How to define Service Modelling and its best practice?

What is modelling? What is a Service Model? How would this Service Model be useful to customer or user?

What is Modelling?

Modelling is a generic approach to collecting all CIs and its relationships to UCMDB or RTSM. These CIs and relationships can come from the following sources:

- **CMS and Discovery (UD):** CIs and topologies can be discovered by UD or modelled manually by the CMDB team.
- **BSM Data Collectors and stand-alone products:** BSM data collectors can create CIs and relationship within RTSM based on monitored and/or discovered data.

What is Service Modelling?

"Physical" CIs can be discovered by tools and its CIs and relationships are mapped according to the pre-defined modelling. For example, A Server has a composition relationship with CPU. This means that CPU is a components within a Server.

However, there is a challenge of representing "Logical" CIs like Business Service, Business Application, Business Function etc. Connecting these "Logical" CIs to "Physical" discovered CIs are referring as Service Modelling. In other words, Service Modelling is about building a representation of CIs that customer/user perceived with their IT supporting components.

How would these models be useful to customer?

Primary focus of the use of building a Service Model is for either impact analysis or for BSM Impact Model calculation.

Note: I believe it is also use as provider of information to Service Manager's Configuration Management Process. For example, when you select an Affected Service, the Affected CI should be the underlying CIs defined in this Service Model

Service Modelling is closely related to this set of Impact Modelling questions, for example:

1) When a server has System Availability Issue, what happen to the Business Service?
2) When a network switch is down, what happen to the Business Service or may be the server connected to it?
3) When 2 out of 5 nodes in a clustering environment is down, what happen to the Business Service? Etc etc.....

What is a Model?

A model is basically a CI in CMDB. Out of box, only some CI Types can act as a model. Those CI Types are marked as MODELLING_ENABLED qualifier on the CI Type. A model is usually a logical element such as a business service or business application.

Modelling is the task of connecting a model CI to its subordinate entities, using relationships. There are 2 different types of relationships which are available during modelling:

- Containment: Use this relationship when the subordinate entity is managed under the lifecycle of the model. These entities can be other models (for example, business applications of a

business service), or key components which are contained in the model (for example web servers or databases of a business application). You need to identify the elements which are dedicated for that model, and not shared by others

- Usage: Use this relationship when your model is using a shared resource and should be impacted by it, such as a shared DB Server or LDAP authentication service

When modelling an application, you should use Running Software CIs whenever possible, for 2 main reasons:

1) It is more accurate to map the model to the Running Software, because this is what the model is using from that node, and not anything else. For example, if you have an Oracle server running on a computer which is shared among multiple applications, you should use the DB Schema when you model a specific business application, because that is the only part it is using from that Oracle server and from that computer
2) Regarding Impact Calculation, the Running Software is impacted by the node and not vice-versa. This is needed for correct status calculation and propagation in BSM. For example, an Oracle Server is down will indicate that Software Availability issue but the node hosted the Oracle Server will not be impacted. However, if the node has System Availability issue then Oracle Server is impacted with System Availability issue too.

Once you have setup your models, you can create views based on your models. Note: The view can have its own layout that is totally different from the Model and the view is simplify to the ways how customer or user wants to see it.

Task 1: Discover your application architecture

Discovering sparingly and objectively! In this example, we are demonstrating the discovery of SM Application. The discovery activities will not be discussed here but it serves as the source of contribution to modelling ITSM Service and SM Application.

