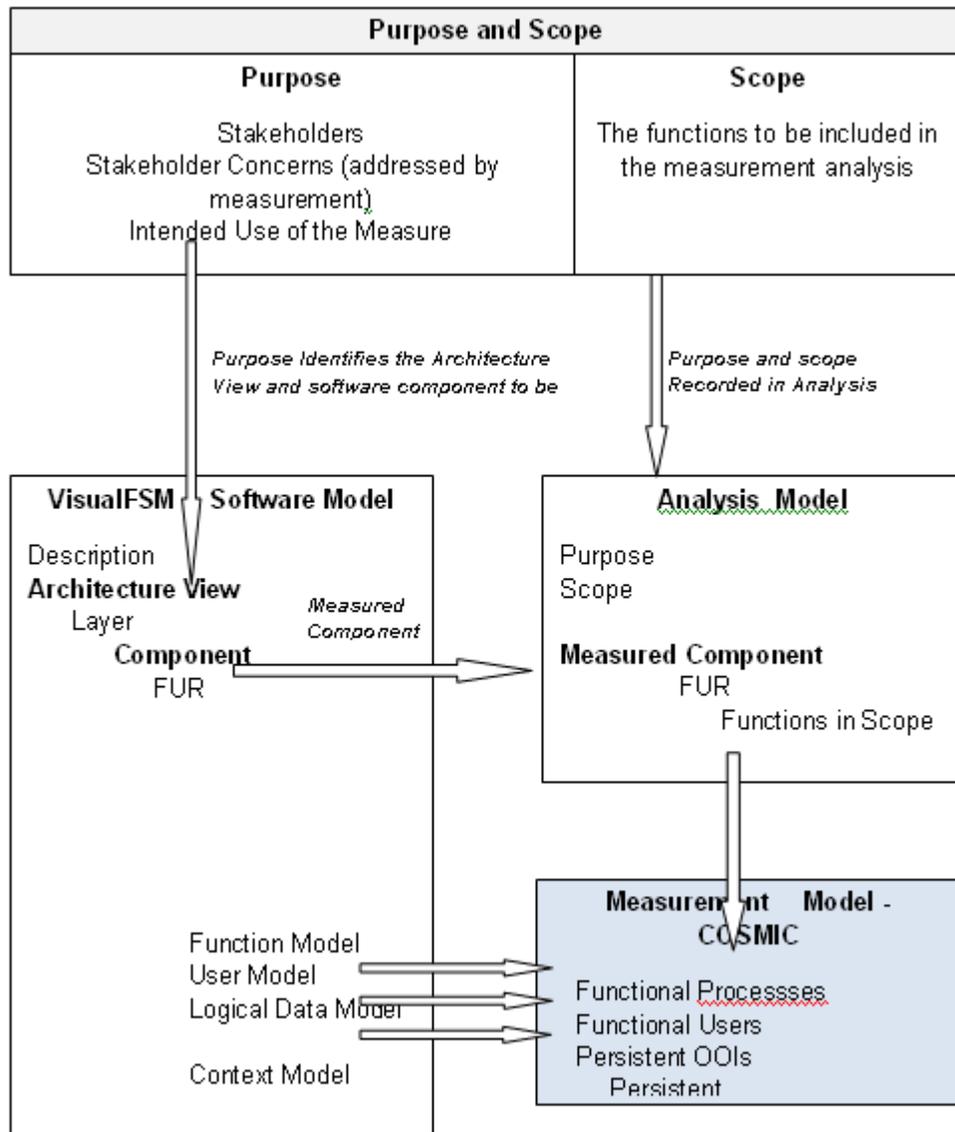**Visual fsm**

# COSMIC

# Quickstart Guide

# 1    Introduction

## The VisualFSM Measurement Process

The VisualFSM measurement process and the role of this Guide within it is shown in the diagram below.



## Process Summary

### Preconditions

1. The Purpose and Scope have been defined

2. The Software Model is complete.

3. The Analysis Model is complete

## Select the Measurement Method

At this stage it is assumed that the COSMIC method has been chosen. VisualFSM will automatically create the COSMIC Generic Software Model, and populate it with the information it already holds

1. From the Function Model it will create a Functional Process for each Function

2. From the Logical Data Model it will create the candidate OOI model by

creating a persistent OOI  for each logical Datafile,

for each OOI it will create a persistent Datagroup

for each OOI-Datagroup it will create an Attribute for each Datafield in he associated Datafile

3. From the User Model it create the Functional User Model creating a Functional User for each User

# Scope of This Guide

The COSMIC Generic Software Model is represented as 2 separate Models to aid the analysis process.

The **Software Model** which contains the

- Functional Users

- Triggering Events

- Object of Interest/Datagroups/Attributes

The **Function Model** which contains the Data Movements for each Functional Process,

The Entry and Exit Data Movements showing the Functional User, OOI and  Data Group

The Read and Write Data Movements showing the Functional OOI and  Data Group retrieved and persisted

# Software Model.

VisualFSM will automatically convert those items in the Generic models tp the equivalent items in the Cosmoc models, however COSMIC has some specific requiremenmt in order to complete the Software model. The first step is to complete it.

## Process

By reference to the FUR for each Functional Process

1. Identify and add any additional Functional Users

2. Identify and add the Triggering Events

3. .Identify and add any additional Persistent Objects of Interest together with the attributes

4. Identify and add the Transient Objects of Interest with the attributes

## Outcome

The outcome is a completed Software Model with every element traceable to the FUR.

# Function Model

The Function Model is construct from the the elements in the Software model

## Process

For each Functional process, by reference to its FUR, create the Function Model by selecting and assigning the appropriate elements from the Software Model

1.  Identify and add the Triggering Entry

2.  Identify and add the other entries

3.  Identify and add the Reads

4.  Identify and add the Writes

5.  Identify and add the Exits

## Outcome

The Outcome is a completed Function Model.

VisualFSM will automatically calculate and document the size id each Functional process. The measurement is fully traceable to the FUR.

The chapters that follow describe how each of the above tasks is performed

# 2     Preparation

## Introduction

VisualFSM is semi-automated in that it creates the first-cut Software Model and the Function Model automatically , based on the information in the User Data and Function models previously created. These models equate broadly to the Generic Software Model and the Measurement Models described in the COSMIC Measurement Manual
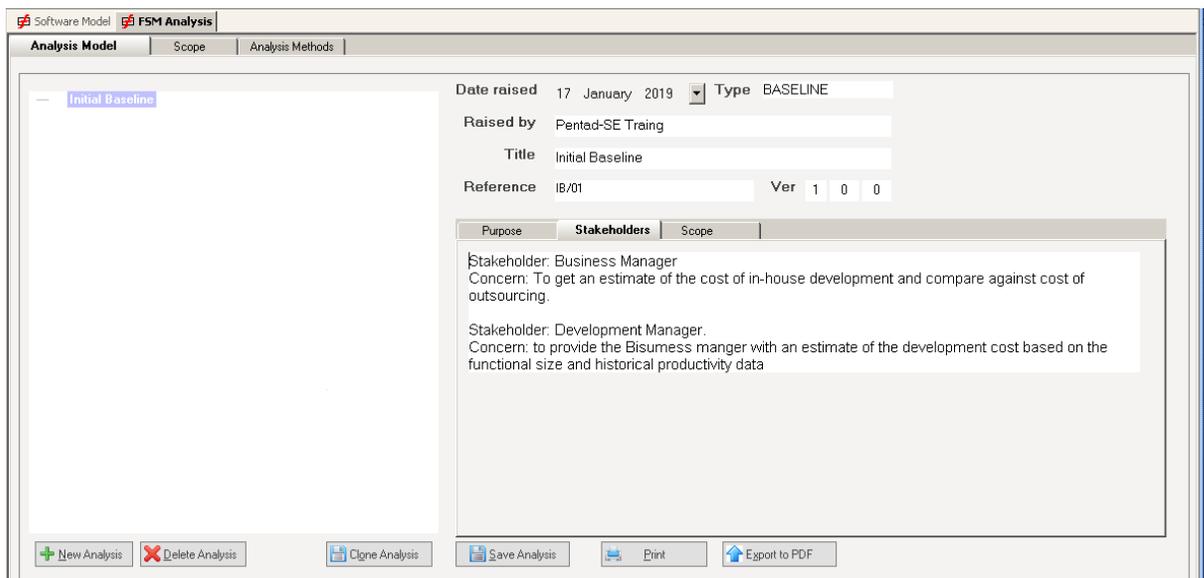
It is important to note that the artefacts resulting are Candidates at this stage. They become "real" when they are referenced by a Functional Process

