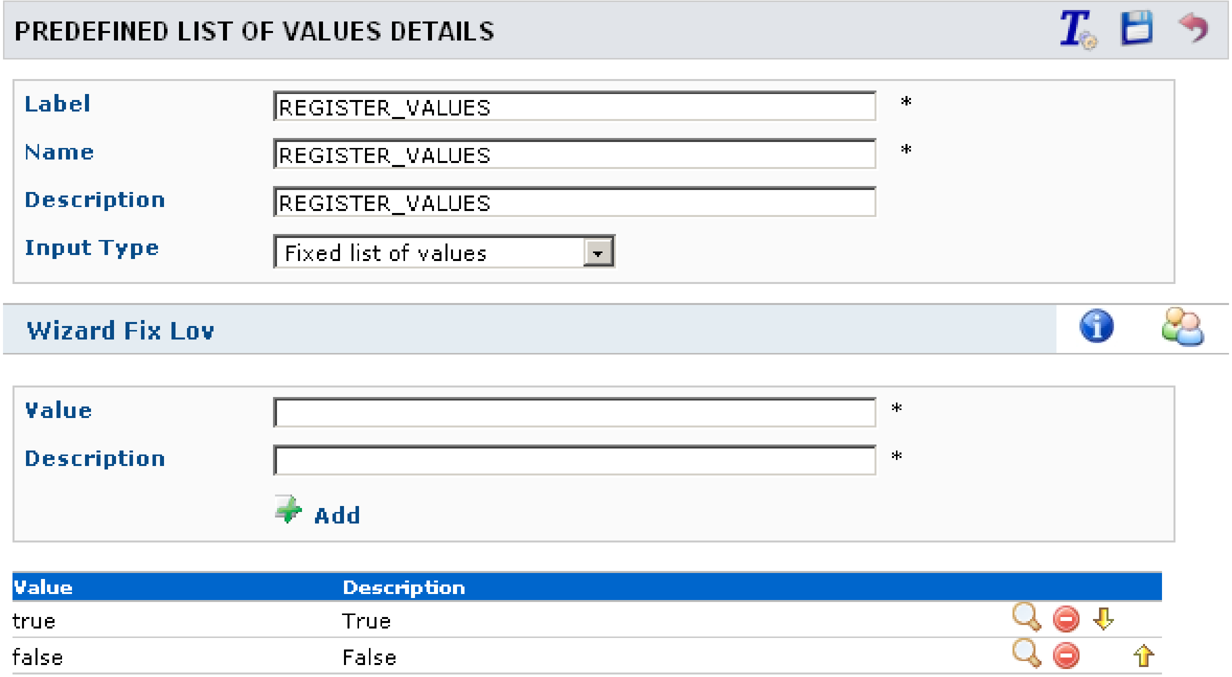
This parameter allow the user to select whether or not to store the calculated KPI values into the database.
The possible values are:

- **True**: All the new calculated values will be registered into the database.
- **False**: All the new calculated values will only be shown to the user, but will not be registered into the database.

DEFAULT: If this parameter is not present in the KPI document execution form or it is present but the user decides to leave it blank, the parameter will use the **True** value.

As for the previous parameter is has been decided to setup two LOVs, one with both *True* and *False* values and another with the *False* value only (for end users).

The First LOV is on the next screenshot:

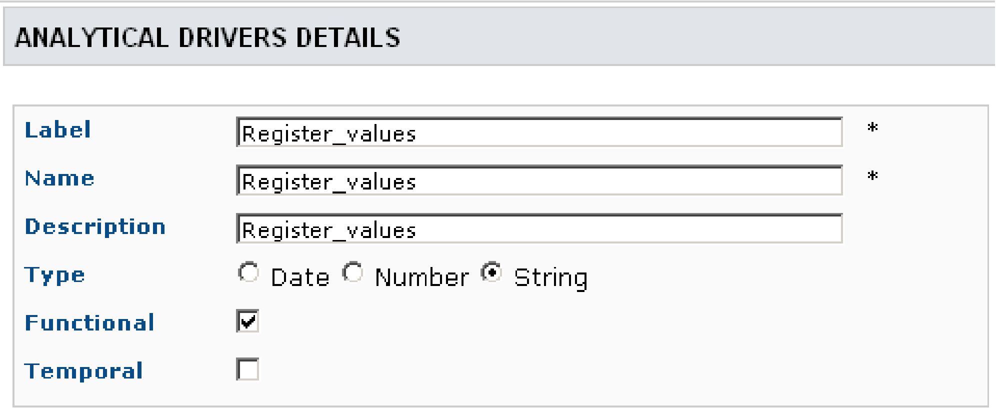


Label	REGISTER_VALUES *
Name	REGISTER_VALUES *
Description	REGISTER_VALUES
Input Type	Fixed list of values
Value	*
Description	*

