ETool Design Proposal

Author: Stupid Sun

Eid: esunfeg

1. Aim

This doc describes how requirements of ETool ’s requirements are comprehended and implemented.

From 2015/8/3 to 2015/8/28, wrote by Stupid Sun;

1. Use Cases

First of all, list of done jobs when I took over this project:

* 1. Import sites(eNodeBs):

Click “load sites” and choose a csv file;

User could only access OSS server and edited this file manually;

* 1. Load and Modify local profiles:

View list of local profiles in GUI;

Import a profile and view areas, pmchs, services and tmgis in a tree-structed GUI;

Edit above data and save to a new profile or overwrite it;

Profile are saved as a self-defined format with extension of “.pfl” ;

* 1. Bind profile and eNodeB to an association

Select a profile and several eNodeBs, click “>>” button to add them into association pool;

* 1. Report:

When association pool is not empty and profiles are “Report”, click “Report” to get a report of the eNodeBs;

Program will:

Upload report filter file, execute report command and download report result file;

Read eNodeBs from report result file and save their profiles into “eNodeBName.pfl“ to local machine;

The following jobs are to be done:

* 1. Config:

When association pool is not empty and profiles are not “Report”, click “Config” to config the eNodeBs with corresponding profiles;

Program should:

Report first and get eNodeBs and read profiles from “profileName.pfl”;  
Apply profiles on corresponding eNodeBs to get modified data, let’s call it deltas.  
Save deltas to “Lock\*.xml” and upload it and execute lock and active command;

Save deltas to “Config\*.xml” and upload it and execute config and active command;

Save deltas to “Unlock\*.xml” and upload it and execute unlock and active command;

* 1. Download sites(eNodeBs)

When using OSS is real, click “download sites” to download eNodeBs from OSS server:

Program should:

Upload eNodeB list filter file, execute report command and download report result file;

Read eNodeBs from report result file and add their ids to GUI and save them on disk;

1. Alternatives

To satisfy above use cases, I considered the previous design and thought out some alternatives:

Red ones were used at last, yellow ones were studied and tried but failed.

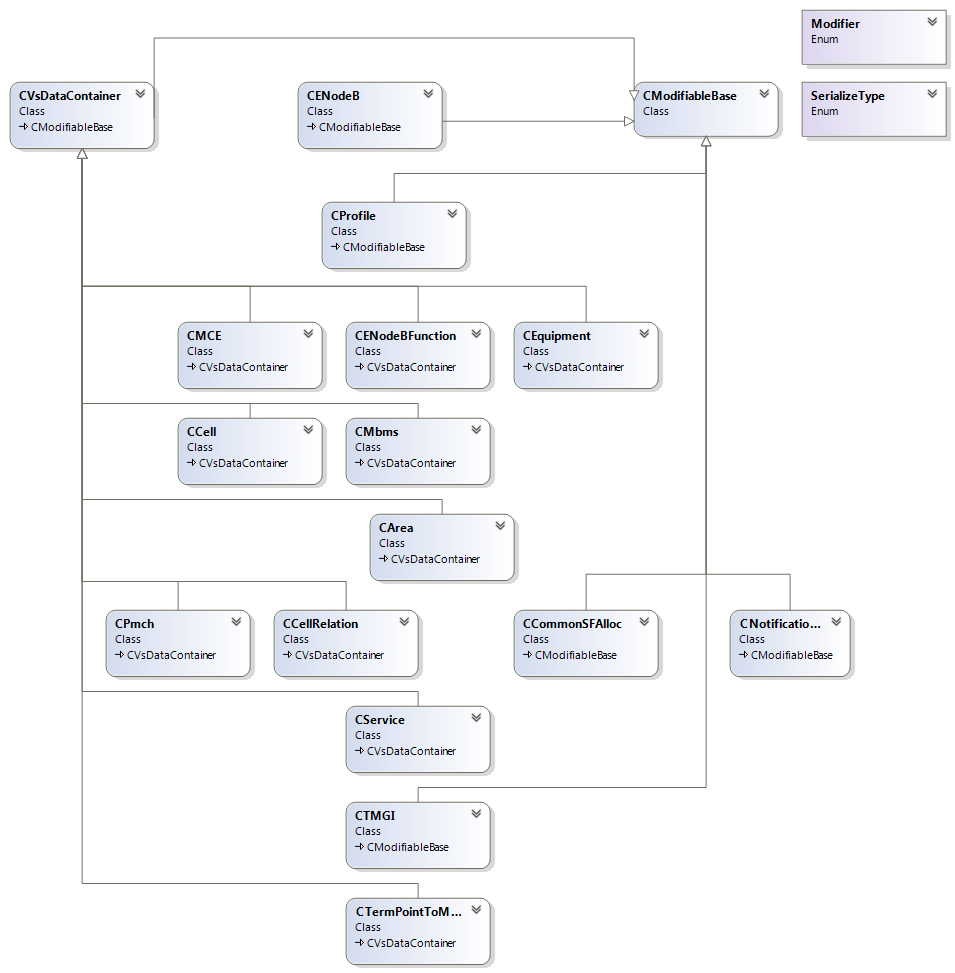
* 1. Considering the business model, C/S should be more suitable.

Reinvent the wheel in my familiar architecture maybe faster, stronger and easier.

* 1. The “pfl” format sucks. Save xml files just as xml files.

We only need a transfer between xml files to our models.

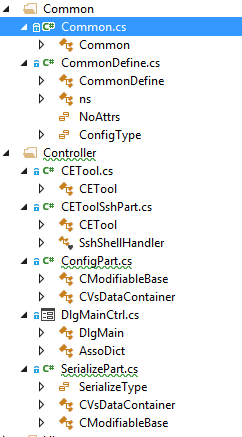
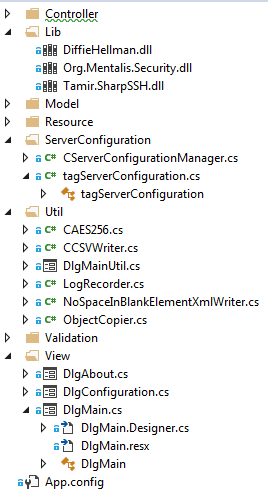
* 1. Use data models generated from xsd files to auto serialize and deSerialize XML files
  2. Use Generic classes to refactor data models to make serialize and deSerialize to be easier.



1. Class Sketch Diagram (required)

Data models’ class diagram is shown above.

Other structures can be seen from below pictures:

5. Sequence Diagram (required)

1. ETool.Sync:

ETool.Report:

ETool.ParseReportFile:

6. Interface Description (required)