## Create the COSMIC Analysis

1. Select the *FSM Analysis* tab
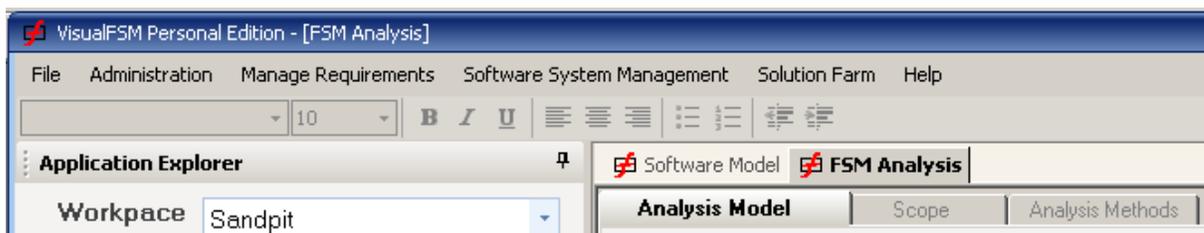


2. From the Analysis Model tree select the Initial Baseline Analysis



4. select the *Analysis Methods Tab*

5. click "**New Measurement**" button
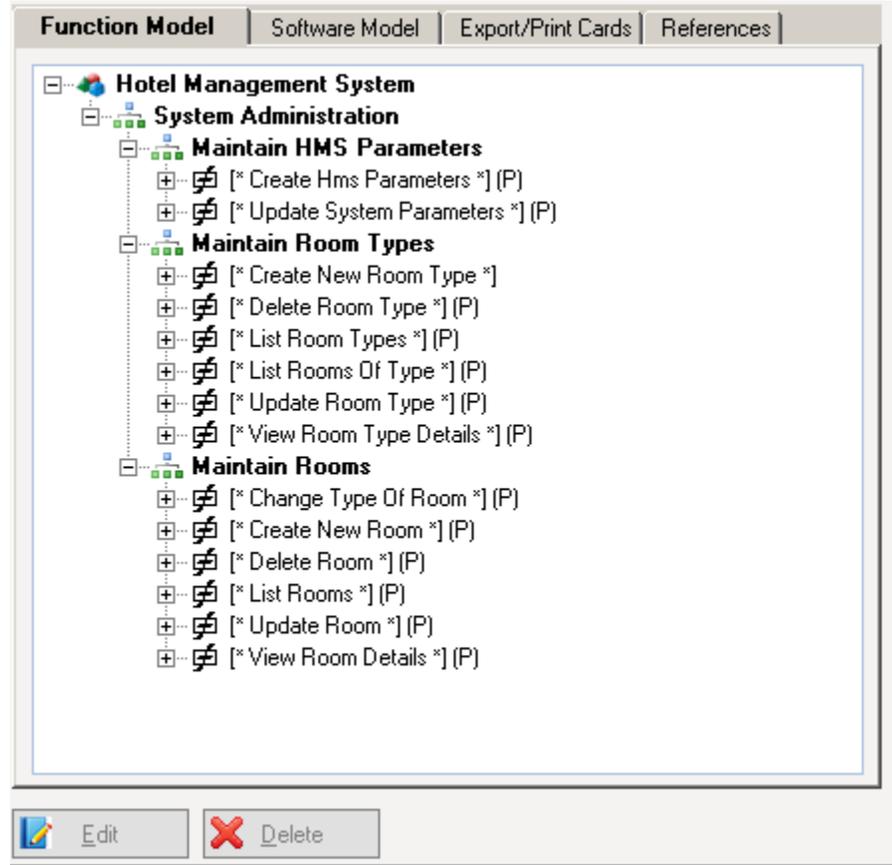


6. The Analysis Method dialog displayed, select **COSMIC**, fill in the detail and save.

   *VisualFSM will automatically create the COSMIC models and display the **COSMIC FSM Analysis** screen.*
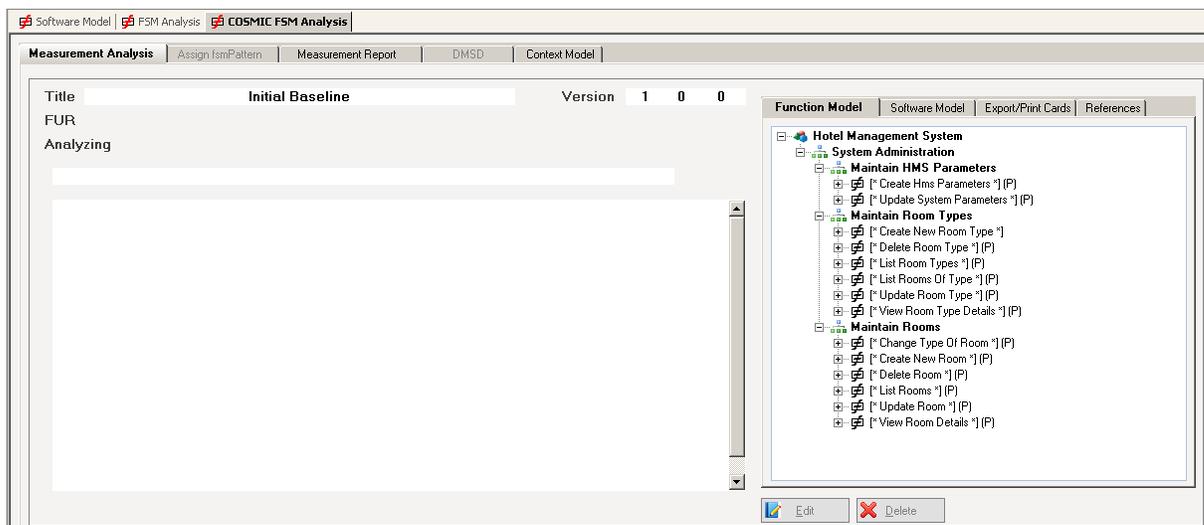
# Function Model

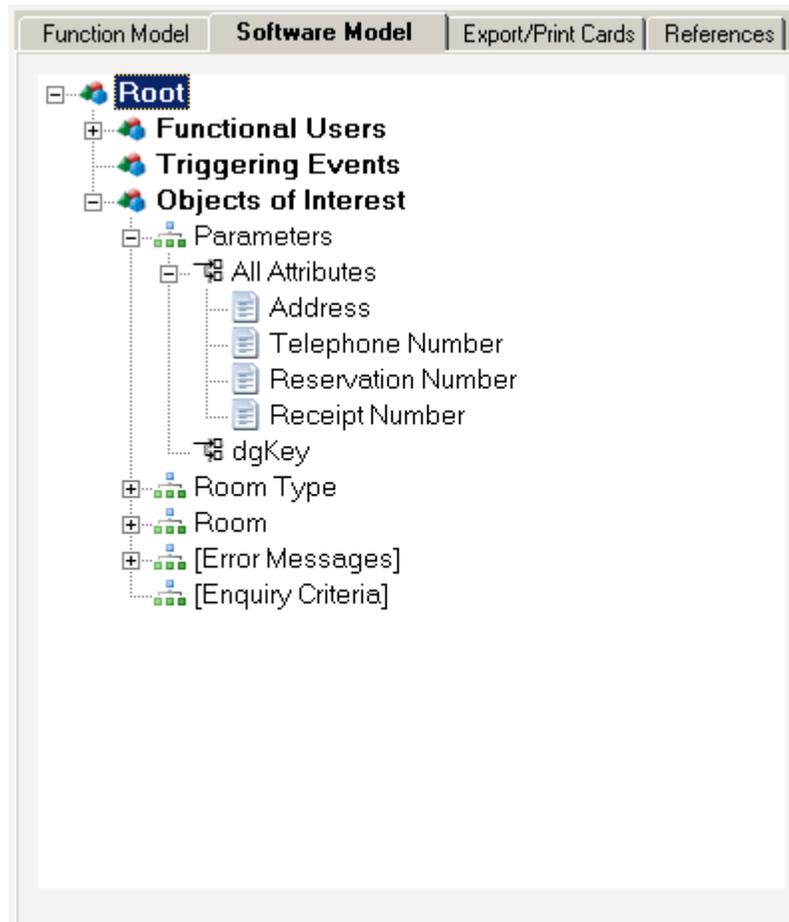VisualFSM creates a Function Process for each Functions.

## Visual Cues

Visual cues indicate the status of each Function

- **[* name *]**  indicates the FP has not yet been analysed

- If the name  is followed by **[P}** means that a fsmPattern has been assigned

## Software Model
The software model contains the "building blocks" for the measurement of the Functional

Processes



The Software Model catalogues the candidate Triggering Events, Functional users, Objects of Interest and Datagroups. Yjey are referenced when populating the Function Model

# VisualFSM Measurement Process

There are three step in the measurement process

- If an estimate is required, assign *fsmPatterns* to the FPs

- Construct the Software Model
    If necessary additional OOIs and Functional Users can be added

    Identify the Transient OOI and DGs. Transient OOI exist only for the lifetime of a

FP, therefore they can only be identified by analysing the FPs

- Analyse the FPs
  identify the Data Movements using the information in the Software model

# 3    Assign fsmPattern

## Introduction

This is an optional step and is provided to enable users quickly to derive an initial estimate of the size.

VisualFSM implements an estimating scheme called "Pattern Size".Software functions of the same type tend to have the same structure. VisualFSM calls these **fsmPatterns**© and it includes a module to manage fsmPatterns for the various domains and function type.

An initial fsmPattern Library is installed automatically and other fsmPatterns can be downloaded from the VisualFSM Solution Farm as they become available. In the Personal and Professional Editions, fsmPatterns can be can be manually added to the library and shared with other VisualFSM users.

This step allows the user to "assign" a predefined fsmPattern to each of the Functional Processes. VisualFSM will then report the "Pattern Size".

## Assign a fsmPattern to a Functional Process

1. Select the "**Assign fsmPat**tern" tab
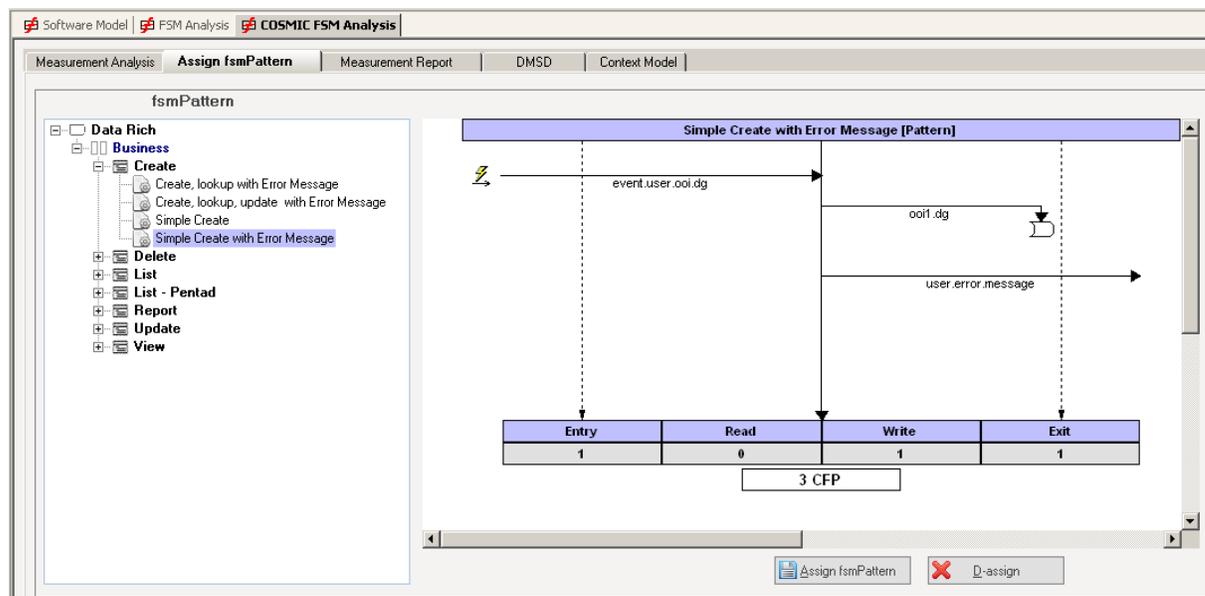


2. Select the Functional Process

3. Select the Pattern from the library

   a *thumbnail of the fsmPattern will be displayed*

4. Press the "**Assign fsmPattern**" button.

   *the name of the Functional Process is post-fixed with [P] to indicate a fsmPattern*

*has been assigned*

## Change a Previously Assigned Pattern

1. Select the Functional Process - the currently assigned pattern will be selected and displayed

2. Select the new pattern and assign it

## De-assign a Previously Assigned fsmPattern

1. Select the Functional Process - the currently assigned pattern will be selected and displayed

2. Press the "**De-assign**" button

*(the **[P]** post-fix to the name will be removed)*

## View an Assigned fsmPattern

1. Select the "***DMSD***" tab

2. The function model show each Functional Process, those with an assigned fsmPattern have the **[P]** post-fix to the title

3. Select the Functional Process

4. At the bottom of the DMSD select the "***fsmPattern View***" option

*The assigned fsmPattern and size will be displayed*

# 4    Construct Software Model

## Introduction

The size of a Functional Process is determined by the component parts of its Data Movements. VisualFSM adopts a very rigorous approach to measurement, each element of the model is traceable back to the Functional Requirements.

The purpose of this step is to identify and record the candidate elements of the Data Movements.  The Software Model comprises 3 sections, one for each

### Functional Users

VisualFSM automatically create a Functional User for each User in the User Model. Initially the user type is Generic. The type can be changed. This is not necessary, but aids in understanding the nature of the users and their interaction with the software.

### Triggering Events

The COSMIC Functional Process Model required that it respond to  events in the real world. It calls these Triggering Events. The software modeling process does not capture this information as it is a unique requirement of the COSMIC method; VisualFSM cannot generate these by reference to the models. However VisualFSM anticipates that in many cases, especially in Business Software a Functional Process is triggered as the result a *User Request* of some kind

VisualFSM can therefore automatically create a "User Request" event if the analyst elect to do so

### Objects of Interest and Data Groups

VisualFSM considers each File in the Logical Data Model to be a candidate OOI, The Analyst can elect to create what is called the "All" data group for the OOI and it will create a DET for each Datafield of the File used to create the OOI.

VisualFSM makes the distinction between an OOI that represents data that persists after the termination of a Functional Process, and an OOI that persists only for the duration of the Functional Process. The former is referred to as a Persistent OOI, the latter a "Transient OOI".

The OOI and Data Groups represent  the Persistent data maintained or referenced by the software via it functional Processes. The Transient OOIs and Data Groups may be identified when each Functional Process is analyzed in detail.

It is recommended that this step is performed iteratively by dealing with each Functional Process in turn by first completing the Software Model with respect to the Functional Process being measured then Complete the Function by creating the Data Movements.
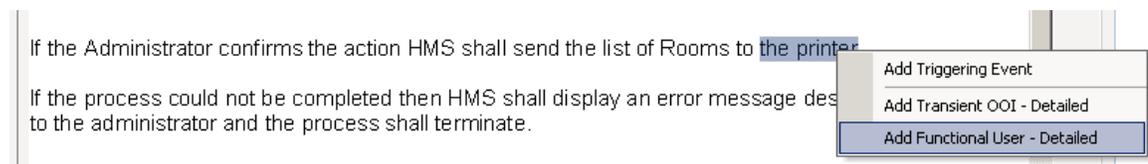
## 4.1    Functional Users

# Identify new Functional Users

All the Functional Users should have been identified when building the general software model, however they can be added at this stage.

In the "***List Room Type***" FP there is a need to send information to the printer, the print is therefore a Functional User.

1. Select ***Function Model*** Tab

2. Expand tree to reveal Functional Process level

3. ***LEFT***-click on the Functional Process to be analyzed

   *The FUR for the FP will be displayed*

4. Select the ***Software Model*** Tab

5. ***LEFT***-Click on the "***Functional Users***" to make this the focus of future actions and to set the correct context-sensitive menus

6. Highlight the text representing the new Functional user and ***LEFT***-Click on it

If the Administrator confirms the action HMS shall send the list of Rooms to the printer

If the process could not be completed then HMS shall display an error message des
to the administrator and the process shall terminate.