Value	Description
true	True
false	False

The parameter configuration is like the one of the *Behaviour* parameter, with the two usage modes:

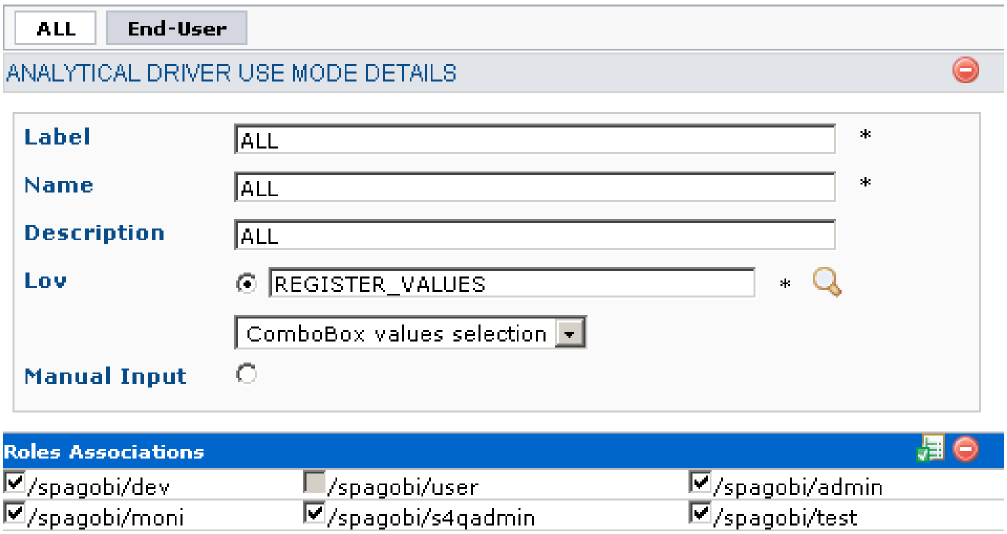
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



ANALYTICAL DRIVERS DETAILS

Label	Register_values	*
Name	Register_values	*
Description	Register_values	
Type	○ Date ○ Number ● String	
Functional	☑	
Temporal	☐	

Usage Modes:

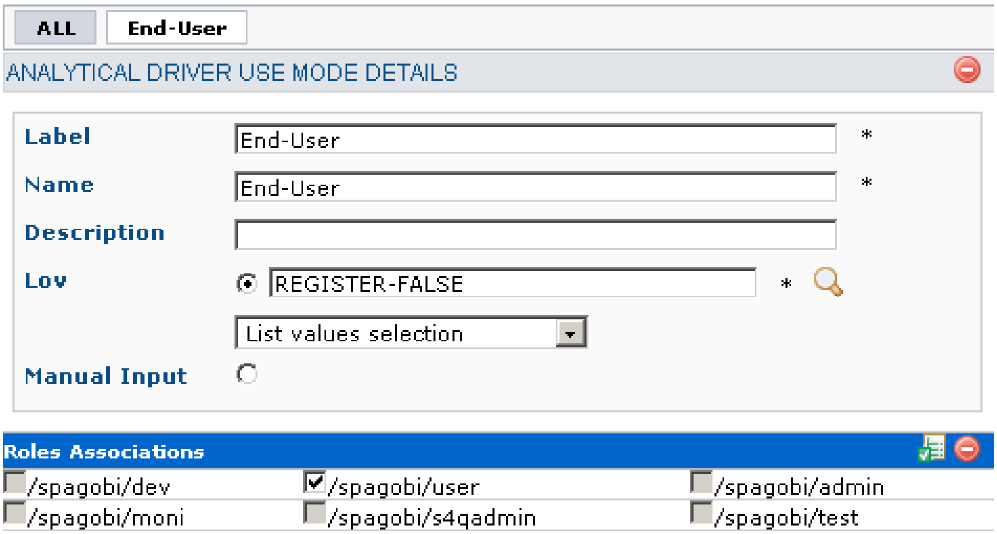


ANALYTICAL DRIVER USE MODE DETAILS

Label	ALL	*
Name	ALL	*
Description	ALL	
Lov	● REGISTER_VALUES	*
	ComboBox values selection	
Manual Input	○	