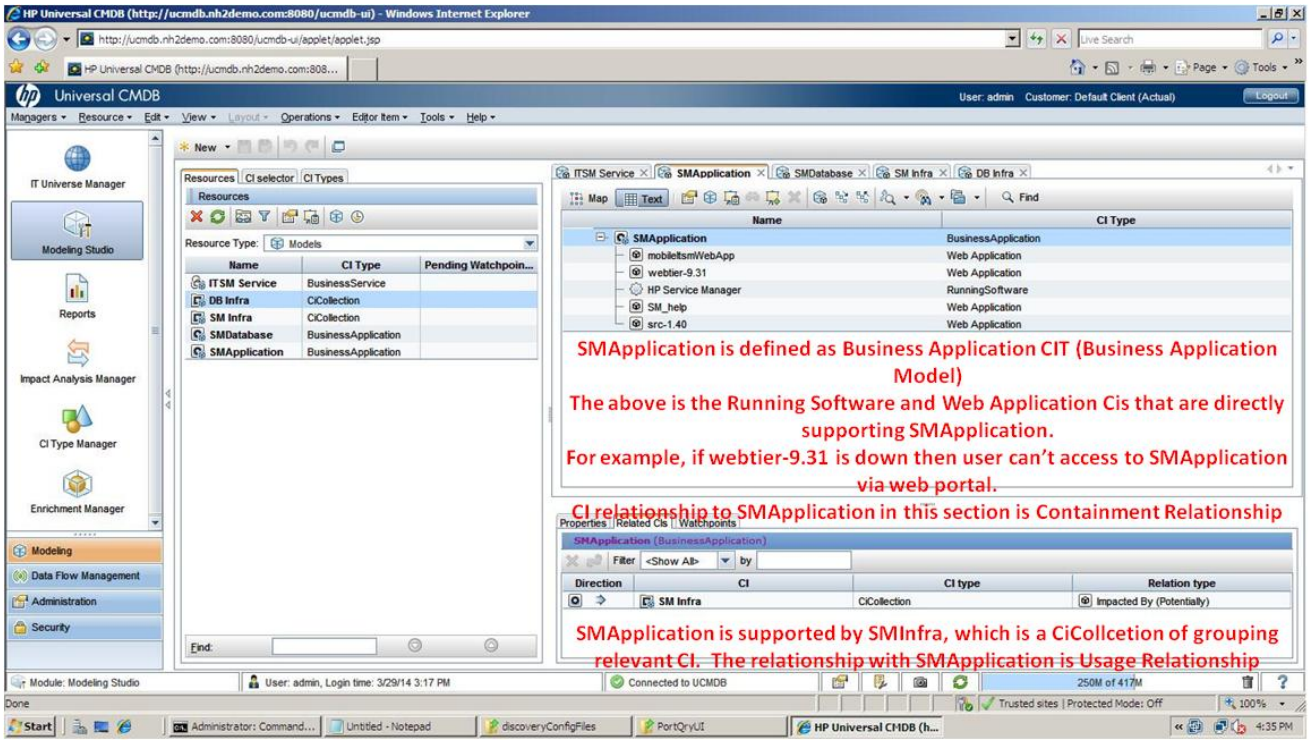
The discovery steps in general as per below:

- Network Infrastructure > Basic > Range IPs by ICMP
  - This is to discover the IP that hosting the SM Application
- Network Infrastructure > Host Connection > Host Connection by WMI and Network Infrastructure > Host Connection > Host Connection by Shell
  - This is to discover the OS so that it enables the subsequent discover job to perform more in depth discovery of information
- Host Resources > Basic Applications > Host Applications By WMI and Host Resources > Basic Applications > Host Applications by Shell
  - This is to discover the details components in a server like its processes, services, node elements, interprocess communications etc
- Database > MS-SQL > MSSQL Topology by SQL
  - This is to discover the details components of MSSQL Server software like its database schema etc
- Middlewares > Java EE Application Servers > Apache Tomcat > Apache Tomcat By Shell
  - This is to discover the details component of web servers hosting the SM Application