Add Triggering Event
Add Transient OOI - Detailed
Add Functional User - Detailed

7. The dropdown menu has one option "***Add Functional User***", select it, the Add User dialog is displayed
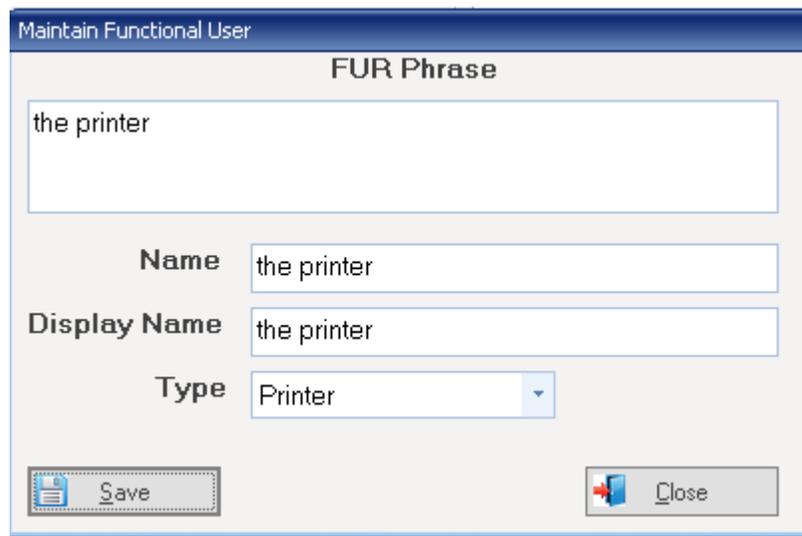
8. Given the user a name, assign a user-type (You would normally save iyt now, but it is already there, so skip this step)t

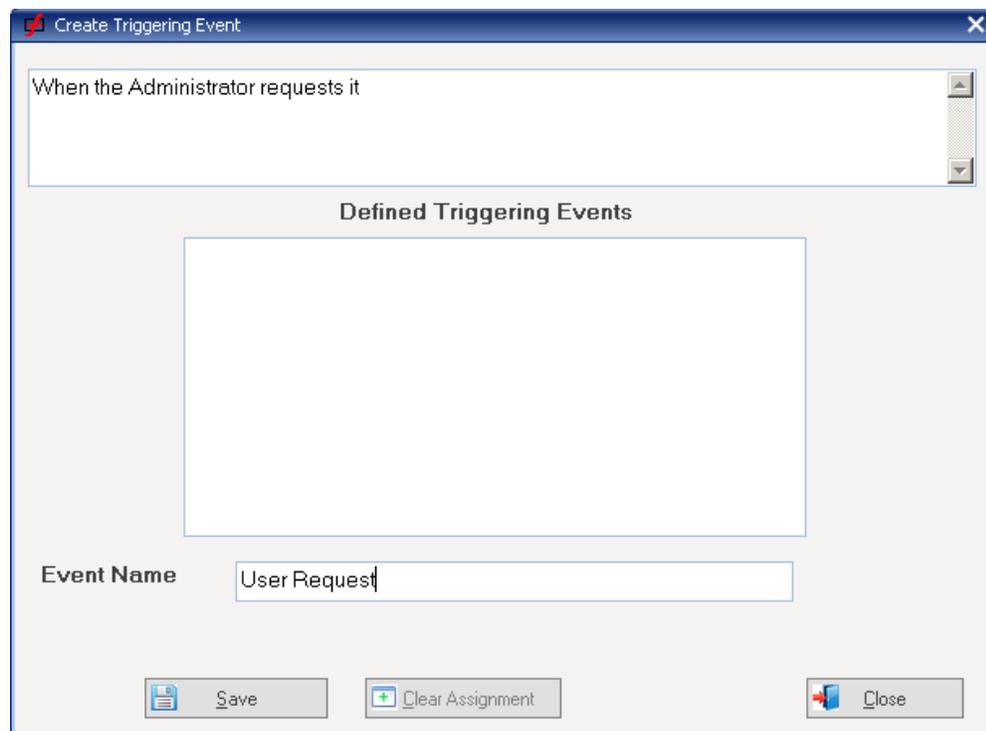   *The Software Model will be updated*

## Edit an Existing Functional User

1. If not already done, select the "***Software Model Tab***"

2. ***LEFT***-Click on the Functional User to be edited

3. The "***Edit***" button will be enabled, press it

   *The **Maintain Functional User** dialog will be displayed*

4. Edit the detail as required and save

5. The Software Model will be updated

## Delete an Existing Functional User

1. If not already done, select the "**Software Model Tab**"

2. **LEFT**-Click on the Functional User to be deleted

3. The "**Delete**" button will be enabled, press it

4. The **Confirm Dialog** will be displayed, confirm it

5. The Software Model will be updated

## 4.2     Triggering Events

## Identify new Triggering Events

1. Select **Function Model** Tab

2. Expand tree to reveal Functional Process level

3. **LEFT**-click on the Functional Process to be analyzed

   (this will display the FUR for the selected FP)

4. Select the **Software Model** Tab

5. **LEFT**-Click on the "**Triggering Events**" to make this the focus of future actions and to set the correct context-sensitive menus

6. Highlight the text representing the new Triggering Event and **LEFT**-Click on it

7. The dropdown menu has one option "**Add Triggering Event**", select it, the **Add Triggering Event** dialog is displayed

8. Either select an existing Triggering Event or give the Event a name and save it

9. The Software Model will be updated

## Edit an Existing Triggering Event

1. If not already done, select the "*Software Model Tab*"

2. *LEFT*-Click on the Triggering Event to be edited

3. The "*Edit*" button will be enabled, press it

4. The *Maintain Triggering Event* dialog will be displayed

5. Edit the detail as required and save

6. The Software Model will be updated

## Delete an Existing Triggering Event

1. If not already done, select the "*Software Model Tab*"

2. *LEFT*-Click on the Triggering Event to be deleted

3. The "*Delete*" button will be enabled, press it

4. The *Confirm Dialog* will be displayed, confirm it

5. The Software Model will be updated

## 4.3 Persistent Objects of Interest and Data Groups

## Identify new Persistent Object of Interest

In the Hotel Management Case Study used in the Quickstart, VisualFSM has automatically created all the persistent Objects of Interest.

To demonstrate how OOI would be  created  we shall create but not save anything

1. Select *Function Model* Tab

2.  Expand tree to reveal Functional Process level

3.  *LEFT*-click on the Functional Process to be analyzed

    (this will display the FUR for the selected FP))

4. Select the *Software Model* Tab

5*. LEFT*-Click on the "Objects of Interest" to make this the focus of future actions and to set the correct context-sensitive menus

6.  Highlight the text representing the new Object of Interest and *LEFT*-Click on it.

    *In this case study we have no unidentified OOIs, but to demonstrate the process select "Parameters"*

    *this will drop down a menu of possible option. in the current context*
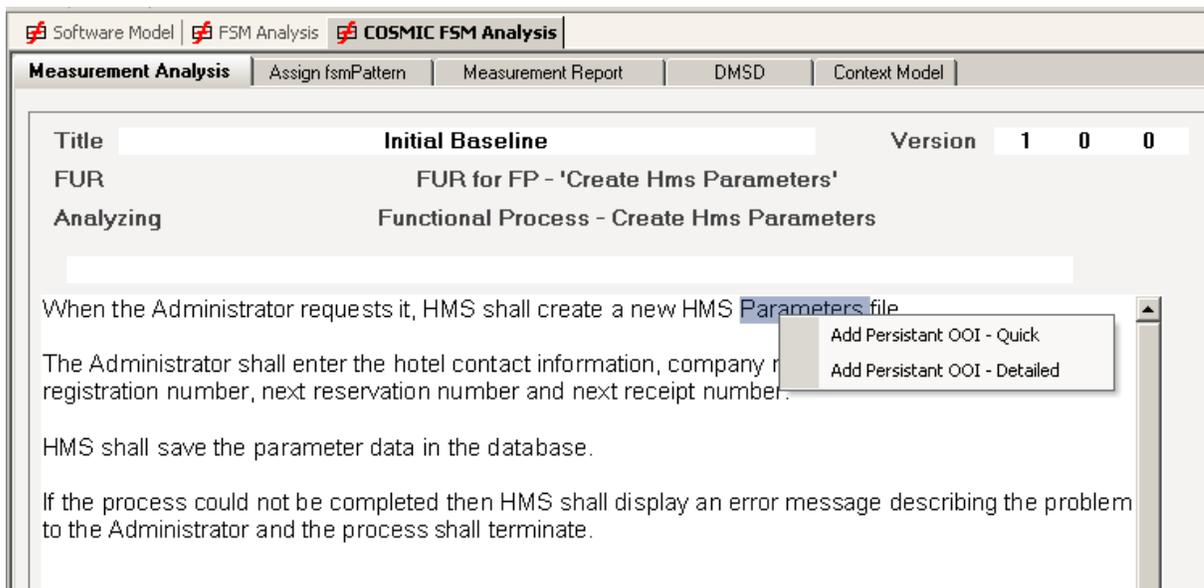
*The dropdown menu has options to add Quick (if the selection is the name of the OOI) or Detailed if some editing is required.*

7.  select "***Add Persistent OOI - Detailed***", Selecting this will display the Create OOI Dialog
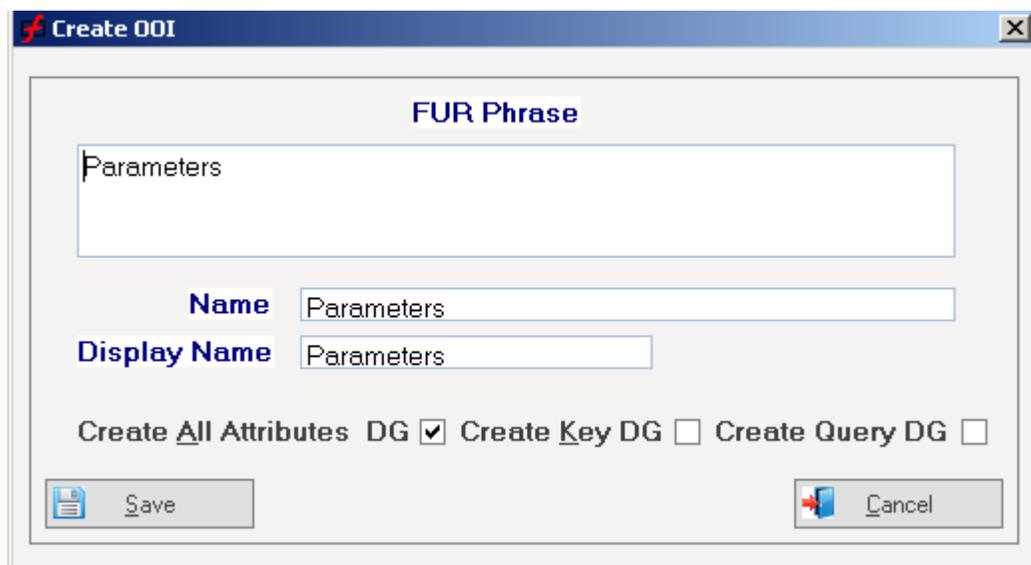


8.  Give the OOI a name

9.  Select the OOI type, either Persistent or Transient

10.  Check the box if you want VisualFSM to create a default "All" Data Group

11.  Check the box if you want VisualFSM to create a default "Key" Data Group representing the unique identity "Key" of the OOI

12.  You would new press *Save*, but we have already identified it so skip this step

*The Software Model will be updated*

# Add a Data Group

If a new Datagroup were needed ...