Roles Associations

☑ /spagobi/dev	☐ /spagobi/user	☑ /spagobi/admin
☑ /spagobi/moni	☑ /spagobi/s4qadmin	☑ /spagobi/test



ANALYTICAL DRIVER USE MODE DETAILS

Label	End-User	*
Name	End-User	*
Description		
Lov	● REGISTER-FALSE	*
	List values selection	
Manual Input	○	

Roles Associations

☐ /spagobi/dev	☑ /spagobi/user	☐ /spagobi/admin
☐ /spagobi/moni	☐ /spagobi/s4qadmin	☐ /spagobi/test

creative commons COMMONS DEED

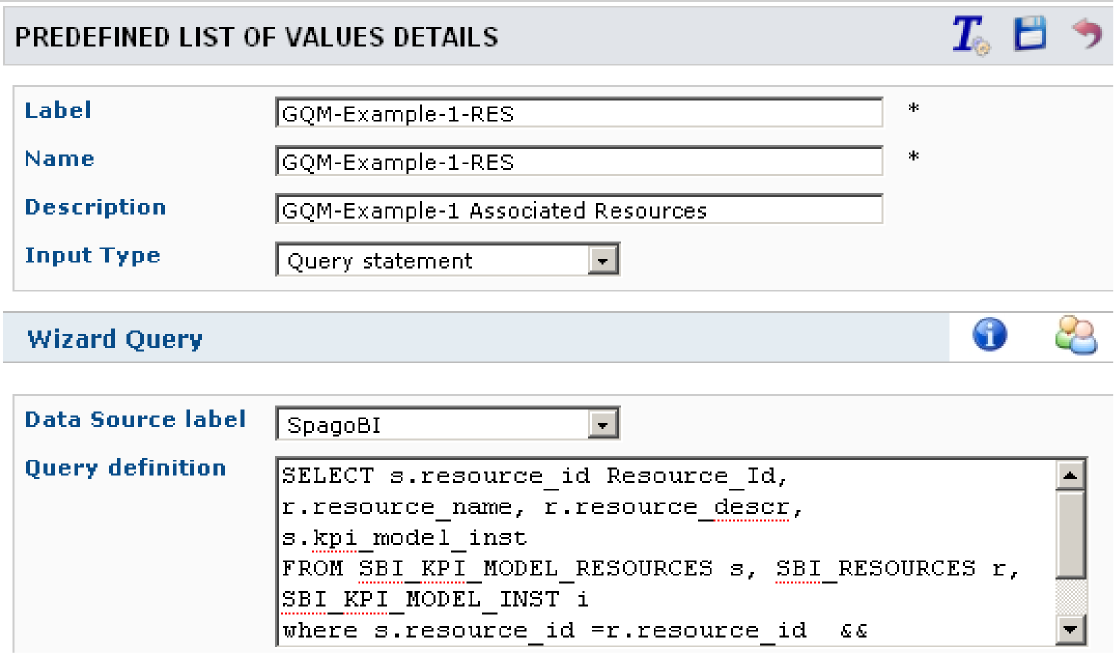
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

8.2.4 “RESOURCES” PARAMETER

This parameter allow the user to select for which resources he wants to see the analysis. One or more resources can be selected at the same time.

DEFAULT: If this parameter is not present in the KPI document execution form or it is present but the user decides to leave it blank, the document will calculate and show the KPI Values for all the resources associated to the Model Instance represented in the requested KPI document.

The LOV used for the parameter is show in the next screenshot:



Label GQM-Example-1-RES *
Name GQM-Example-1-RES *
Description GQM-Example-1 Associated Resources
Input Type Query statement