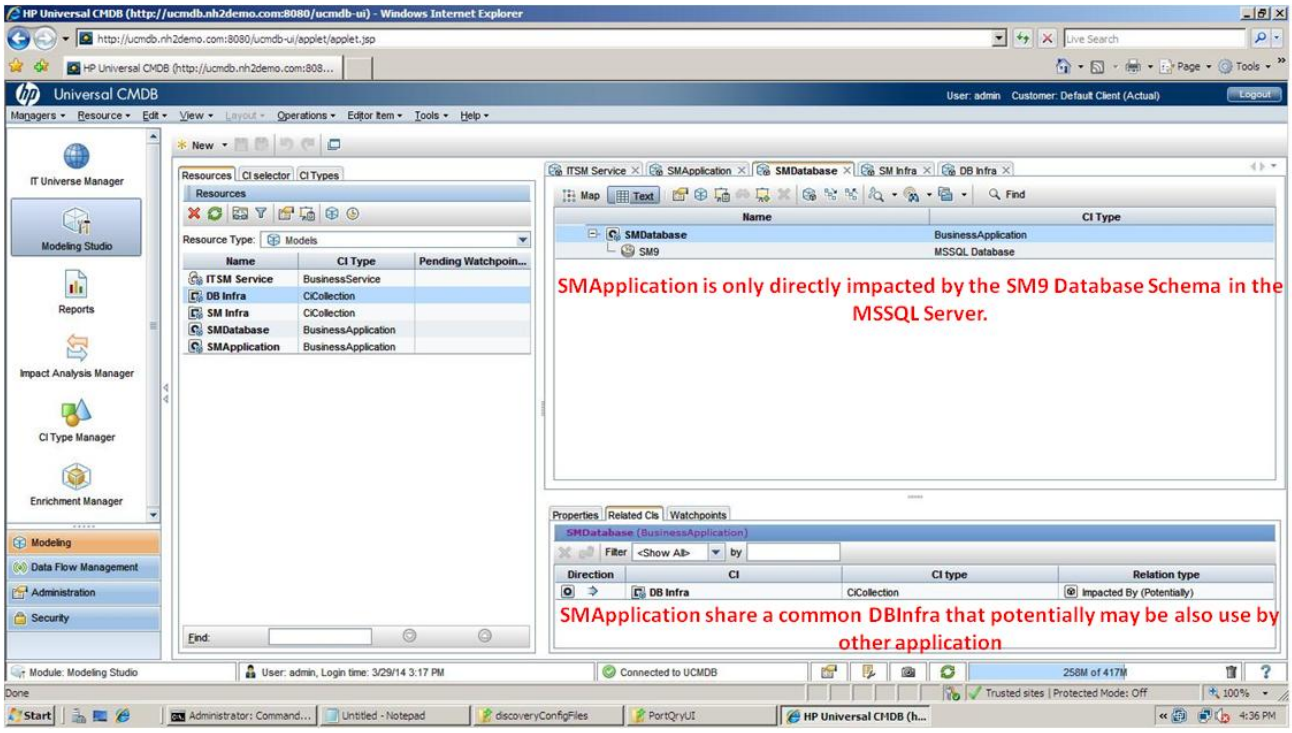
The above discovery is not exhaustive but serves as a recommendation enough to demonstrate the modelling of ITSM Service and SM Application.