1.  **LEFT**-Click on the **Object of Interest** for the new Data Group to make this the focus of future actions and to set the correct context-sensitive menus

2.  Highlight the text representing the new Data Group and **LEFT**-Click on it

3.  The dropdown menu has several options, select  "*Add Datagroup to OOI*" , the Add Datagroup dialog will be displayed
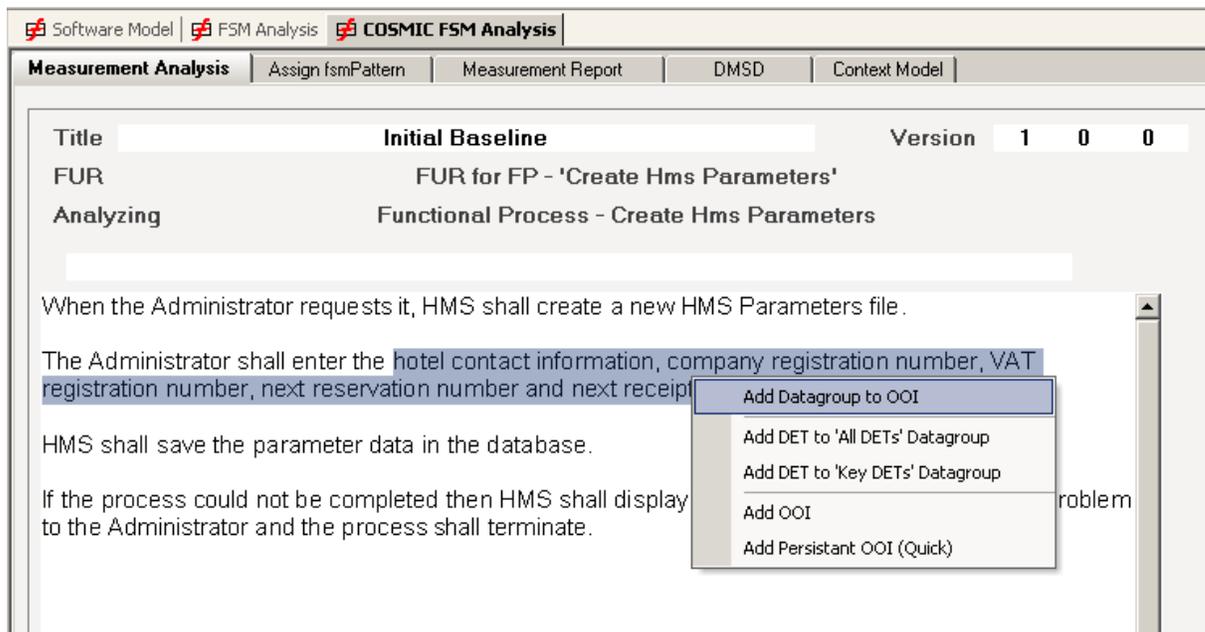


*The Create Datagroup dialog will be displayed*

5. Give it a name

6. We would normally press **Save** but we will skip this step

## Add an Attribute to a Datagroup (optional)

1. **LEFT**-Click on the Datagroup for the Attribute to make this the focus of future actions and to set the correct context-sensitive menus

2. Highlight the text representing the new Attribute (it should be the name of the Attribute as no Dialog will be shown

3. **LEFT**-Click on it, the Attribute will be created

## Edit an Existing Object of Interest

1. If not already done, select the "*Software Model Tab*"

2. **LEFT**-Click on the Object of Interest to be edited

3. The "*Edit*" button will be enabled, press it

4. The Maintain Object of Interest dialog will be displayed

5. Edit the detail as required and save

## Delete an Existing Object of Interest

1. If not already done, select the "*Software Model Tab*"

2. **LEFT**-Click on the Object of Interest to be deleted

3. The "*Delete*" button will be enabled, press it

4. The Confirm Dialog will be displayed, confirm it

5. The Software Model will be updated

# Edit an Existing  Datagroup

1. If not already done, select the "*Software Model Tab*"

2. **LEFT**-Click on the Datagroup  to be edited

3. The "*Edit*" button will be enabled, press it

4. The Maintain Datagroup dialog will be displayed

5. Edit the detail as required and save

6. The Software Model will be updated

# Delete an Existing Datagroup

1..If not already done, select the "*Software Model Tab*"

2. LEFT-Click on the Datagroup to be deleted

3. The "*Delete*" button will be enabled, press it

4. The Confirm Dialog will be displayed, confirm it

# Edit an Existing  Attribute

1..If not already done, select the "*Software Model Tab*"

2. **LEFT**-Click on the DET  to be edited

3. The "*Edit*" button will be enabled, press it

4. The Maintain DET dialog will be displayed

5. Edit the detail as required and save

# Delete an Existing Attribute

1. If not already done, select the "*Software Model Tab*"

2. **LEFT**-Click on the Datagroup to be deleted

3. The "*Delete*" button will be enabled, press it

4. The Confirm Dialog will be displayed, confirm it

# 5    Construct the Function Model

## Introduction

Now that all the information for creating the Data Movements has been captured, the purpose of this step is for each Functional Process:to construct the Function Model .
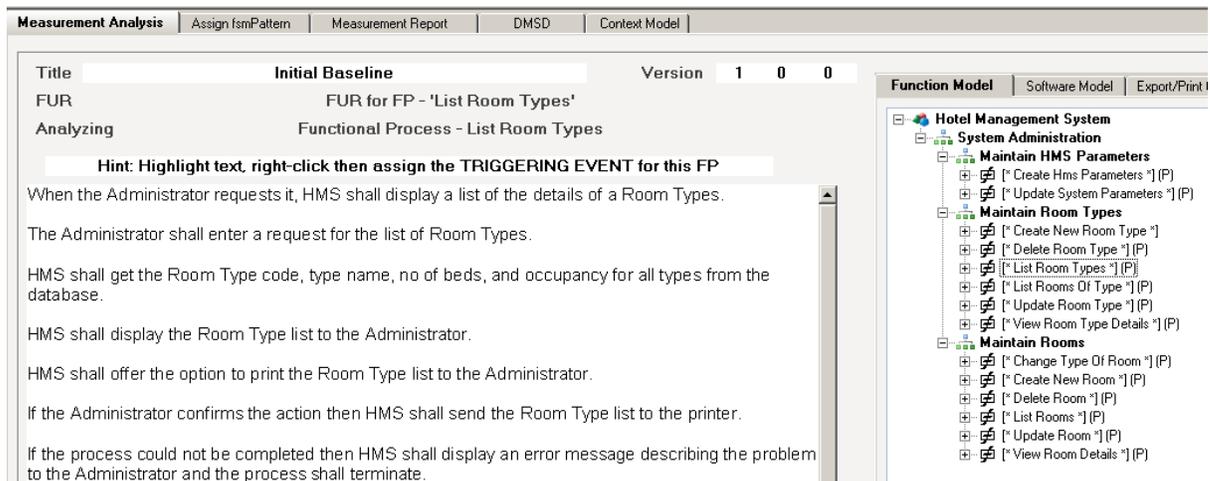
- Identify the Triggering Event

- Identify the Transient OOIs if any

- identify the Data Movements

Select each Functional Process in turn and follow the process for the above activities.

## Select the Functional Process

1. Select **Function Model** Tab

2. Expand tree to reveal Functional Process level

3. **LEFT**-click on the "**List Room Types**" FP.

    This will load the FUR for the Functional Process



*To assist the beginner, VisualFSM will display hints that advise the analyst what to do next in a given context.*

*In this case the FP has not yet been analysed so  VisualFSM is saying the next step is to assign the Triggering Event*

## 5.1 Identify the Transient Objects of Interest

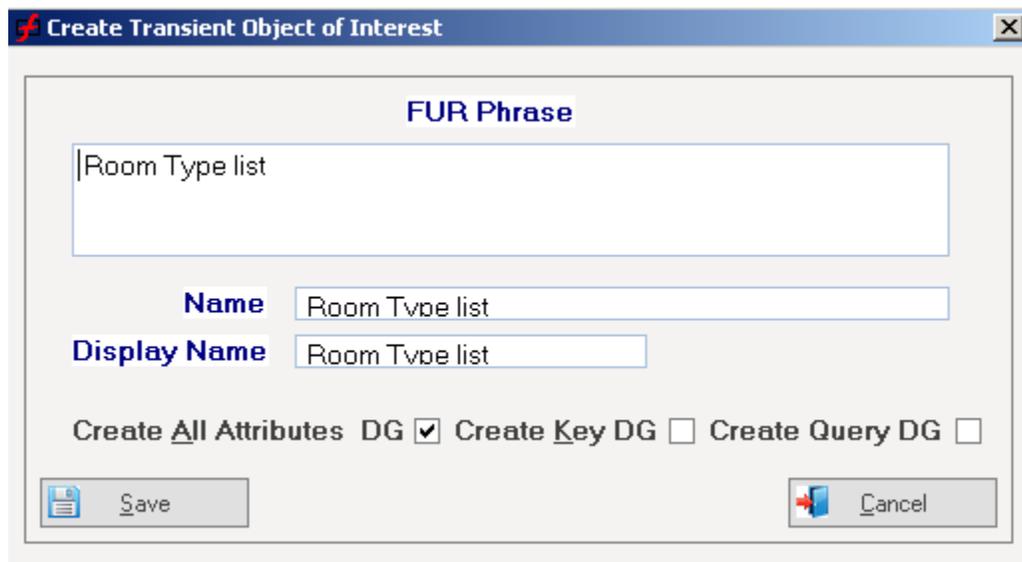### Identify the Transient Objects of Interest

1. Scan the FUR and find the statement that describes the transient OOI

2. Highlight the text and **RIGHT**-Click it .A dropdown menu item is displayed

**Create Transient Object of Interest**

**FUR Phrase**

Room Type list

**Name** Room Type list
**Display Name** Room Type list

Create **A**ll Attributes DG ☑ Create **K**ey DG ☐ Create Query DG ☐

Save                    Cancel
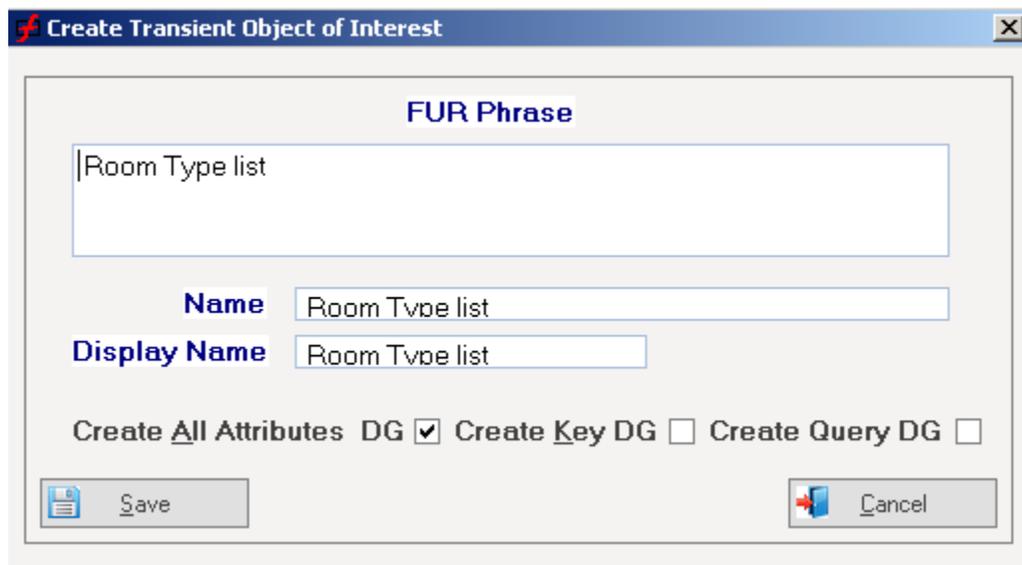
3. Select the **Add Transient OOI - Detailed** option

The **Create Transient OOI** Dialog will be displayed

**Create Transient Object of Interest**

**FUR Phrase**

Room Type list

**Name** Room Type list
**Display Name** Room Type list

Create **A**ll Attributes DG ☑ Create **K**ey DG ☐ Create Query DG ☐

Save                    Cancel

4. Give a meaningful name
    *Note: You may optionally create default Datagroups at this time by checking the*

*appropriate boxes.*

5. Press **Save**

   ***The Transient OOI will be added to the Function and***

**Add the Attributes**

1. Select the **Room Type List** datagroup

2. To add a single Attribute, highlight the text representing the name of the Attribute

3. **LEFT**-click on the selection  to show the menu
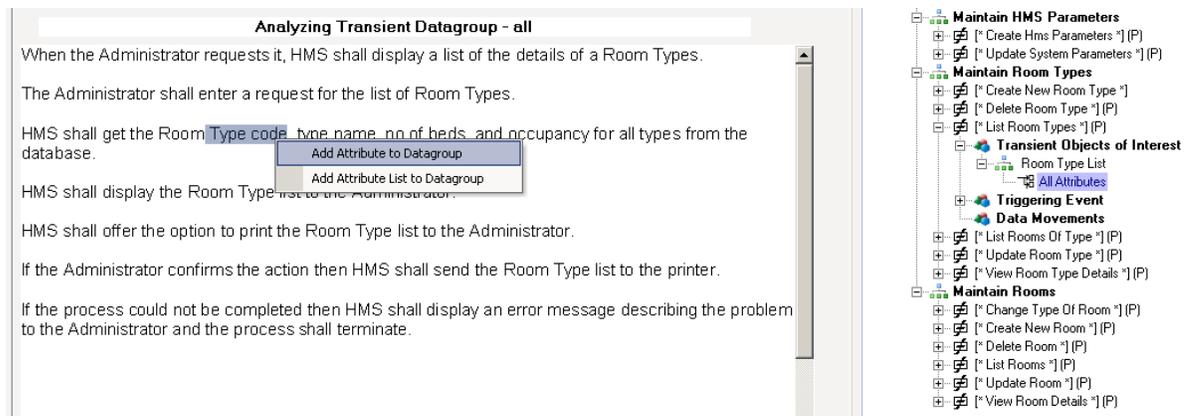