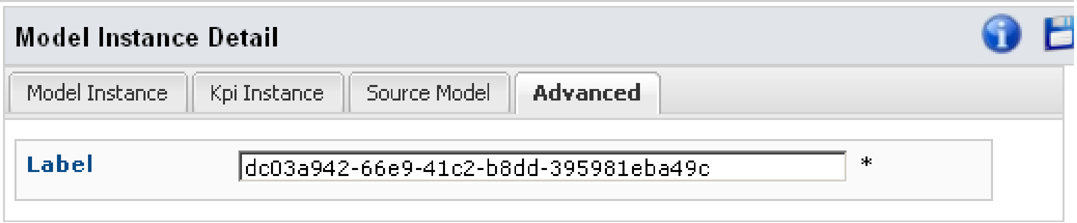
Data Source label SpagoBI
Query definition

```
SELECT s.resource_id Resource_Id,
r.resource_name, r.resource_descr,
s.kpi_model_inst
FROM SBI_KPI_MODEL_RESOURCES s, SBI_RESOURCES r,
SBI_KPI_MODEL_INST i
where s.resource_id =r.resource_id  &&
```

The query will retrieve all the resources associated to the defined Model Instance (here the *Model Instance Label* is used). The query is:

```
SELECT s.resource_id Resource_Id, r.resource_name, r.resource_descr,
s.kpi_model_inst
FROM SBI_KPI_MODEL_RESOURCES s, SBI_RESOURCES r,
SBI_KPI_MODEL_INST i
where s.resource_id =r.resource_id && s.kpi_model_inst =
i.kpi_model_inst && i.label = 'dc03a942-66e9-41c2-b8dd-
395981eba49c';
```

The *Label* is taken from the Model Instance detail, *Advanced* tab:



Label dc03a942-66e9-41c2-b8dd-395981eba49c *

The parameter definition is the same for all the roles (one usage mode).

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

8.3 TEMPLATE FILE

The Document template has to be manually prepared following the structure defined and described below.

```
<?xml version="1.0" encoding="windows-1250"?>
<KPI model_node_instance='[Model Instance Label]' name='Title of the document'>
  <STYLE_SUBTITLE font='Arial' size='10' color='#6699FF' name='Subtitle of the document – Date: $P{ParKpiDate}'/>
  <STYLE_TITLE font='Arial' size='12' color='#000000'/>
  <CONF>
    <PARAMETER name='display_semaphore' value='true'/>
    <PARAMETER name='display_bullet_chart' value='true'/>
    <PARAMETER name='display_weight' value='true'/>
    <PARAMETER name='display_alarm' value='true'/>
  </CONF>
</KPI>
```

You can make your own template by substituting 'Model Instance Label' with the Label of the root node of your model instance (the same used in LOV for the resources associated to a Model Instance), 'Title of the document' with the title you want to have for your document, 'Subtitle of the document' for the subtitle and of course you can change the style of title and subtitle. To print out the Date parameter value you can use the *$P{ParKpiDate}* placeholder.
The *Conf* part is used to define to show (true) or hide (false) some parts of the KPI Document report:

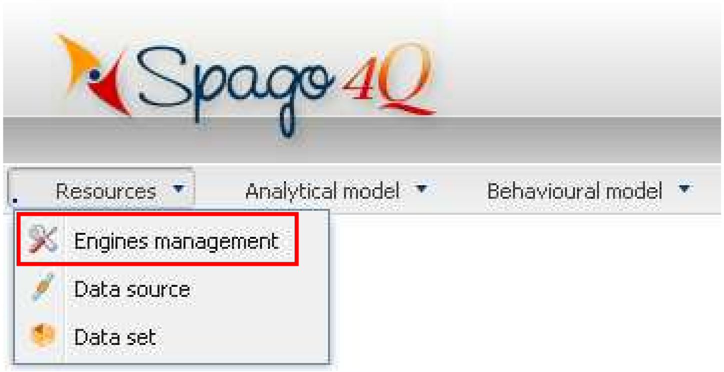
- **display_semaphore**: it's a small semaphore that show the color of the threshold range that the KPI value is in.
- **display_bullet_chart**: it's a chart representing the requested KPI value on it's whole threshold ranges.
- **display_weight**: the weight defined on the KPI instance.
- **display_alarm**: highlight if there is an alarm configured for the KPI.

Once the template is ready it's possible to proceed with the Document Configuration.