Task 2: Define Models

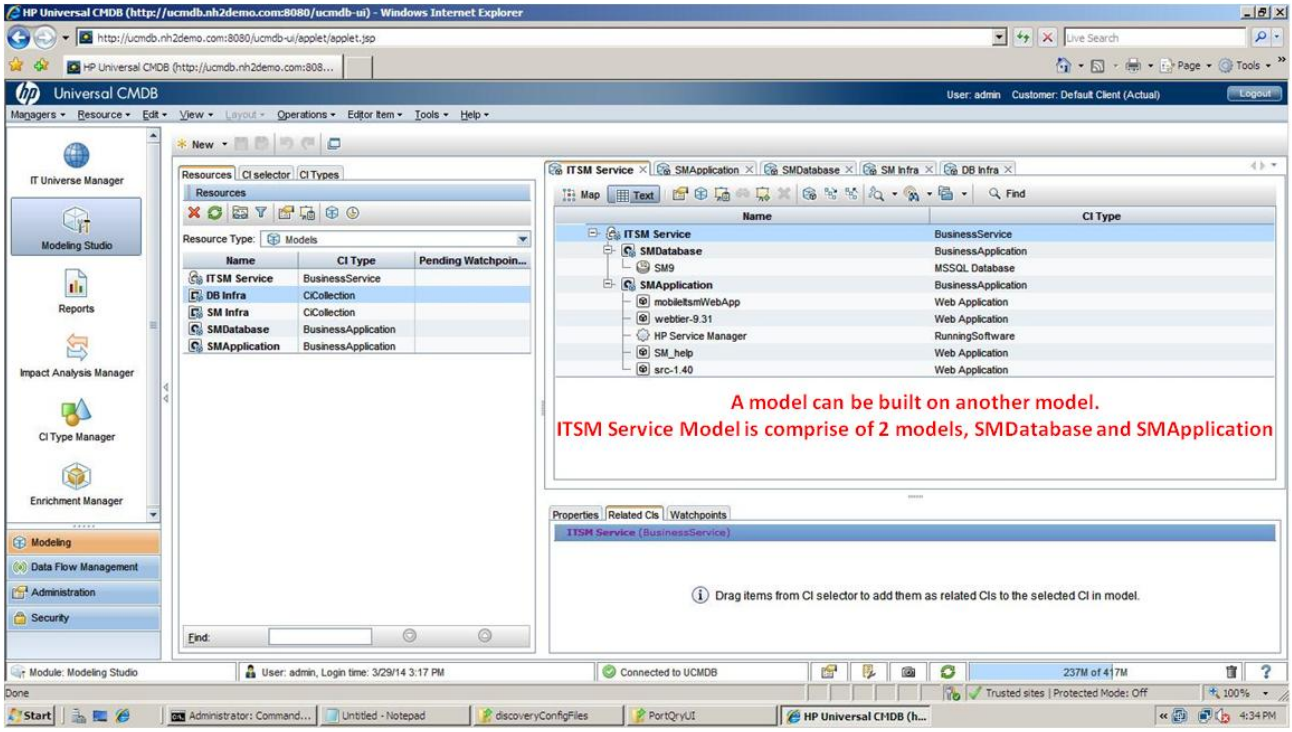
SMApplication Model:



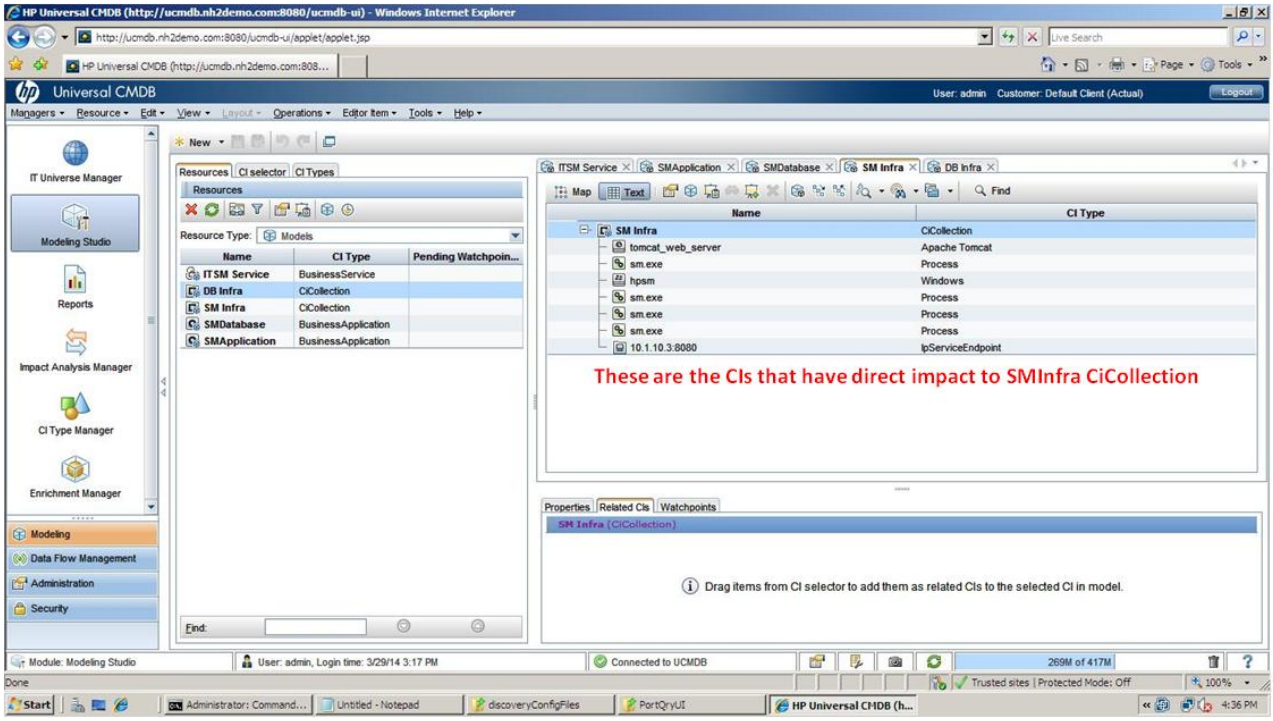
SMDatabase Model:



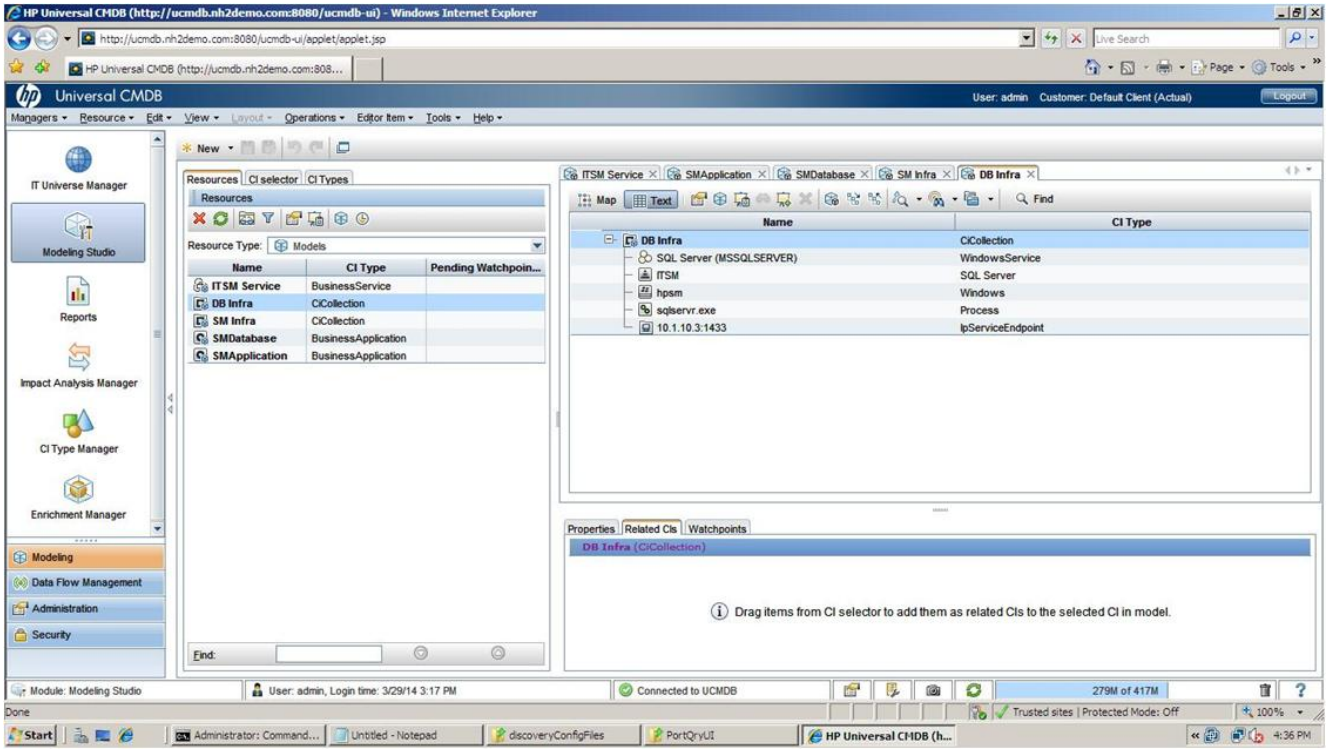
ITSM Service Model:



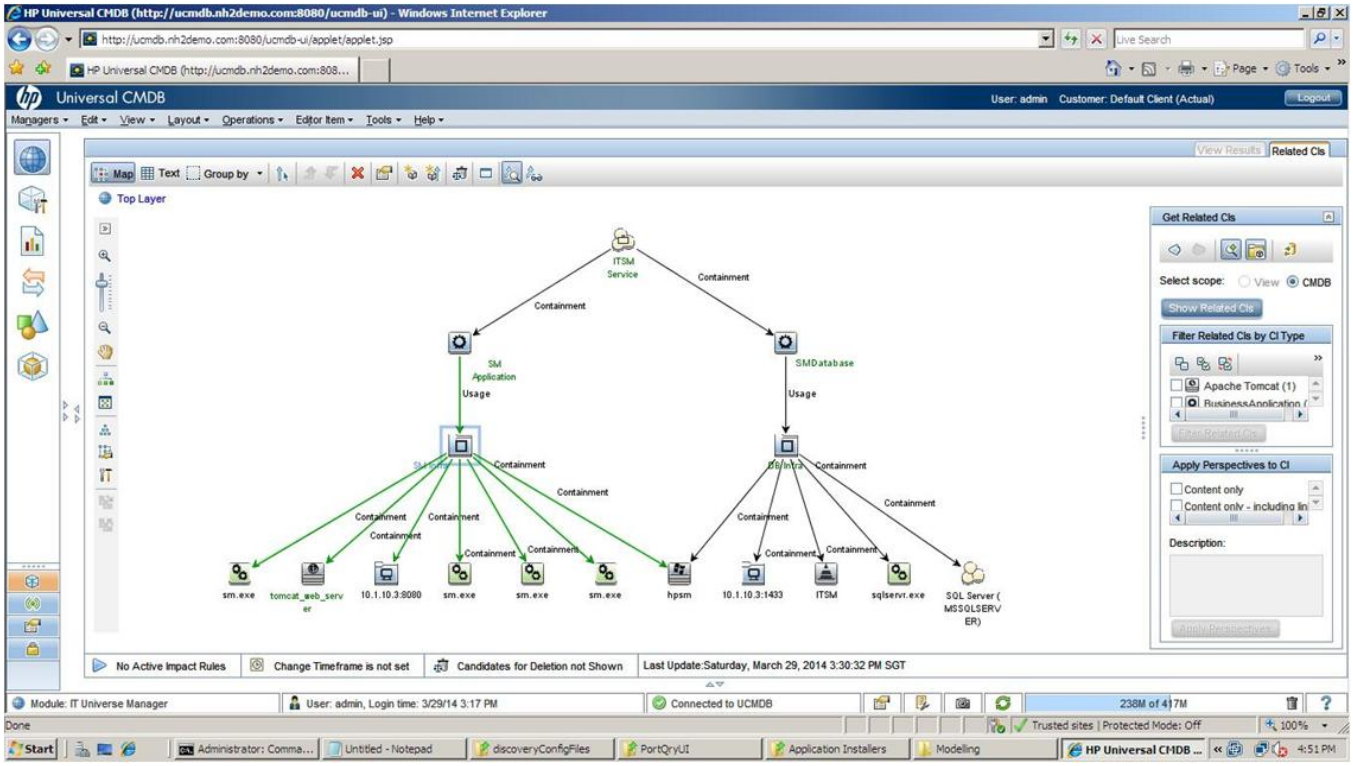
SMInfra CiCollection:



DBInfra CiCollection:



A complete ITSM Service Model:



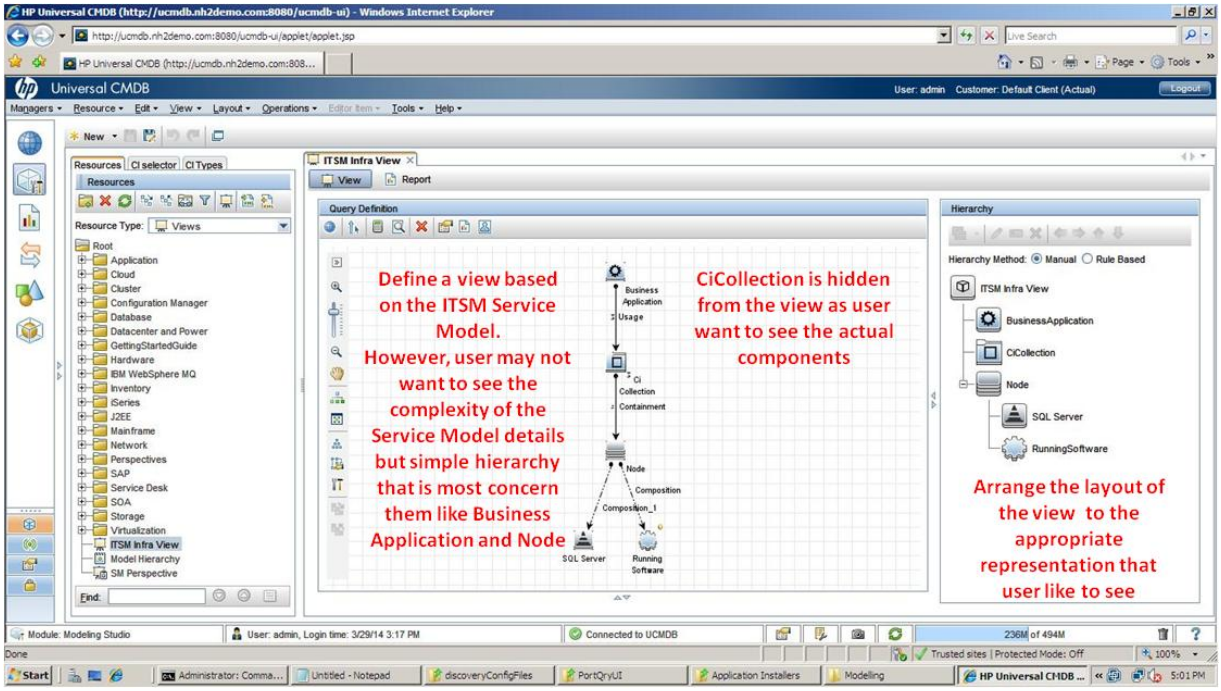
ITSM Service Model will be used for Impact Calculation. Out-Of-Box Triplets definition covered the Containment and Usage relationships for these given CI Types. Containment is defined as Directly Impacted Calculation Relationship whereas Usage is defined as Potentially Impacted Calculation Relationship.

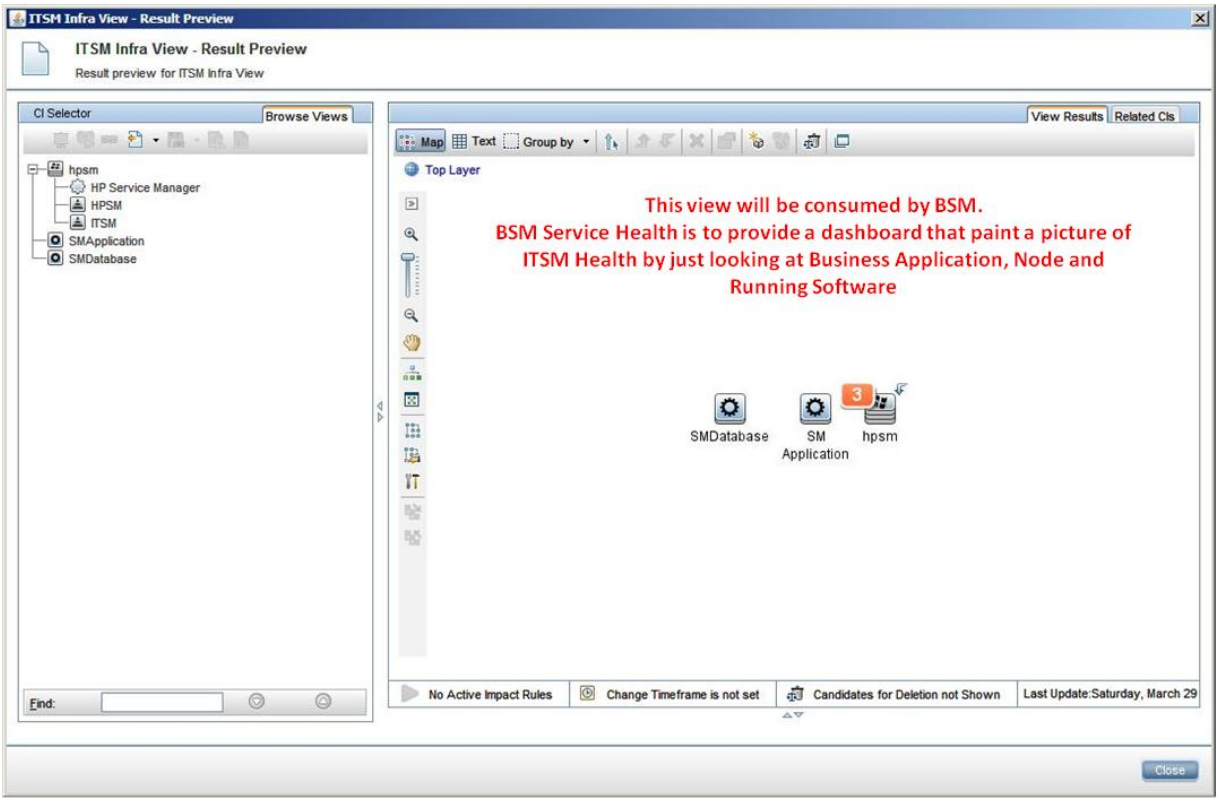
This model is important for use in BSM for the KPI Propagation rules. Based on the above Service Model, whenever 10.1.10.3:8080 port is not accessible then SM infra will be propagated with the KPI of 10.1.10.3:8080 (assuming is System Availability KPI). This propagation goes upward to SMApplication and ITSM Service.

In other words, Service Modelling is also to avoid impact calculation using the entire CMDB model. This may not be efficient for the system and many more triplets will need to be defined.

Task 3: Define View

Views are user preferences and provided for the appropriate to his/her working needs. It may not need to show the depths down to the model defined.





Service Model is used for KPI Propagation. The view will show all the relevant KPIs for the given CI. BSM will have its Business Rules to calculate the KPIs and CI Status based on the KPIs assigned after the propagation.

For complex KPI Statuses like 3 out of 5 nodes have System Availability issue, then only the CI's System Availability KPI is "red". This can be achieved using the CiCollection.

KPI Calculation: If you want to have a dedicated KPI calculated on a group of CIs, you need to use a CI Collection. For example, you can model an application across multiple data centers, or perform Percentage Rule calculations on multiple databases.