4. select "Add Attribute to Datagroup"
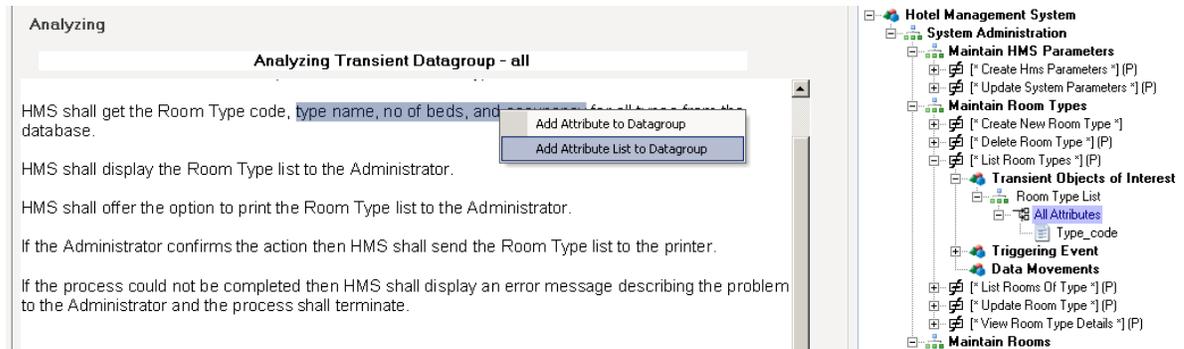   *An attribute with the selected name will be created and added to the Function Model*

*tree*
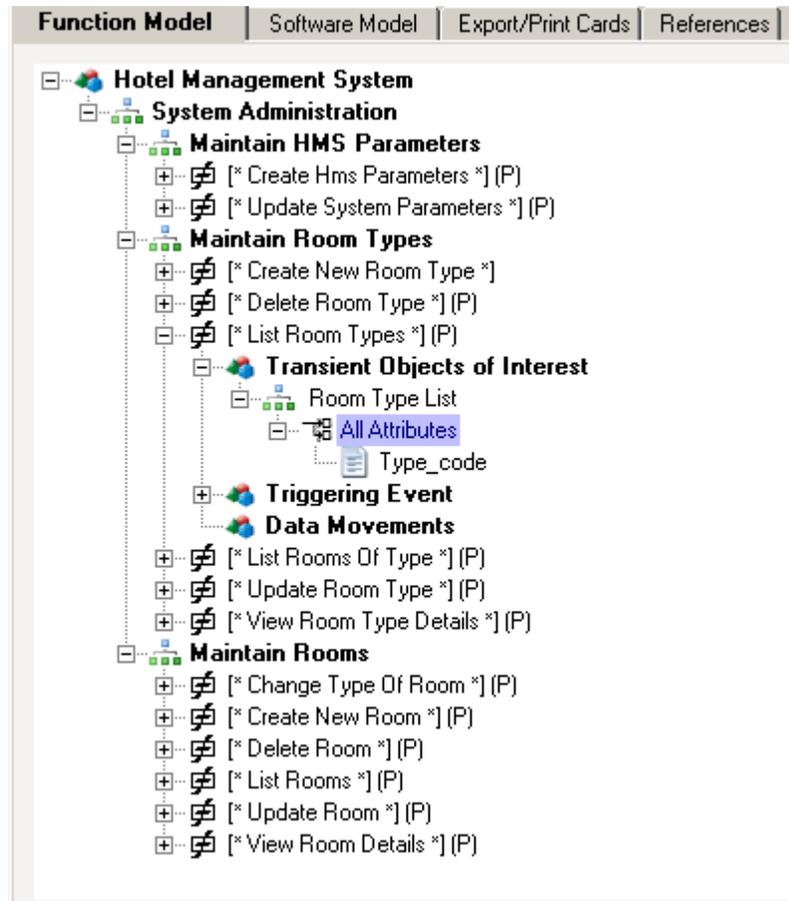


2. To add a comma-separated list of Attributes, highlight the list representing the names

of the set of Attributes

3. **LEFT**-click on the selection  to show the menu



4. Select "Add Attribute List to Datagroup"
   *attribute with the names in the list will be created and added to the Function Model*

*tree*
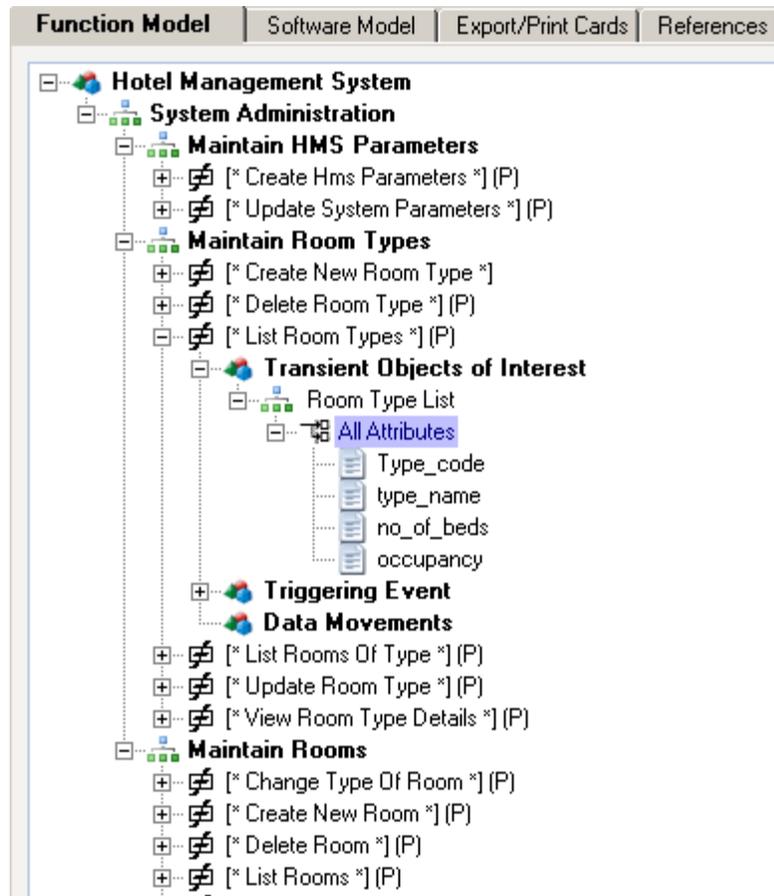


The FP is required to produce a list, This is a type of Query  We could have create the Query DG when we created the Transient OOI, however we can do it now

## Create the Query Datagroup

1.  Select the Room Type List OOI

2. Highlight the text that represents the  Datagroup that will form the Query.
    Hint, we have to have a reference in the FUR,  select the action that initiates the FP
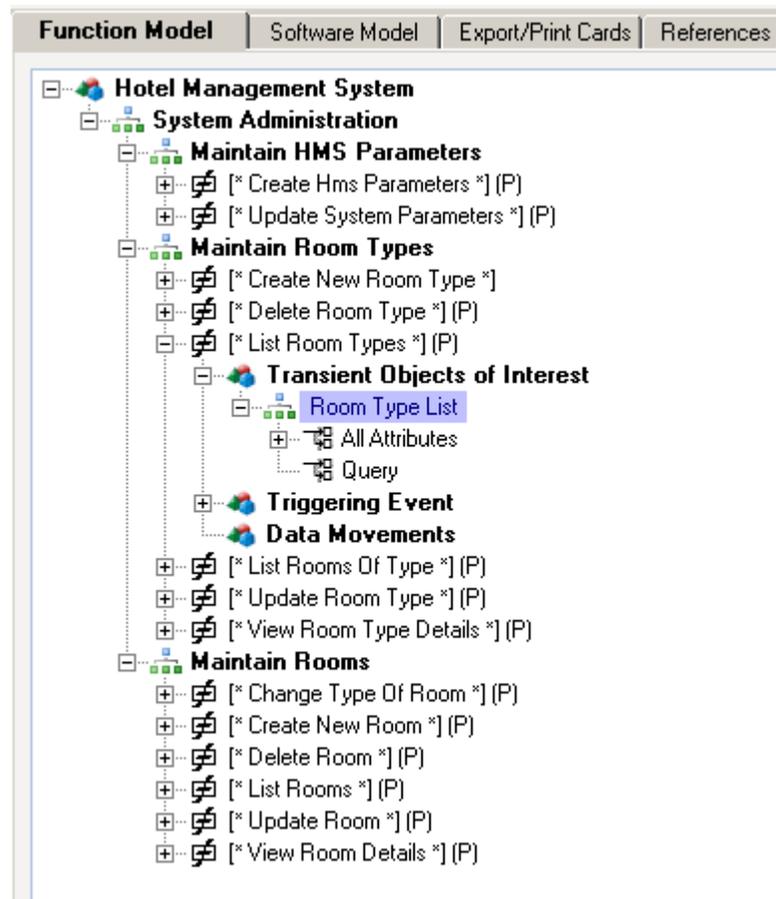
we will rename it next
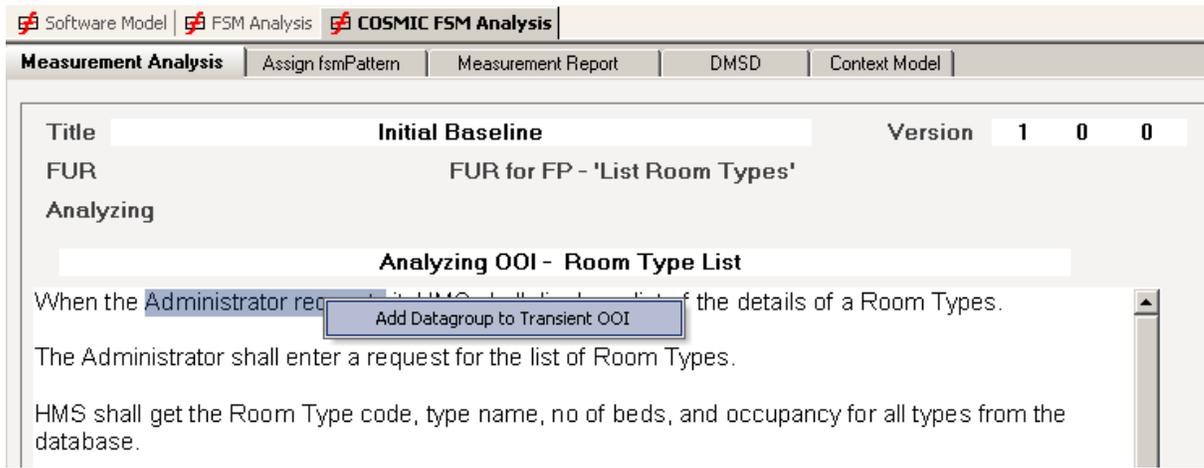