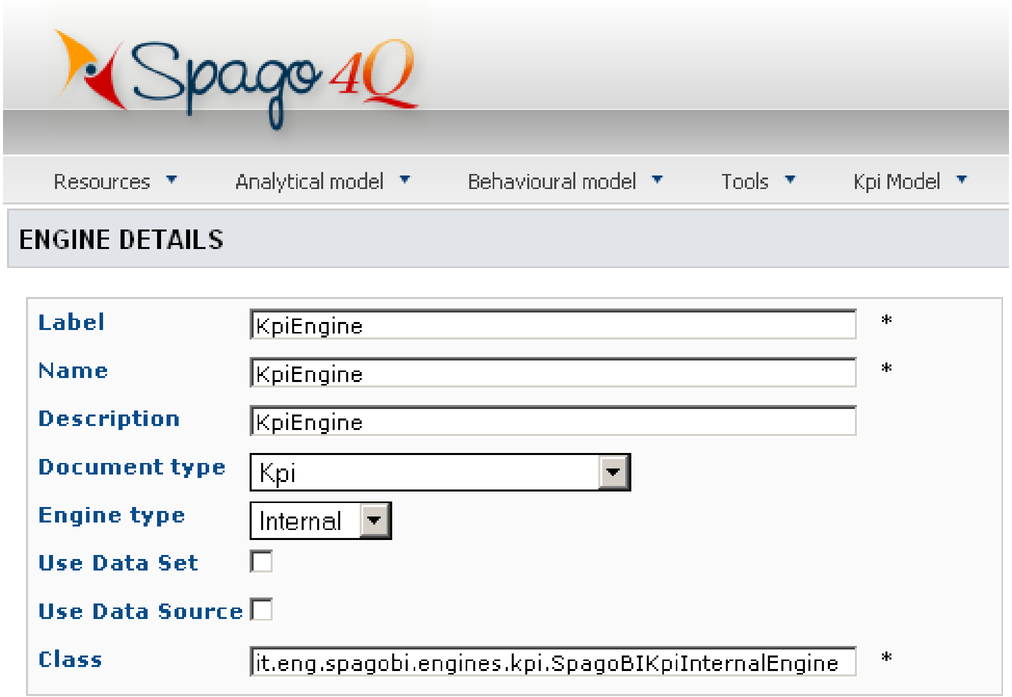
8.4 KPI ENGINE

First of all check the existence of the *KPI Engine*: Click on the menu item "Engine Management" and you will enter in the Engines list page. Check if you have an engine called "KpiEngine".

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



In case the Engine is not present, add it yourself by clicking on the "Add" button on the top right corner of the page and fill the blanks as in the following figure:



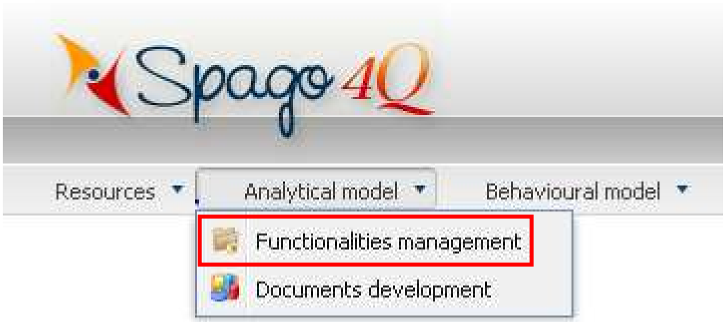
ENGINE DETAILS

Label	KpiEngine	*
Name	KpiEngine	*
Description	KpiEngine	
Document type	Kpi	
Engine type	Internal	
Use Data Set	☐	
Use Data Source	☐	
Class	it.eng.spagobi.engines.kpi.SpagoBIKpiInternalEngine	*

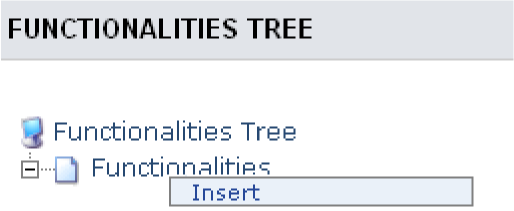
8.5 DOCUMENT FOLDER

Every document needs a folder under the *Functionalities Tree* to be stored in. Create a KPI folder child of the Functionalities one.