*The Add Datagroup Dialog will be displayed*



3. Rename the Datagroup to "Query"
4. Press **Save**
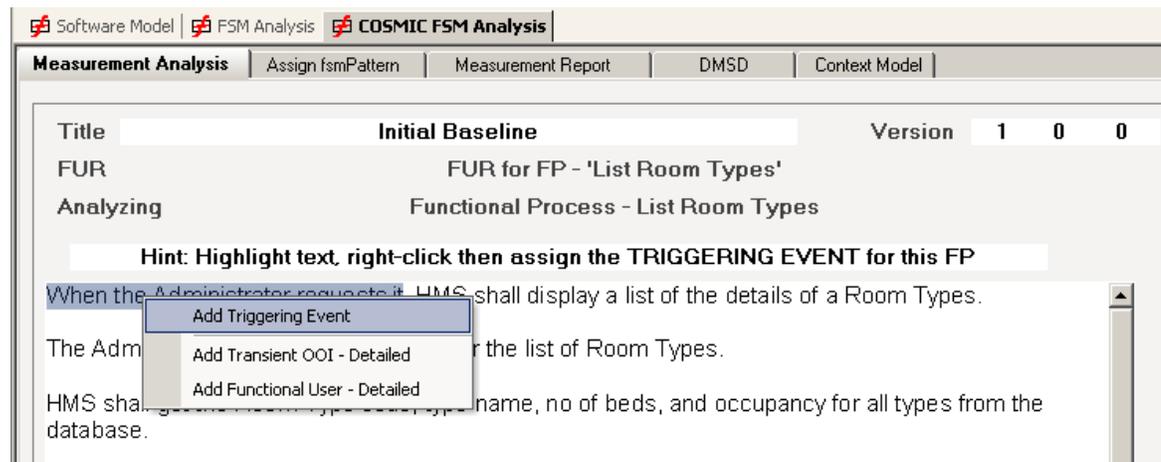
*The Datagroup will be added to the Function Model*

Next Step,  identify the Triggering Event for the selected FP

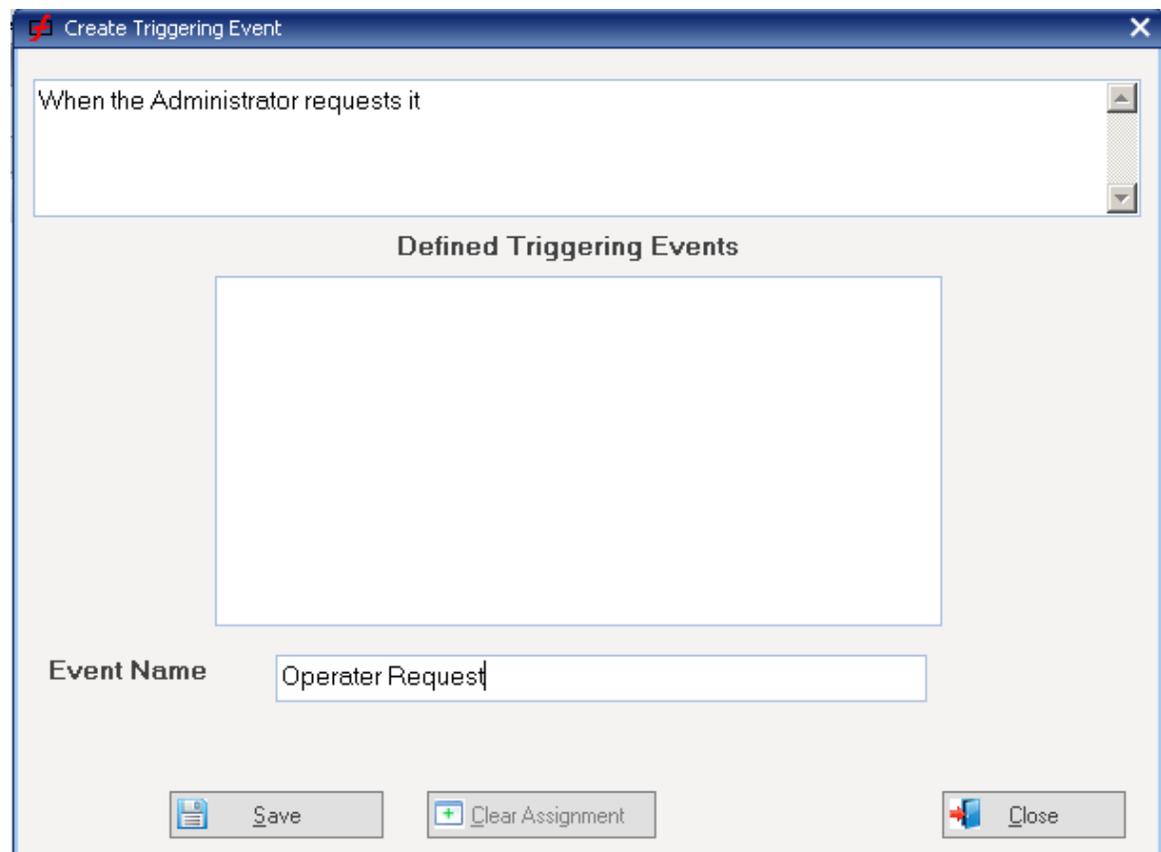## 5.2    Identify the Triggering Event

# Identify the Triggering Event

1.  Scan the FUR and find the statement that describes the input by the user that initiates the Functional Process

2.  Highlight the text and **RIGHT**-Click it .A dropdown menu item is displayed



3.  Select it and the Add Triggering Event Dialog is displayed.



7.  Either select an existing Event from the List, or enter the name of a new one
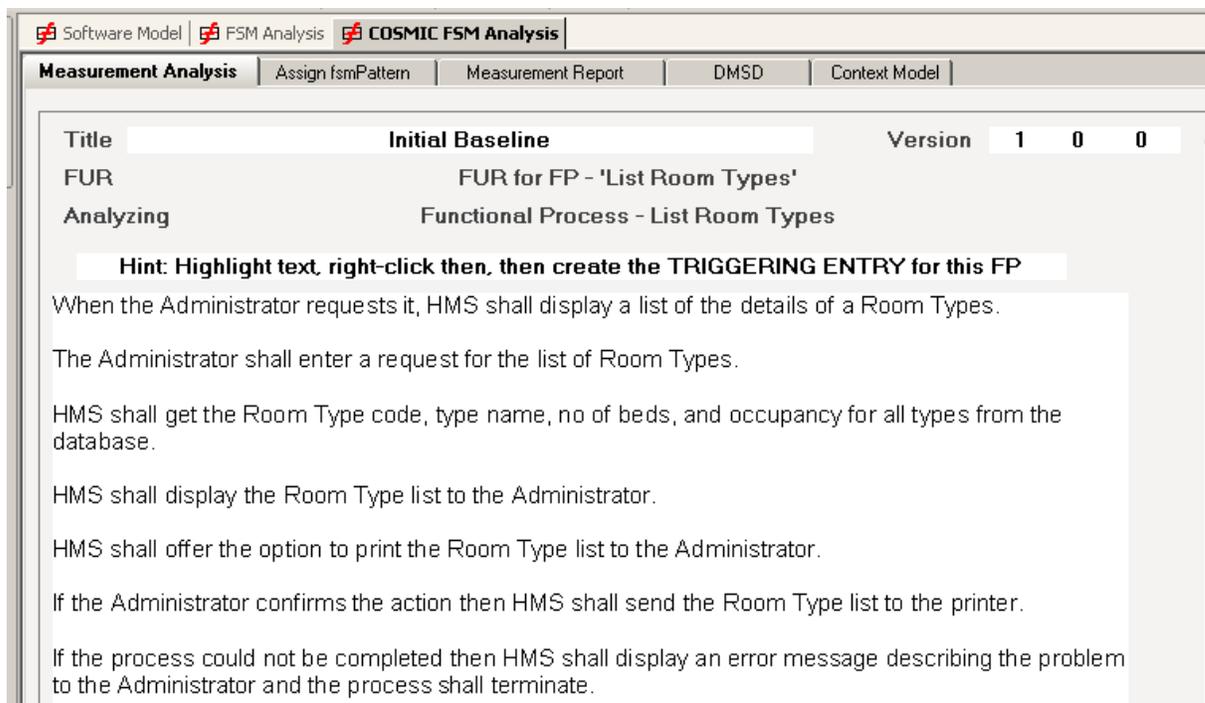
8. Press **Save**

*The Triggering event will be added to the Function model*

Next step: Identify the Triggering Entry

## 5.3    **Identify the Triggering Entry**

# Introduction

VisualFSM enforces the  requirements of a FP,  and the rule is that an FP is triggering by a Triggering Entry (TENT). It will display a hint id that task has not yet been completed



To Create the Triggering Entry...

1. Select the *List Room Types* FP

2.  Highlight the text representing the Datagroup moved to trigger the FP

3.  LEFT-click on the selection to show the mene

4.  Select  "*Add Triggering Entry*"

The Data Movement - Triggering Entry will be displayed with the selected text as an aide-memoir for the analysts
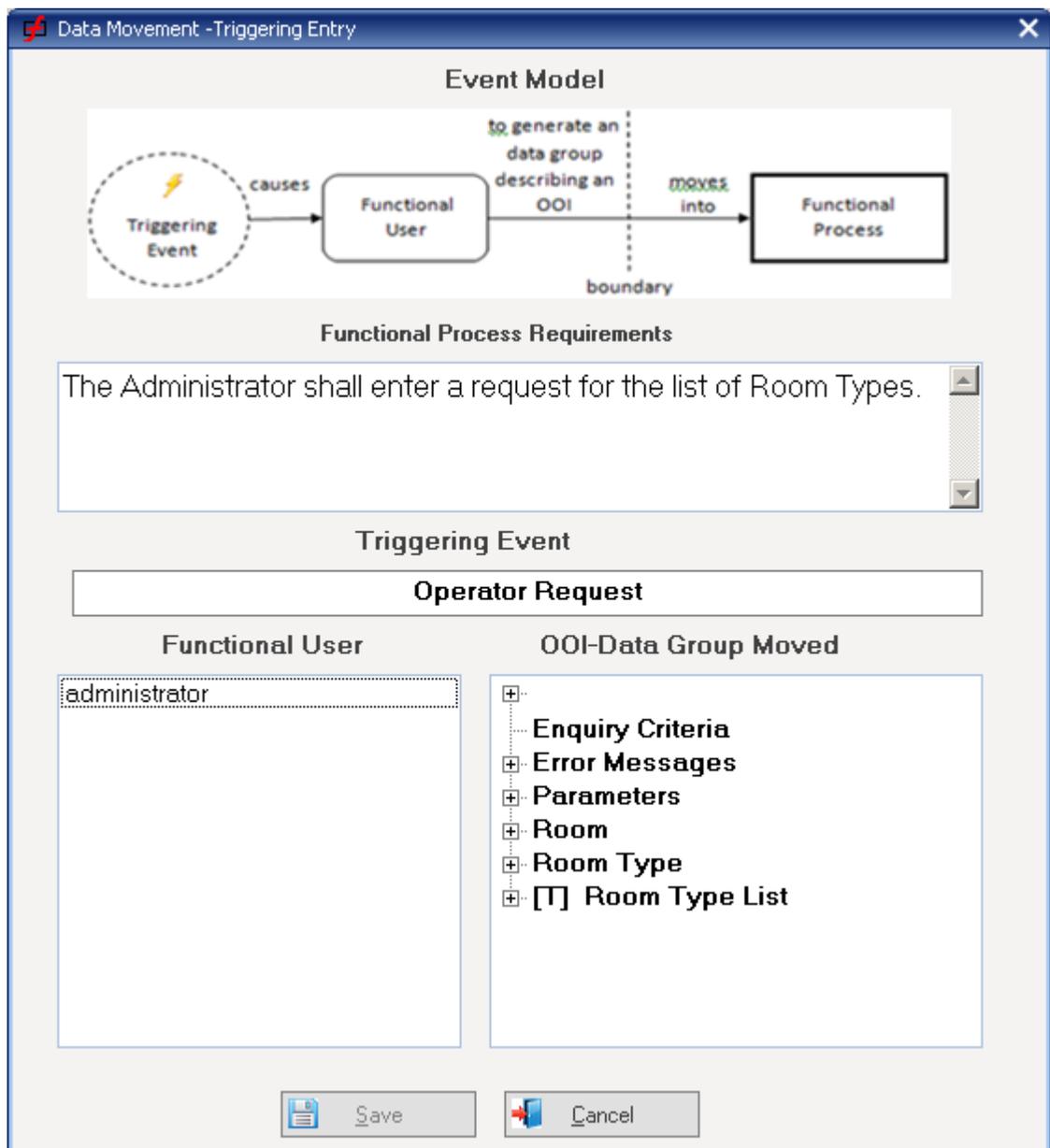
5. Select the Functional User (in this case the Administrator)

6, Expand the transient OOI in the tree (in this case the Room Type List)
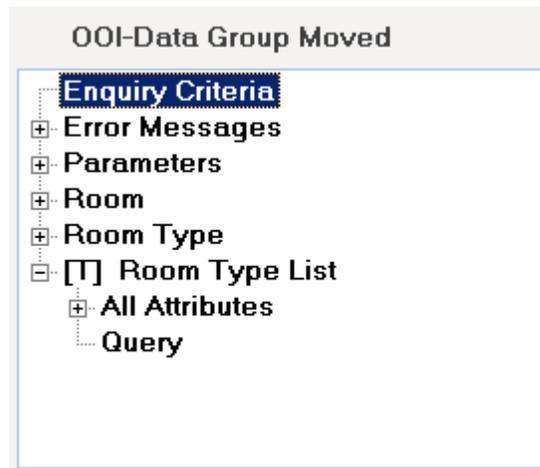
7. Select the Query Datagroup

8. Press **Save**

*the Data Movements is created and added to the function Model*
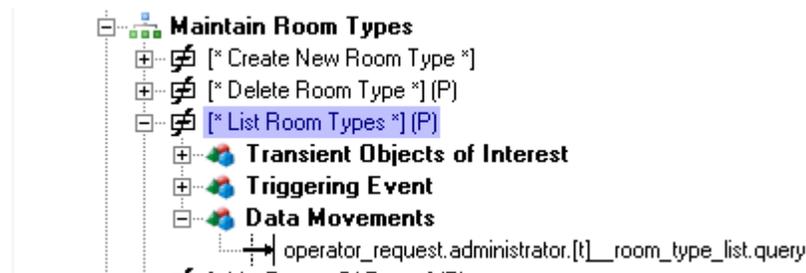


Next Step :

## 5.4    Identify the Data Movements

### Introduction

Scan the FUR and identify the other Data Movements. There is a different Dialog for each DM Type, Entry, Exit, Read and Write. They are similar to the one used to capture the Triggering Entry with a diagram of the model being constructed, and a set of lists of item appropriate to the DM Type as follows:

- **Entry and Exit** .    Function User, Object of Interest and Datagroup
- **Read and Write**.    Object of Interest and Datagroup

VisualFSM uses a DOT notation to name a DM comprising the names of each element, separated by "."  (see the entry in the Function Model for the Triggering Entry just created)

## Process
As an example we will create the next DM.

1. Highlight the text representing a DM

Software Model | FSM Analysis | **COSMIC FSM Analysis**

| Measurement Analysis | Assign fsmPattern | Measurement Report | DMSD | Context Model |

| Title | **Initial Baseline** | Version | **1** | **0** | **0** |
| FUR | FUR for FP - 'List Room Types' |
| Analyzing | Functional Process - List Room Types |

**Hint: Highlight text, right-click then create a DATA MOVEMENT for this FP**

When the Administrator requests it, HMS shall display a list of the details of a Room Types.

The Administrator shall enter a request for the list of Room Types.

HMS shall get the Room Type code, type name, no of beds, and occupancy for all types from the database.

HMS shall display the Room Type list to the Administrator.

HMS shall offer the option to print the Room Type list to the Administrator.

If the Administrator confirms the action then HMS shall send the Room Type list to the printer.

If the process could not be completed then HMS shall display an error message describing the problem to the Administrator and the process shall terminate.

*Note: The analyst must decide which type of DM it is, In this case information is being retried fro a the Database, so it is a **READ**.*

2.  **LEFT**-Click of the selection

   *(the menu will now offer a new set of options*)

| Title | Initial Baseline | Version | 1 | 0 | 0 |

FUR — FUR for FP - 'List Room Types'

Analyzing — Functional Process - List Room Types

**Hint: Highlight text, right-click then create a DATA MOVEMENT for this FP**

When the Administrator requests it, HMS shall display a list of the details of a Room Types.

The Administrator shall enter a request for the list of Room Types.

HMS shall get the Room Type code, type name, no of beds, and occupancy for all types from the datab

- Add ENTRY
- Add READ
- Add WRITE
- Add EXIT
- Add Transient OOI
- Add Persistent OOI
- Add Functional User

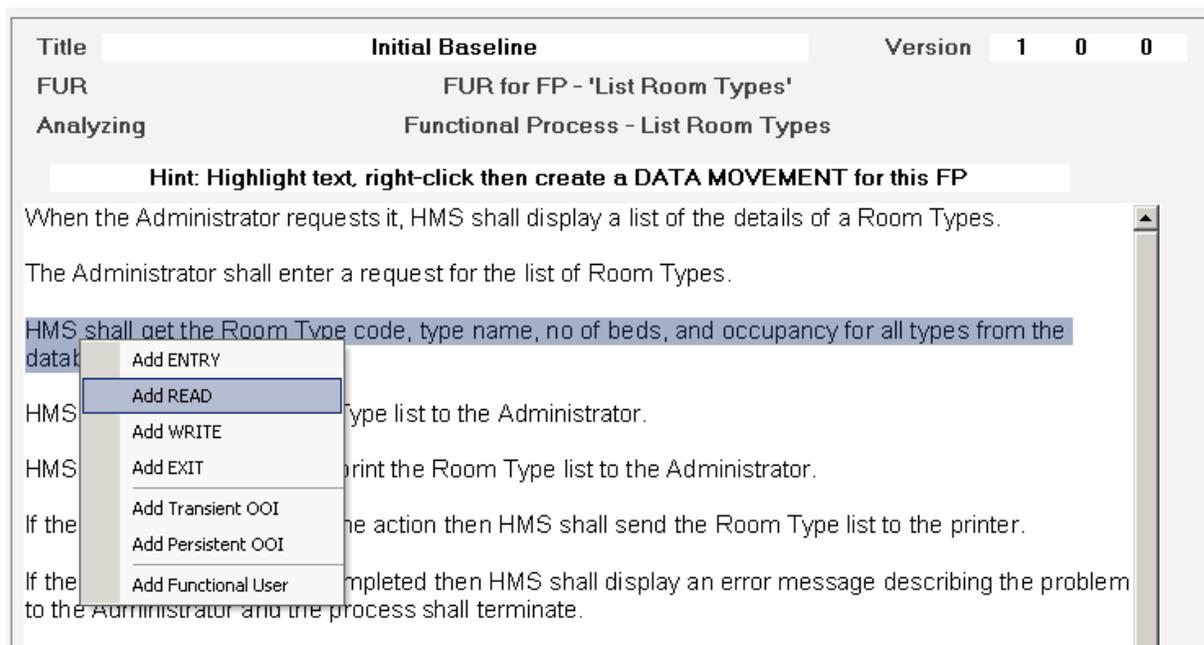HMS [ ] Type list to the Administrator.

HMS [ ] print the Room Type list to the Administrator.

If the [ ] he action then HMS shall send the Room Type list to the printer.

If the [ ] mpleted then HMS shall display an error message describing the problem to the Administrator and the process shall terminate.

The RAD dialog is displayed, To be absolutely correct we should really create a Datagroup of the Attribute actually retrieved from the Database, In almost all circumstances it is sufficient to retrieve all the Attribute (please refer to the identification rules for ant ENTRY DM for an explanation)

3. Select the OOI and DG (In this case the **Room Type** OOI and the **All** Datagroup)
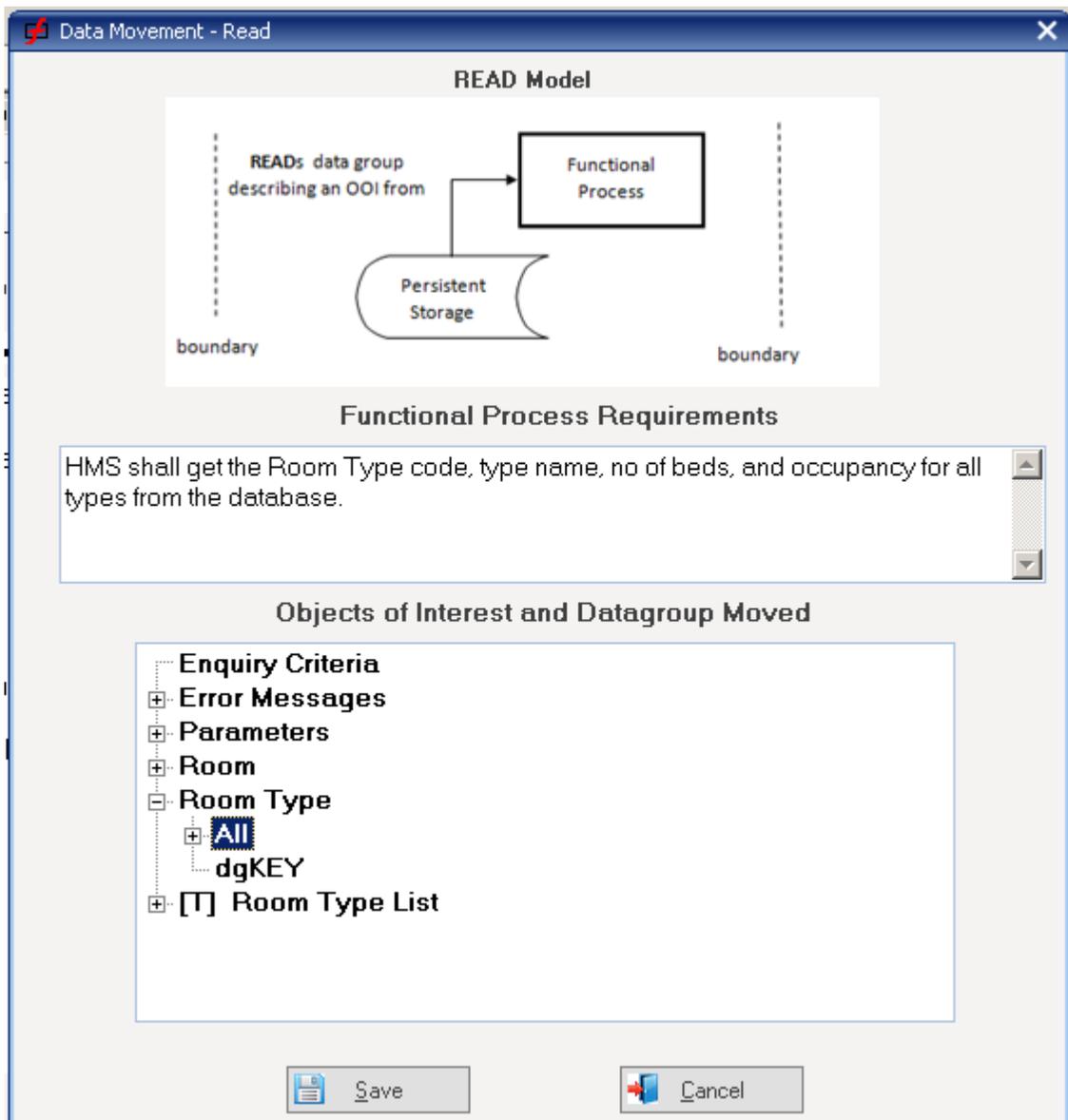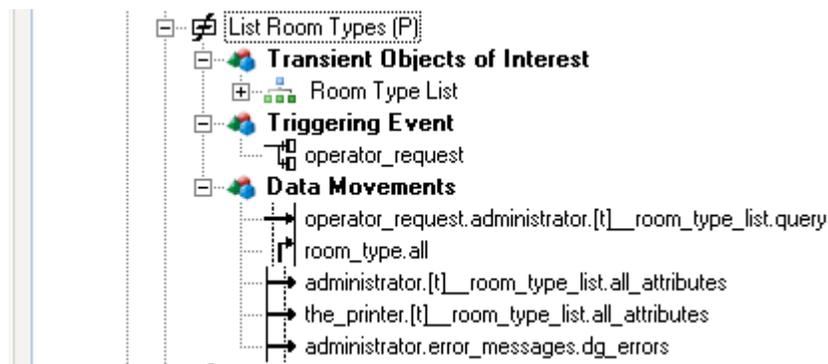
4.  Press **Save**