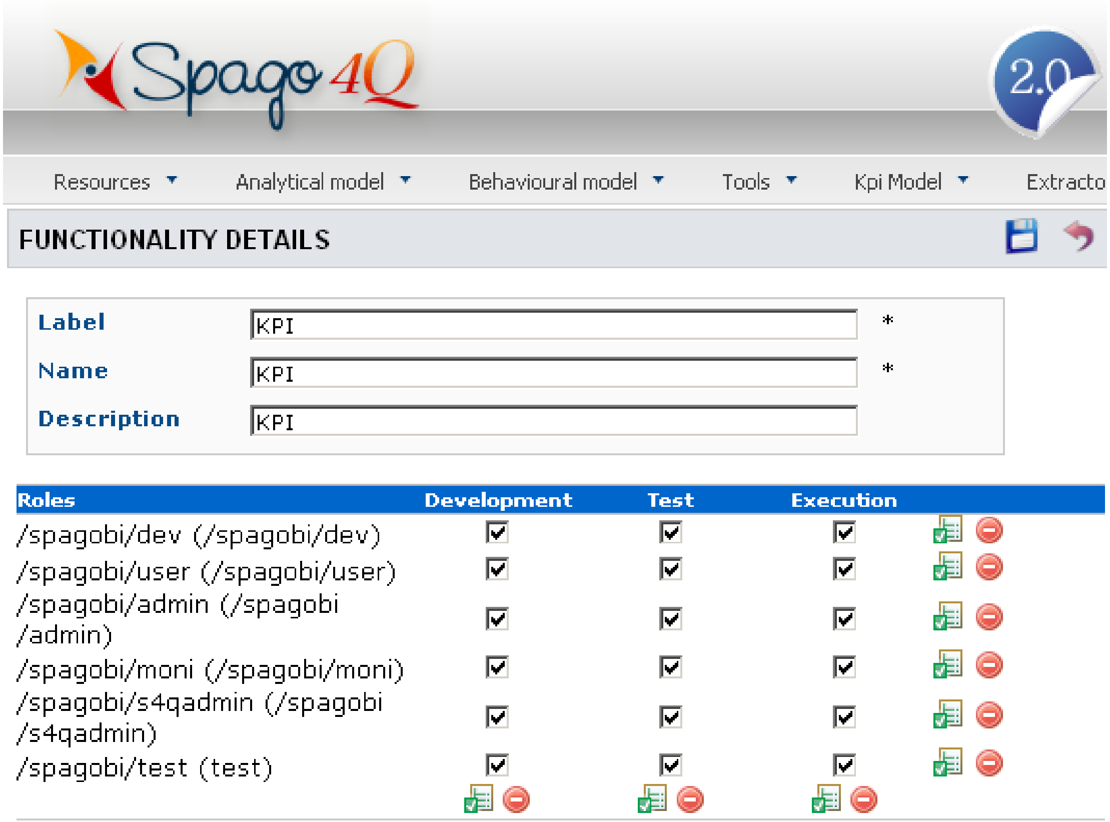
creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



Add a new folder by clicking on the "Insert" menu item that appears by clicking over the word "Functionalities"



Fill the form with the configuration as shown in the screenshot and set the actions for each role:



FUNCTIONALITY DETAILS

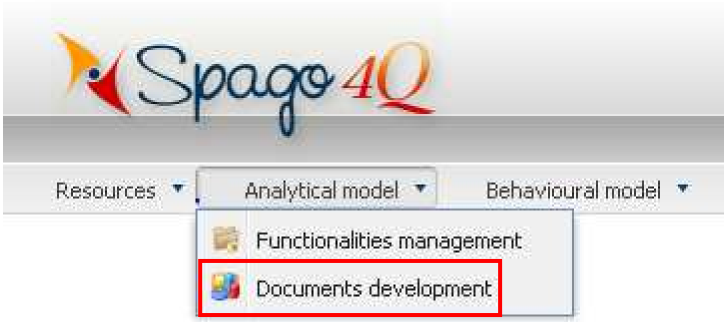
Label	KPI	*
Name	KPI	*
Description	KPI	

Roles	Development	Test	Execution
/spagobi/dev (/spagobi/dev)	☑	☑	☑
/spagobi/user (/spagobi/user)	☑	☑	☑
/spagobi/admin (/spagobi /admin)	☑	☑	☑
/spagobi/moni (/spagobi/moni)	☑	☑	☑
/spagobi/s4qadmin (/spagobi /s4qadmin)	☑	☑	☑
/spagobi/test (test)	☑	☑	☑