    *the READ DM is added to the Function Model*
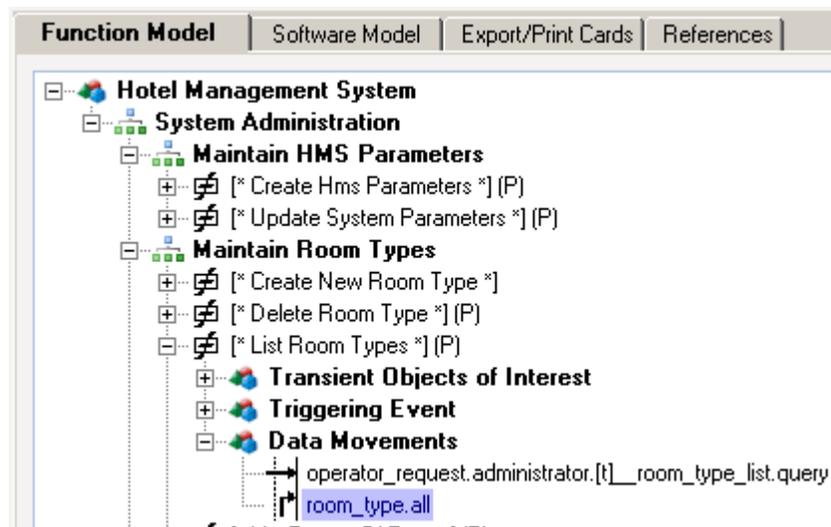
Repeat the process for the remaining DMs and the final Function Model for the *List Room Types* FP should look like this
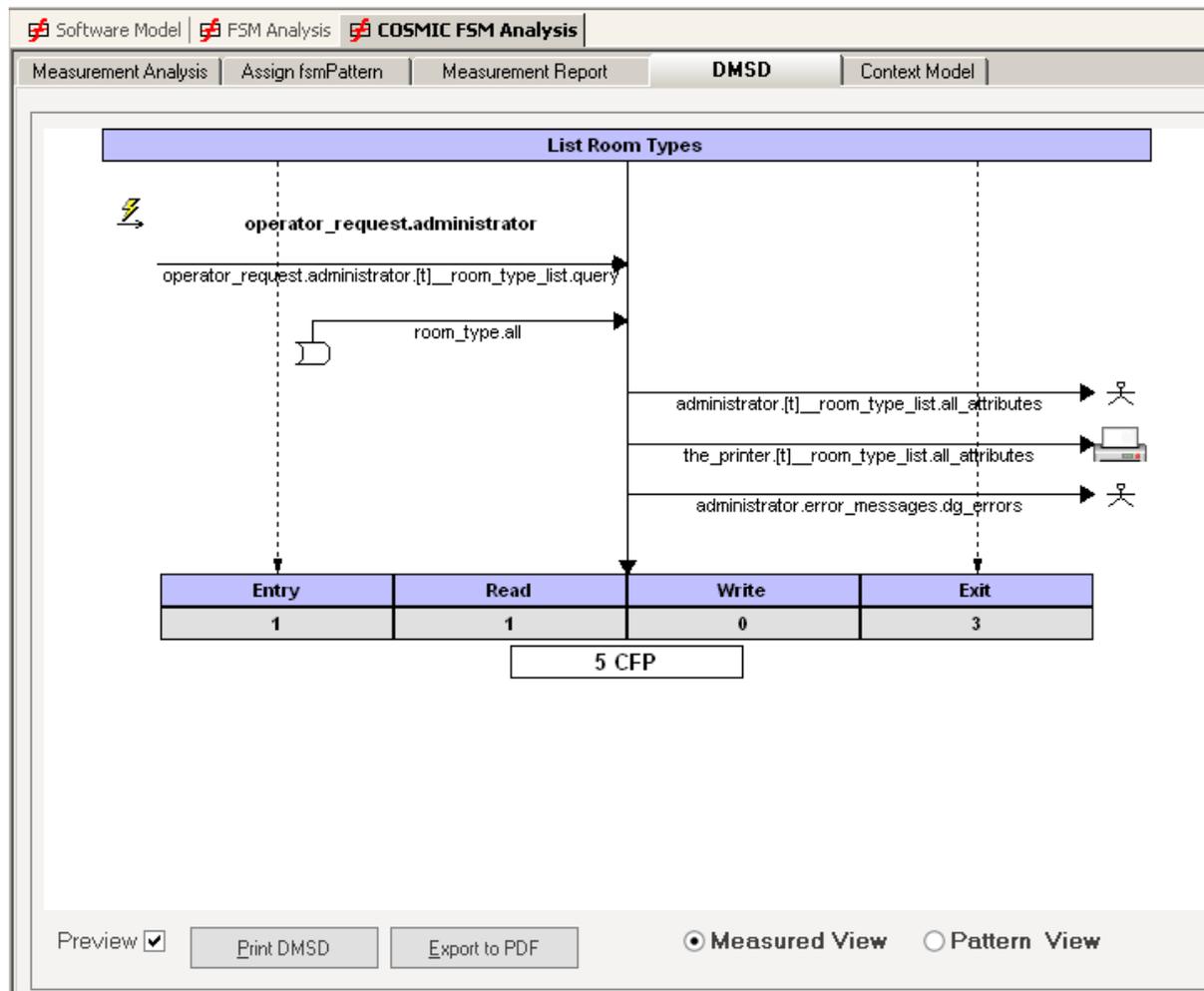


Repeat the process for all Functional Processes

# 6     View the Data Movement Sequence Diagram

## Introduction

The Data Movement Sequence Diagram is a  diagrammatic representation of the Functional Process unique to VisualFSM. It was designed to assist counting the contribution of each individual Data Movement Type and from that deriving the total size.



It emulates the way a spreadsheet works by placing each data movement type in separate column. It is a simply matter then to total each column and add the totals together, something that users are familiar with.

Although each Data Movement Type contribute 1 to the size,  it is necessary to separate them as there are rules to prevent duplication and over-counting. (see the COSMIC Measurement Manual). VisualFSM will detect and warn that duplication has occurred.

Although sequence is not essential for size measurement it does assist in validating that

the analysis is correct and complete.  Selecting a Datamovement  and simultaneously holding down the SHIFT key enables the user to move a Data Movement up or down the sequence.

## DOT Notation

The DMSD uses the VisualFSM DOT notation to identify each Data Movement. The notation separates each element of a Data Movement by a full-stop as follows

### Triggering Entry

*eventName.functionalUserName.ooiName.datagroupName*

### Entry and Exit

*functionalUserName.ooiName.datagroupName*

### Read and Write

*ooiName.datagroupName*

This assists in understand the nature of each Data Movement and exposes any duplicates which if intended may have to be explained

# 7 View the  Measurement Report

## Introduction

 Measurement report contains the size analysis of each Functional Process in the scope of measurement, aggregated by its Functional Breakdown Level etc.

## Sections

Select the "Measurement Report" tab. The
There are 3 Sections in the report, shown by selecting the appropriate Tab.

**Measured Size**. This is the size of the Functional process derived by identifying the actual Data Movements.



**fsmPattern Size.** If fsmPatterns have been assigned in Step 1 the report will show

the fsmPattern Size of those Functional Processes



**Composite Size**.  This part of the report shows the Measured size for the Functional Processes that have been measured, and the fsmPattern size for those that have not yet been measured but have been assigned a fsmPattern . This are indicated by

showing "**[P]**" in the final column.

| Measured Size | fsmPattern Size | Composite Size | | | | | |
|---|---|---|---|---|---|---|---|
| **Application** | Hotel Management System – Qui... | | | | | | |
| **Layer** | Application | | | | | | |
| **Component** | Hotel Management System | | | | | | |
| | | | | | | | |
| **FBS** | **FP** | **ENTRY** | **READ** | **WRITE** | **EXIT** | **SIZE** | |
| | | | | | | | |
| System Administration | | | | | | | |
| Maintain HMS Parameters | | | | | | | |
| | Create Hms Parameters | 2 | 0 | 2 | 2 | 6 | [p] |
| | Update System Parameters | 2 | 0 | 2 | 2 | 6 | [p] |
| Maintain Room Types | | | | | | | |
| | Create New Room Type | | | | | | |
| | Delete Room Type | 2 | 0 | 2 | 2 | 6 | [p] |
| | List Room Types | 1 | 1 | 0 | 3 | 5 | [m] |
| | List Rooms Of Type | 2 | 2 | 0 | 4 | 8 | [p] |
| | Update Room Type | 2 | 0 | 2 | 2 | 6 | [p] |
| | View Room Type Details | 2 | 2 | 0 | 4 | 8 | [p] |
| Maintain Rooms | | | | | | | |
| | Change Type Of Room | 2 | 0 | 2 | 2 | 6 | [p] |
| | Create New Room | 2 | 0 | 2 | 2 | 6 | [p] |
| | Delete Room | 2 | 0 | 2 | 2 | 6 | [p] |
| | List Rooms | 2 | 2 | 0 | 6 | 10 | [p] |

Export to Excel    Export Detailed to Excel    Quick Print

This is useful for occasion where a sample is measured in order to gain confidence in the use of the fsmPattern Size as both sizes can be compared.

# 8 Reporting

## Introduction

VisualFSM can, depending on the context, print reports. expport report to PDF or export reports in Microsoft Excel format

## DMSD Cards

- Print Individual Functional Process. Prints the diagrams current selected Functional Process
  Select the Functional process to be printed
  Press the Print Button

- Export Individual Functional Process. Prints the diagrams current selected Functional
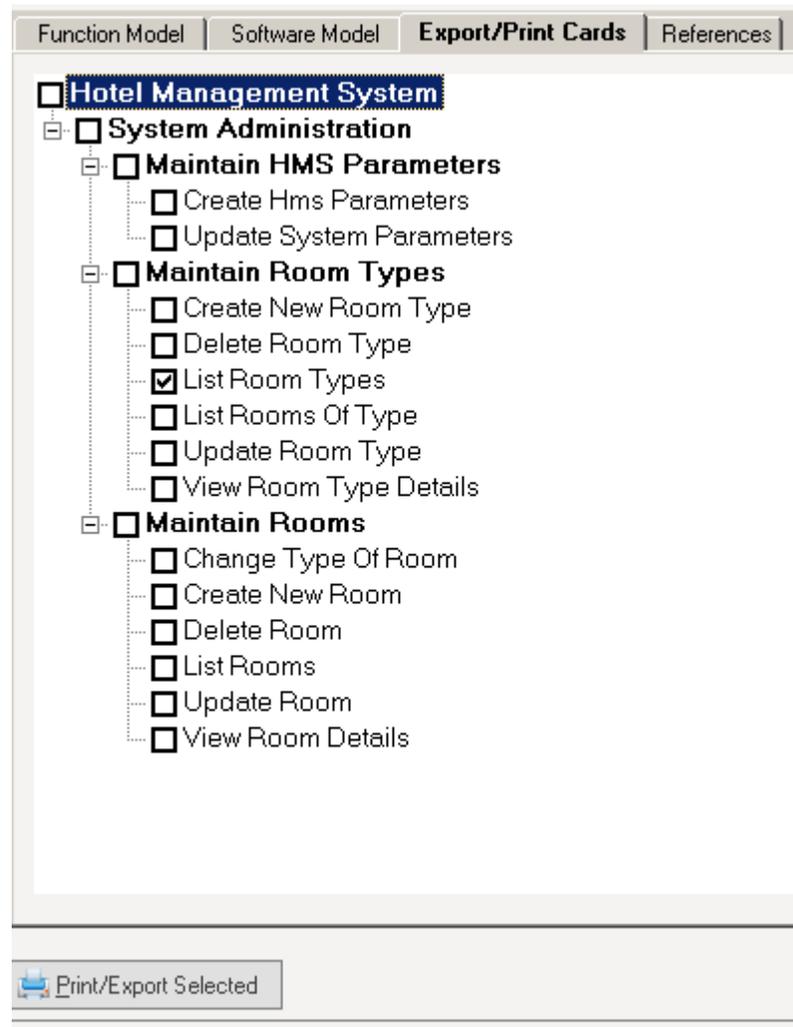
Process to a PDF File
Select the Functional process to be printed
Press the Export Button

- Print or Export a set of selected Functional Processes.
  Select the DMSD Tab

1. Select the Measurement Analysis tab

2. In the panel to the right, select the Export/Print Cards tab



3. Press the **Print Export** Button

A DMSD card shows for each selected FP the DMSD diagram and the FUR for the FP

**Visual*fsm*** Functional Process Requirements

System/Application    Hotel Management System - Quickstart

Functional Process    List Room Types

BR-SA-105: The Administrator shall be able to display, with the option of printing, a list of Room Types with information about each type.

When the Administrator requests it, HMS shall display a list of the details of a Room Types.

The Administrator shall enter a request for the list of Room Types.

HMS shall get the Room Type code, type name, no of beds, and occupancy for all types from the database.

HMS shall display the Room Type list to the Administrator.

HMS shall offer the option to print the Room Type list to the Administrator.

If the Administrator confirms the action then HMS shall send the Room Type list to the printer.

If the process could not be completed then HMS shall display an error message describing the problem to the Administrator and the process shall terminate.



| Entry | Read | Write | Exit |
|-------|------|-------|------|
| 1 | 1 | 0 | 3 |

5 CFP