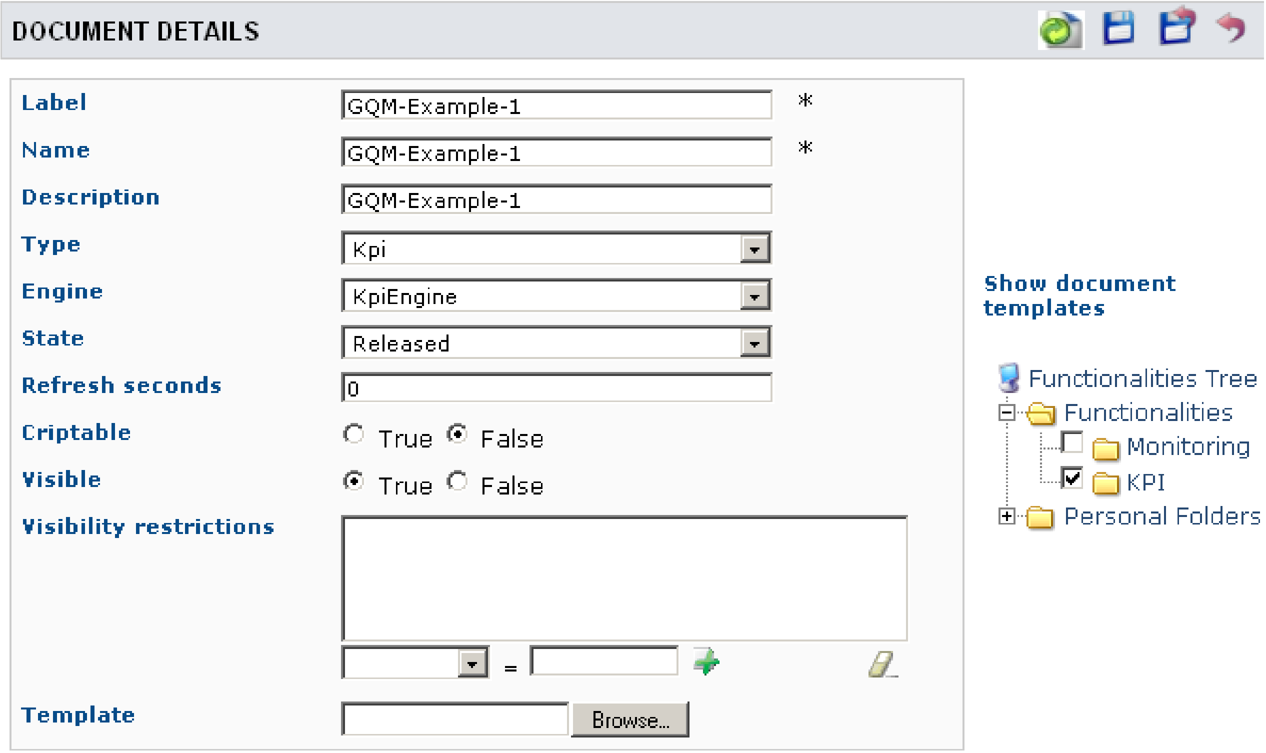
creative commons COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

8.6 DOCUMENT CREATION

In order to create a new Document click on the menu item “Documents development” and you will enter into the list of existing Documents.



Click on the “Add” button on the top right corner of the page and fill the first form information:



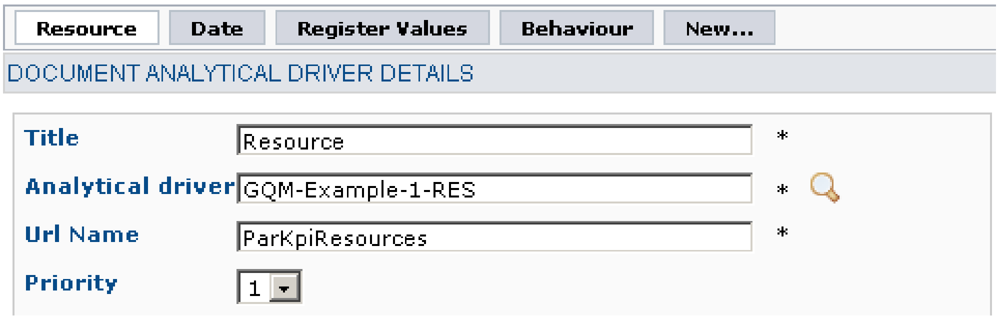
DOCUMENT DETAILS

Label	GQM-Example-1 *
Name	GQM-Example-1 *
Description	GQM-Example-1
Type	Kpi
Engine	KpiEngine
State	Released
Refresh seconds	0
Criptable	○ True ● False
Visible	● True ○ False
Visibility restrictions	
Template	

Here you have to insert the template created before, select in which folders of the *Functionalities Tree* you want to make the *KPI Document* available and click on the 'Save' Button.

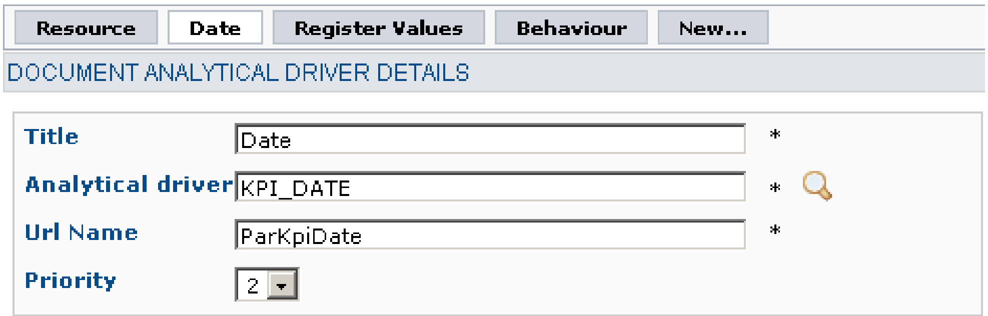
Here it is shown how to insert all the parameters already defined. You can insert them in the order you prefer. For each parameter you will have to click on the Tab “New...” and before going on with another one you have to click on the 'Save' button.

creative commons
COMMONS DEED
Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/



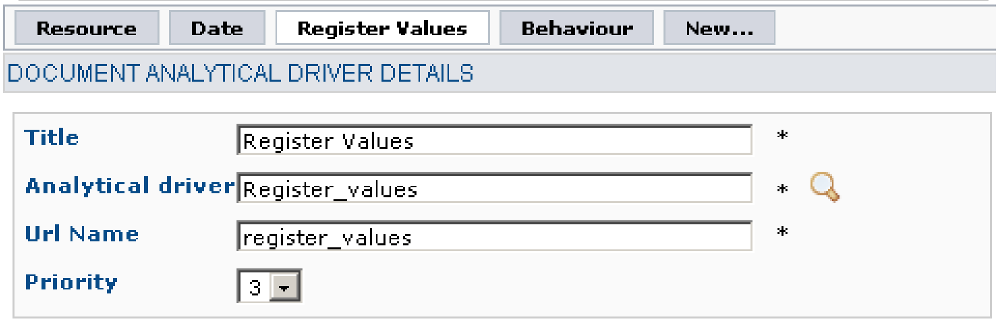
DOCUMENT ANALYTICAL DRIVER DETAILS

Title Resource *
Analytical driver GQM-Example-1-RES *
Url Name ParKpiResources *
Priority 1



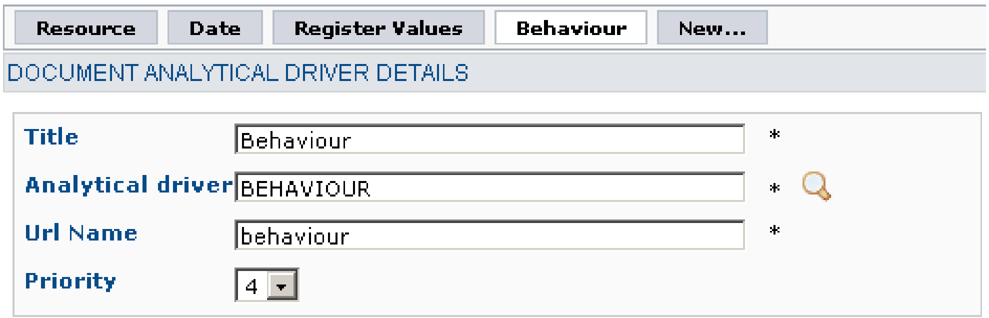
DOCUMENT ANALYTICAL DRIVER DETAILS

Title Date *
Analytical driver KPI_DATE *
Url Name ParKpiDate *
Priority 2



DOCUMENT ANALYTICAL DRIVER DETAILS

Title Register Values *
Analytical driver Register_values *
Url Name register_values *
Priority 3



DOCUMENT ANALYTICAL DRIVER DETAILS

Title Behaviour *
Analytical driver BEHAVIOUR *
Url Name behaviour *
Priority 4

Your KPI Document is done.

creative commons COMMONS DEED

Engineering Ingegneria Informatica S.p.A., 2009. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/

BY: