

Organ Studio

Design, Implement, Deploy Now

1.19



Table of Contents

Organ Studio.....	3
Installing Organ Studio.....	3
Development Environment.....	3
1.Connecting with Cloud Service.....	3
2.Creating a new Project.....	5
3.Connecting with Cardinal Series Terminal.....	5
4.Adding Certificate for SSL Operation.....	5
5.Adding 3rd Party Library.....	5
6.Deploying Application to CloudStack (Deploy Now).....	6
7.Debugging and Simulation.....	7
8.GUI Designer.....	8
9.Organ Limitation.....	9
File System Builder.....	10
1.Creating File System.....	10
2.Scanning File System.....	12
3.Modifying File Contents.....	12
4.Commit Changes.....	14
5.Compare File System.....	14
6.Changing Property.....	14
7.Image Operation.....	15
8.Extension.....	16

Organ Studio

Organ Studio is an Integrated Development Environment (IDE) for Orb-Weaver, It includes Compiler, Script Editor, Project Management, Project Deployment and Simulator intended to simplify card application development.

Minimum System Requirement :

Intel/AMD processor (x86 based/compatible)

128 MB free memory

Windows, Linux (with Mono)

Microsoft dotNET Framework 3.5 and Visual C++ Redistributable 2008

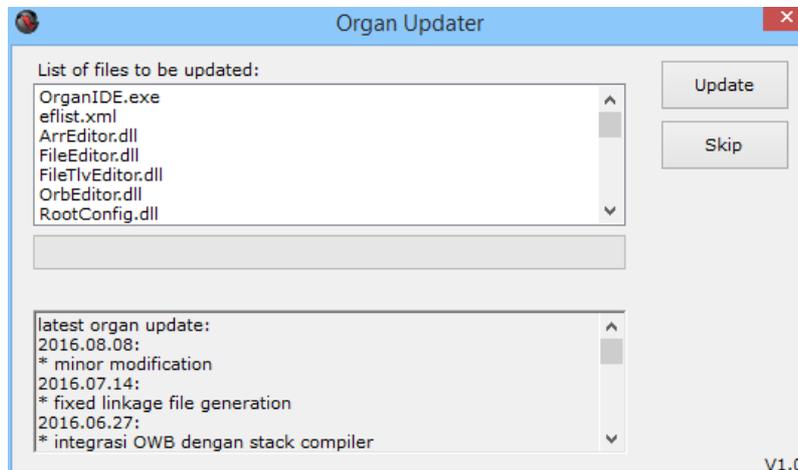
Internet Access

1280 x 720 screen resolution or larger (for better point of view)

Etheron Interface driver (optional)

Installing Organ Studio

Download Organ Studio zip file from www.orbleaf.com, extract to directory that you want to install to, a single file called OrganUpdater.exe will automatically download the rest of file, click Update to start the installation process.



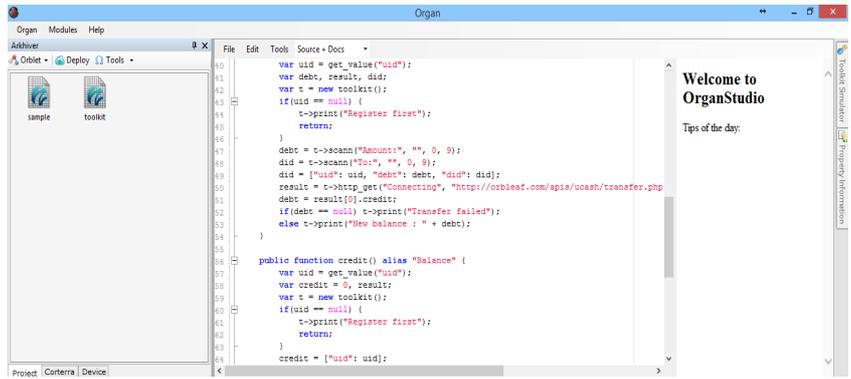
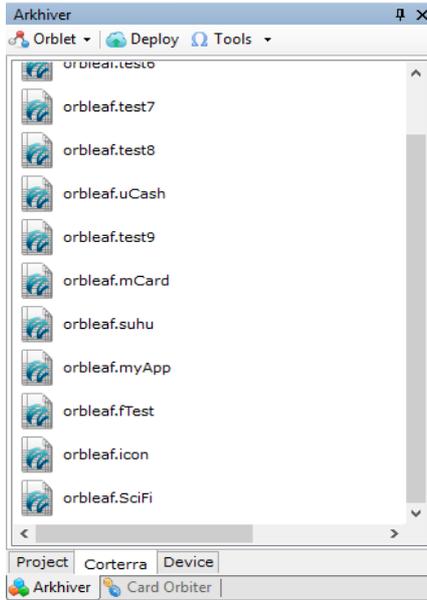
Organ automatically check for new update each time application run, it will show user confirmation dialog whether or not to download new updates.

Development Environment

One of many features of Organ Studio is for development of Orb-Weaver application for Orb-Weaver based device, it support Orb-Weaver programming language (OS-Script) as default programming language for Stack compiler integrated within Organ Studio (please refer to [Orb-Weaver Programming Manual](#) for specification of the language and framework)

1. Connecting with Cloud Service

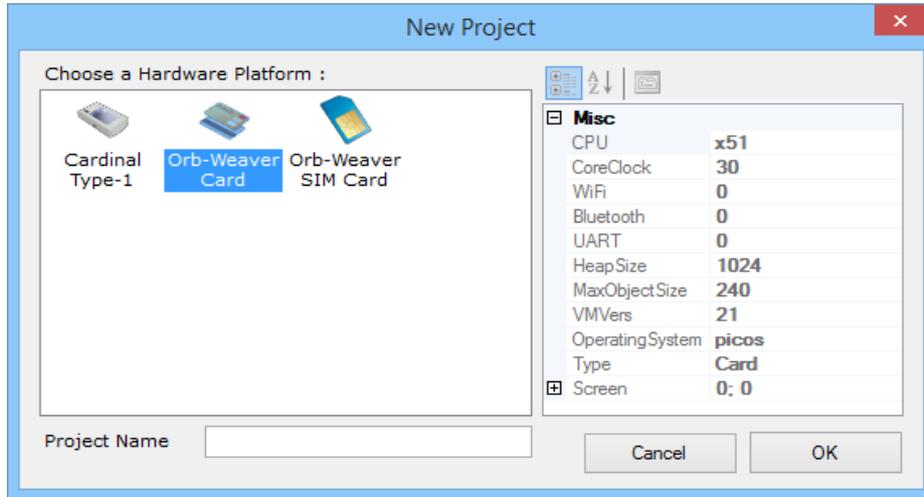
From Organ Studio mainview, select Modules > Project Explorer to view Project Explorer window, click sign-in to start sign in to your OrbLeaf Developer account, after sign-in you list your application by selecting Deployed Apps tab, or changing your default password to new password by clicking tools



Organ require user to connect to it's cloud service in-order to use all of it;s features such as online repository, deploy now and cloud apps management, without any connection to cloud user could still use it;s basic feature such as creating, building, debugging and simulating an application.

2. Creating a new Project

At Organ menu item at the top of the window select new, a new window will be popped up and you can select type of target device which application will be deployed.



3. Connecting with Cardinal Series Terminal

To start accessing Cardinal Terminal from Organ Studio, first you should install Etheron driver provided by EtheronInterface_vXX.zip which can be downloaded from www.orbleaf.com, depending on your Windows OS, an access to diagnostic mode might required.

After you've installed the driver for Cardinal Terminal, you can start connecting Cardinal Terminal to your PC through USB port, if the installation of driver from previous step was successful, a device setup dialog would appear, just wait till it done.

Accessing Cardinal Terminal on Organ Studio could be done through Project Explorer window, by selecting Device tab, a new dialog box to select which terminal to connect would appear, you should insert the card to the terminal by now, and start selecting which terminal to access.

4. Adding Certificate for SSL Operation

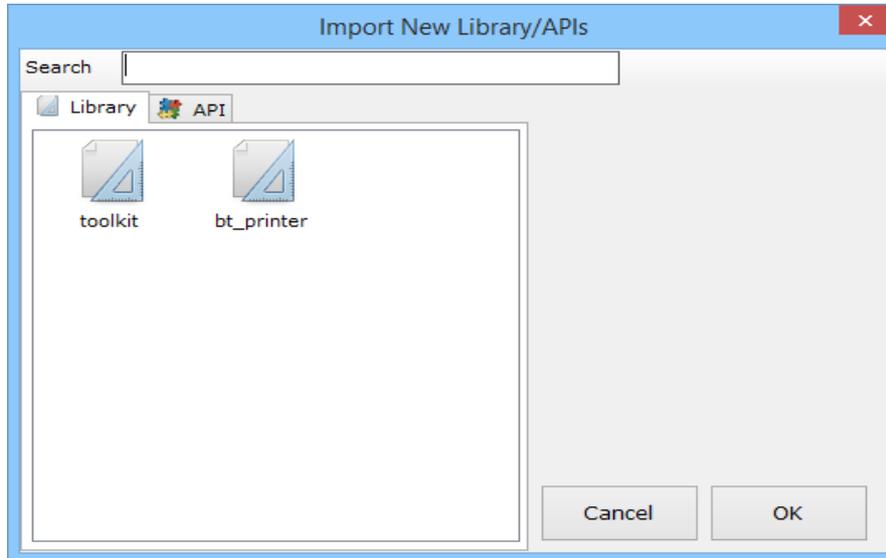
Adding SSL certificate for SSL operation on Orb-Weaver is quite easy, start by clicking Orblet > Add > Certificate and select which certificate file you want to add to your project. Certificate must encoded in PEM format otherwise it won't be recognized by Orb-Weaver runtime.



Terminal would later automatically access your certificate during HTTPS request, SSL handshaking on Cardinal Terminal is quite slow compared with normal HTTP request, we suggest you should only use SSL only when the data to be send/receive is really sensitive, depending on the application.

5. Adding 3rd Party Library

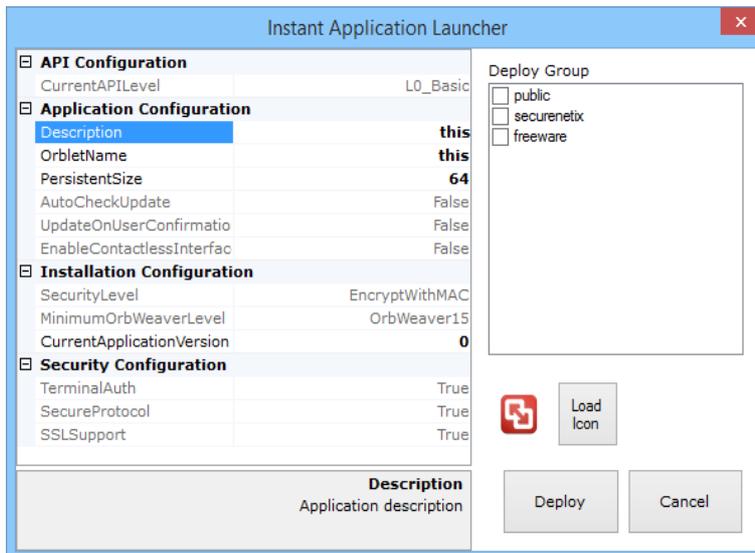
Organ was designed to provide exchange of APIs through the use online repository, this feature enable collaboration between different APIs provider such as cloud service or device driver with application developer. Adding a 3rd party library for project can be done from Project Explorer by selecting Files > Add > Library menu item



Organ must be connected to internet to use this feature, user could also search OrbLeaf collection of library by typing a specific keyword on the search box, this would filtered out libraries compared with searching them manually.

6. Deploying Application to CloudStack (Deploy Now)

CloudStack is our Card Application Distribution Service for any Orb-Weaver based device, to start deploying application to CloudStack service you should start by building your application first, selecting Deploy on Project Explorer would automatically build all your project file and displaying Instant Application Launcher (Organ must be connected to cloud service in order to deploy application)

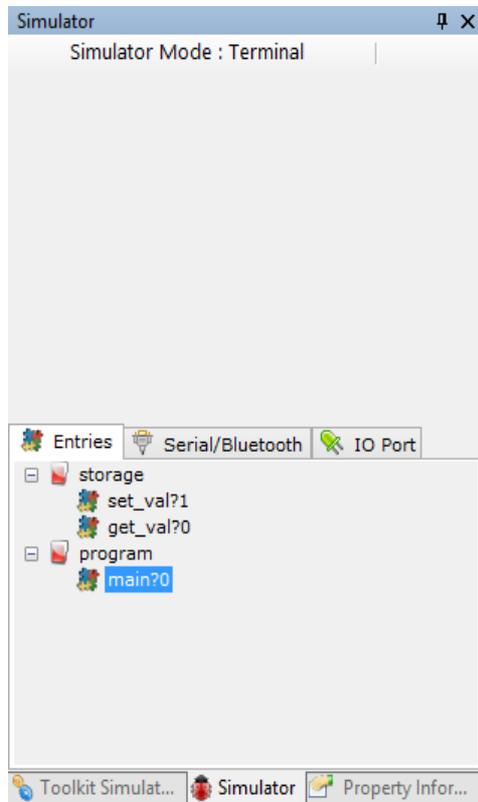


Here's list of action which you can do during Deployment process

- Changing application icon by clicking Load Icon button, application icon must be in PNG format (also support transparency), it's resolution will automatically re-sized by Organ, icon only works for active device such as Cardinal series terminal, for Orb-Weaver card and Orb-Weaver SIM card icon are automatically replaced by system icon.
- Changing allocated persistent storage size for your application, necessary when you use persistent memory APIs, persistent memory determined amount of pre-allocated storage for user application.
- Changing your application name, application name must be unique within your list of application.
- Change your deployment group, deploying it to public group would allowing people to download your application.

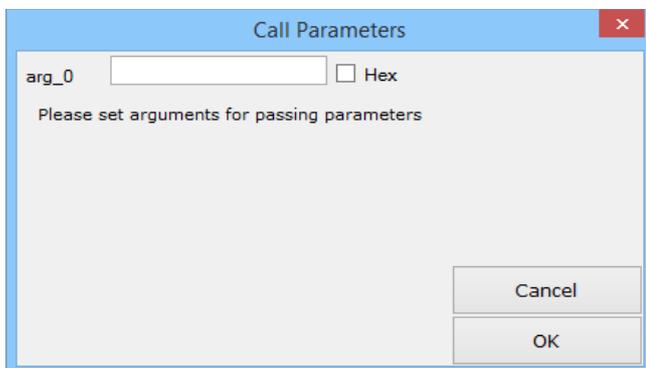
7. Debugging and Simulation

Debugging application can be done through Project Explorer, select Debug from Project Explorer menu item. All registered entries can be seen on Simulator > Entries tab, in-case of card application simulator automatically displayed all available card menus.



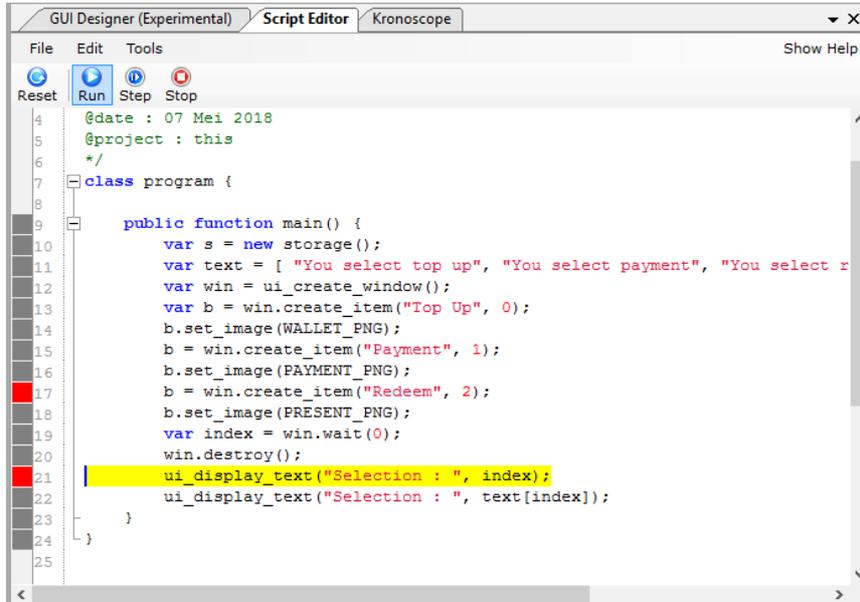
Within Simulator window, user could trigger specific function registered within application.

Double clicking entries will displayed Call Parameters window which asked for user input argument for callee function, amount of arguments will depend on executed entries.



Double clicking line on the program automatically set a breakpoint, double clicking marked breakpoint line remove the breakpoint itself, breakpoint can be used to stop program execution at specific line. After a breakpoint triggered user could use reset, run, step or stop program execution

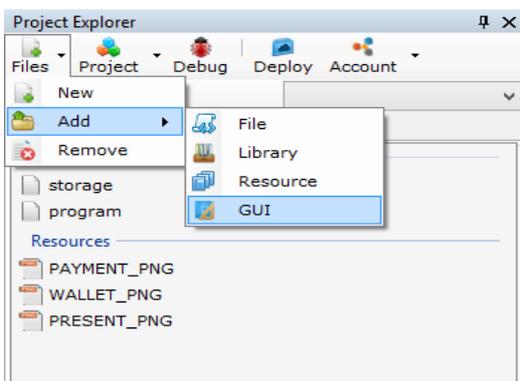
- reset, restart program execution
- run, execute program execution until next breakpoint triggered on end of script
- step, execute program execution to next line of statement
- stop, terminate program execution



Content of variables can be traced from Observer window > Call Stack tab, some variable content such as object or array will be displayed in parsed tree structure.

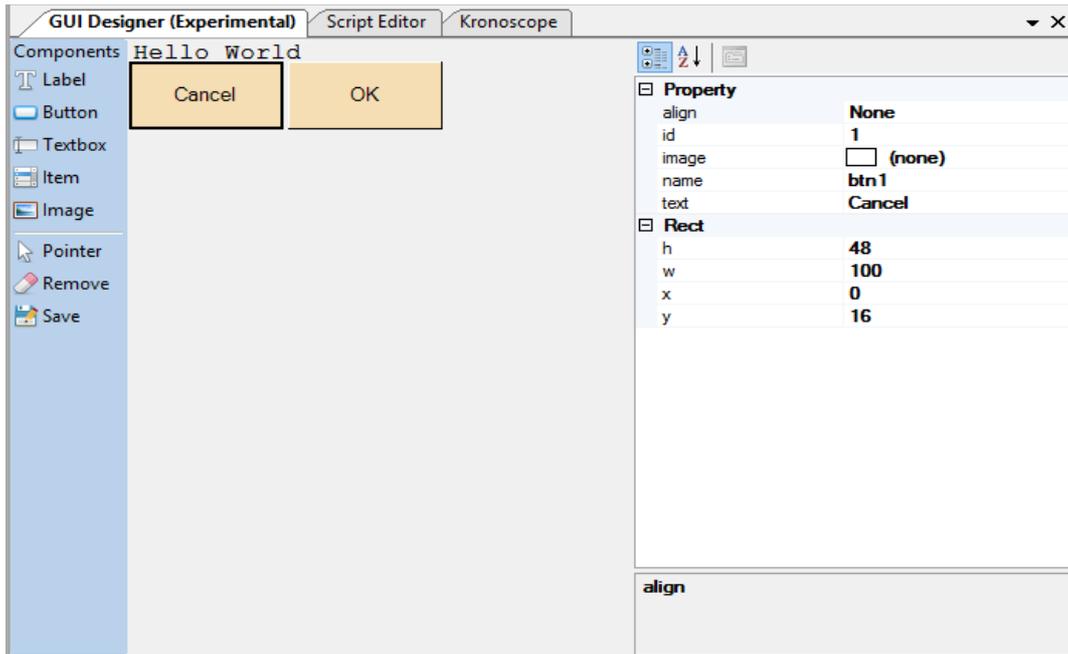
8. GUI Designer

GUI designer can be used to build custom GUI application instead of pre-defined GUI, GUI builder only applicable for non-card application, in order to trigger GUI designer start by adding GUI file into Project Explorer, Files > Add > GUI menu item.

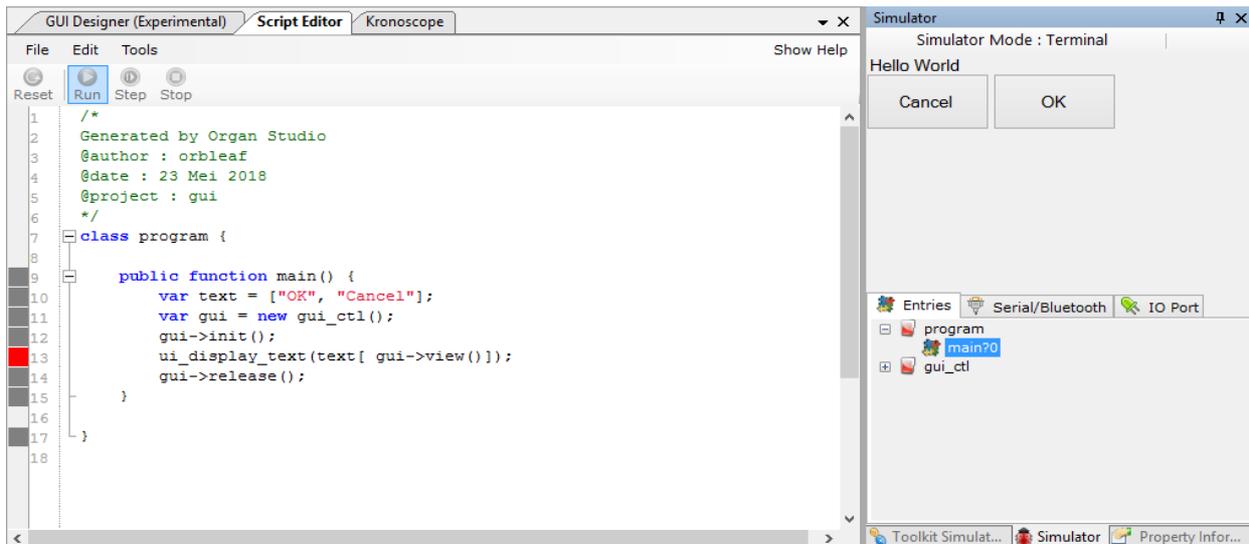


GUI designer will automatically popped up when double clicking GUI file, GUI file can be invoked through MVC mechanism on any function.

Component can be added by clicking toolbar on the left side of GUI designer, it's property later can be changed using property grid on the right.



When a component align property set to none, please make sure that it's area still inside screen size (see rect property) otherwise it wouldn't displayed properly on screen, alignment none remove auto-layout feature of GUI engine, therefore component position are determined solely by GUI designer.



When a GUI file created, it will create two files, one is GUI resource file (which can be edited by GUI designer) and the other is GUI control source file, which contain all necessary method that can be used related to designed GUI, this file generated and modified automatically by GUI designer, user should not modify this file.

Invoking GUI file would be as simple as calling it's generated method, start by initializing GUI by invoking it's control instance, use init function to initialize GUI control with it's respective GUI resource file, use view to display the GUI to user, view method will automatically halt script execution until a user event is generated and id of triggered component will be returned to caller. Release function can be used to release GUI resources from memory, otherwise the GUI resources will be only released when script execution finished or terminated.

9. Organ Limitation

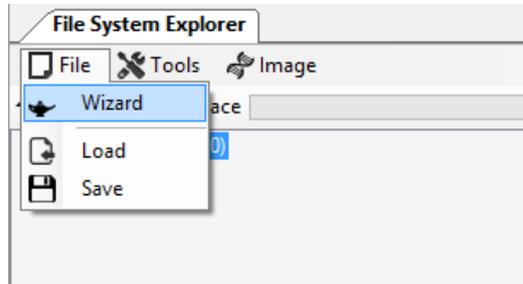
Not all APIs of Orb-Weaver platform can be simulated by Organ Simulator, some APIs such as `ti_load` (invoking a package from terminal application/framework) and `ci_load` (invoking a package from other card application) cannot be simulated by Organ because it's require implementation of external application package, which currently not supported under current Organ version.

File System Builder

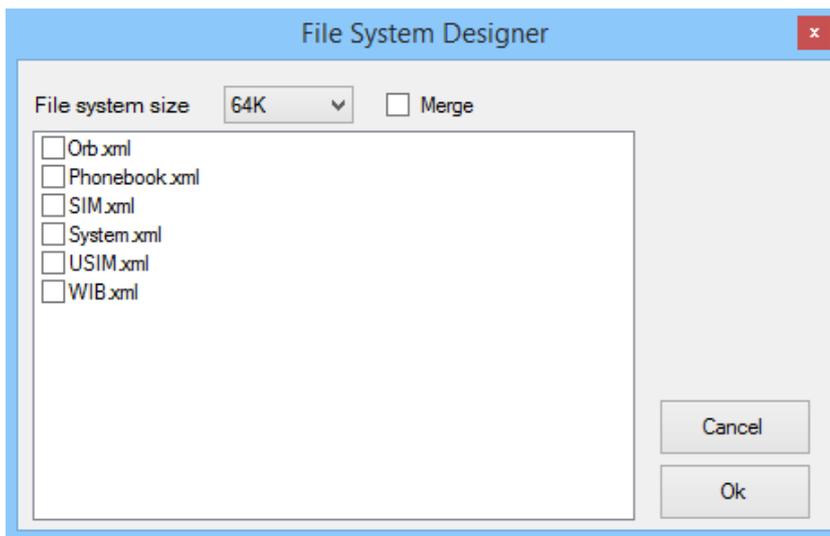
File system builder is one feature of Organ Studio, it can be used for creating a file system for smart card based device according to ISO7816-4 or 3GPPP standard, in-order to use this feature a PCSC smart card reader must be connected to PC. File system builder is only available for the enterprise version of Organ Studio, for the standard development version this feature is not available for public use.

1. Creating File System

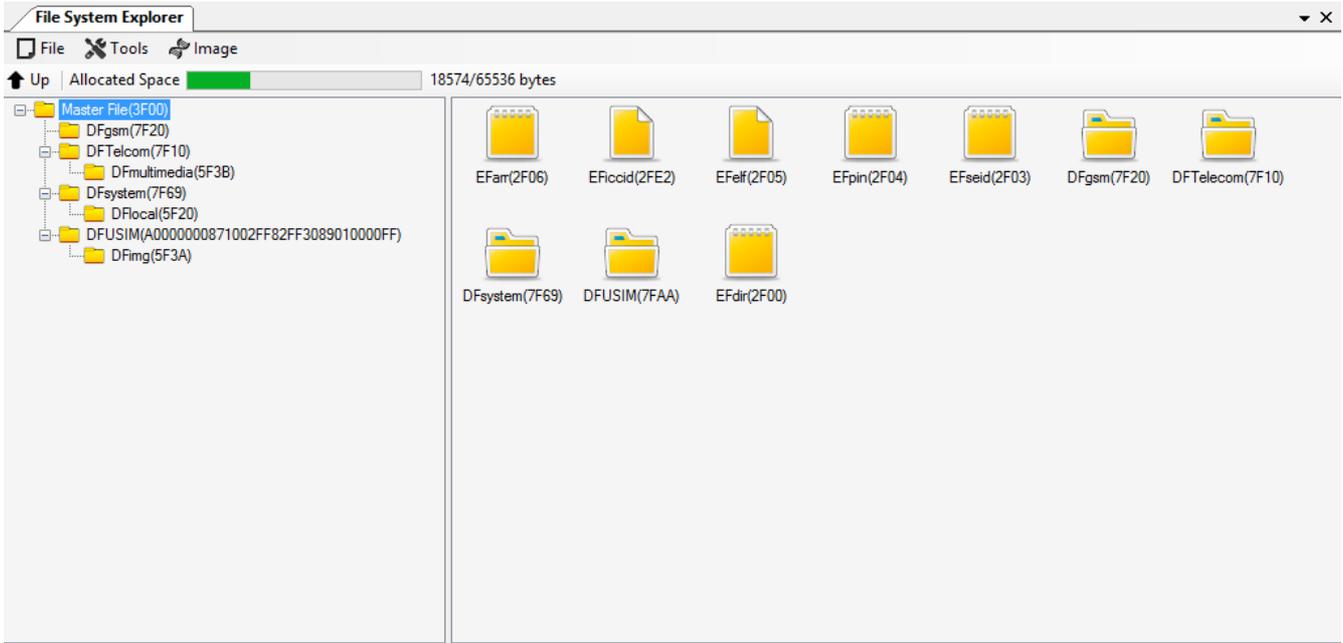
From File System Explorer click File->Wizard to pop up wizard dialog for file system creation



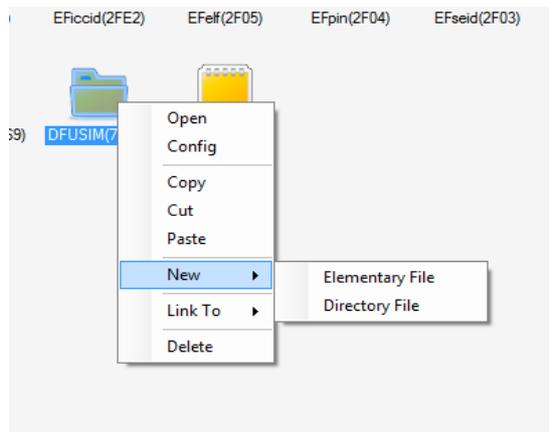
once the wizard window popped up, select which default profile to be used as file system template, do not select any template for empty file system



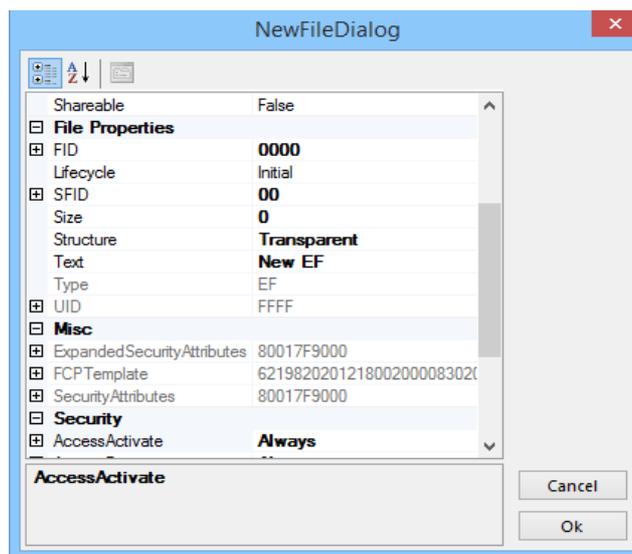
for USIM/SIM application, system.xml template should always be included, which will be used for system directory, configuration of system directory please refer to iCOS user manual.



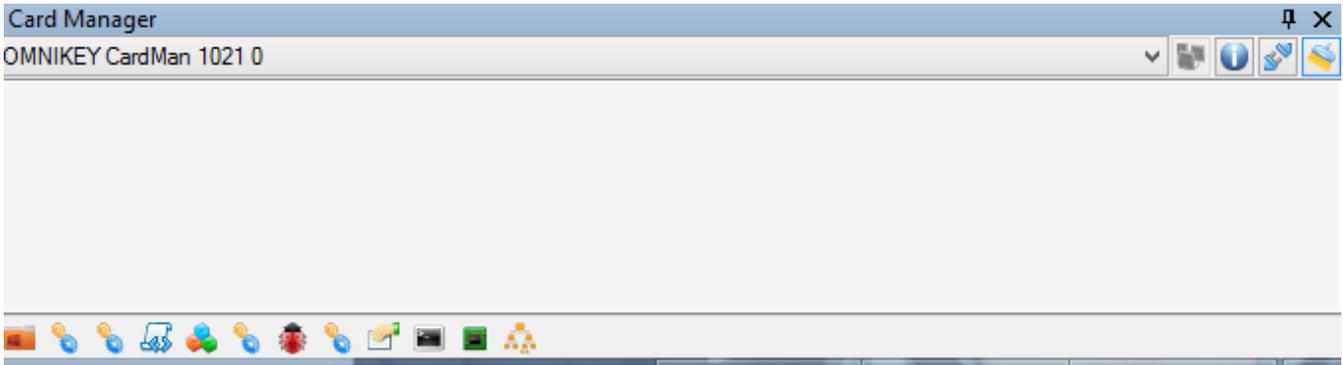
You can start editing file information such as record file, size, structure, moving file can easily be done by dragging file to another directory. Use right click to create new file or directory.



A window will popped up containing configuration of newly created file, configure the file which you want to create.



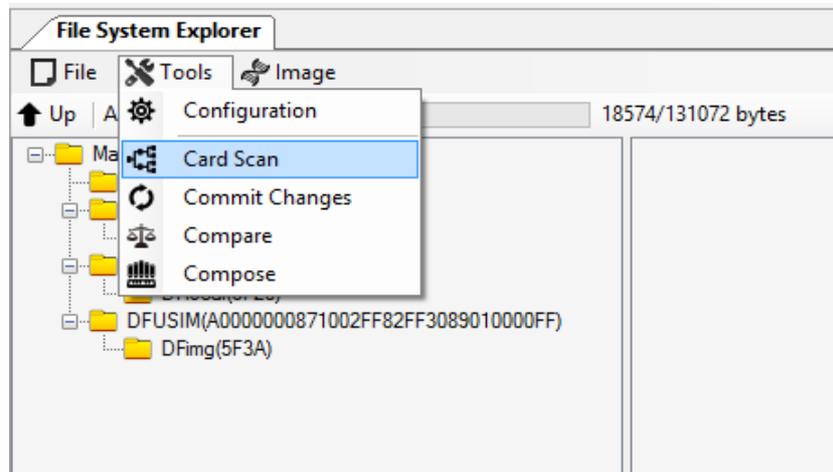
Insert the card to card reader, select the card reader from card manager



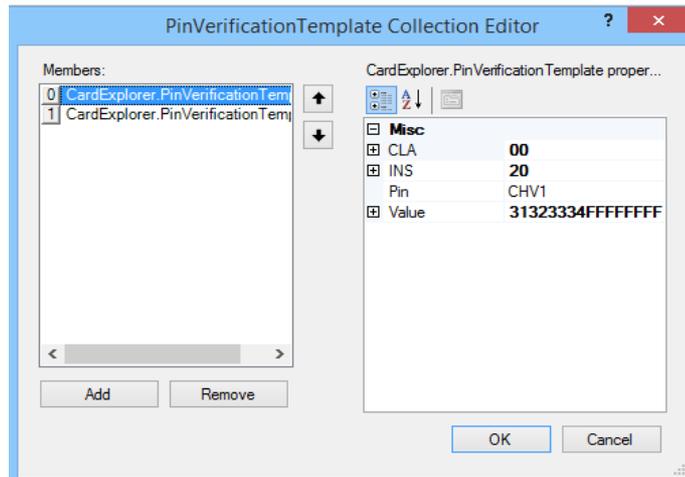
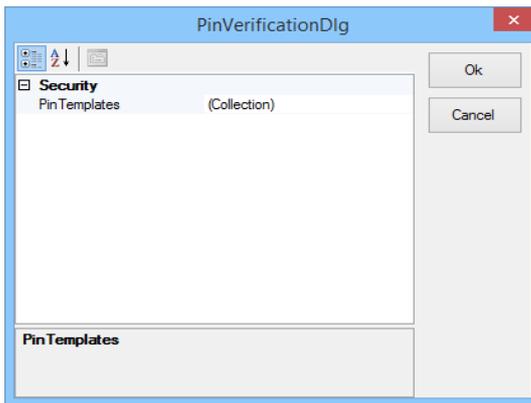
once card inserted, you can start building by clicking Tools->Compose, Organ Designer composer would automatically generate sequence of APDU for file system creation.

2. Scanning File System

Insert the smart card device onto smart card reader selected from Card Manager window, on Card Explorer click Tools->Card Scan, Organ automatically scan inserted card for any compliant 3GPPP files.



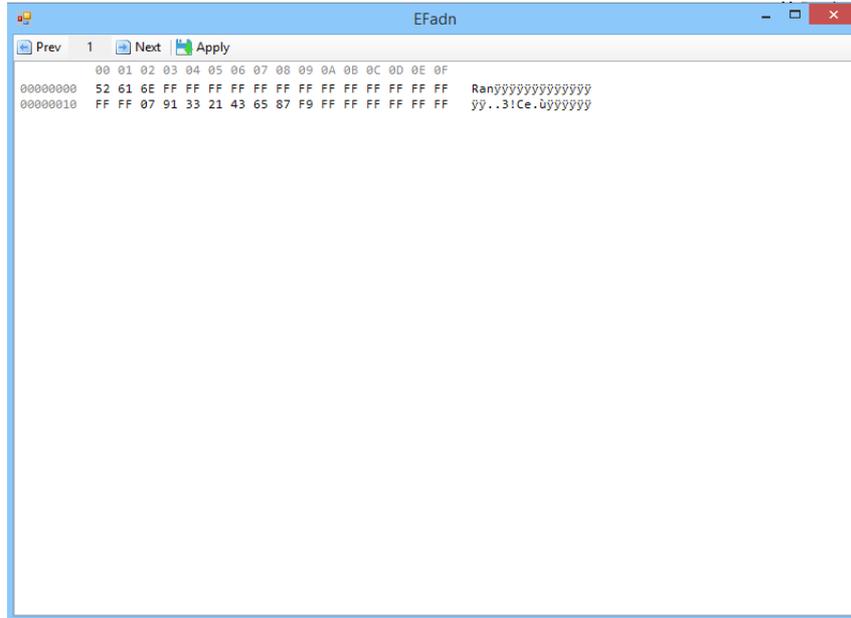
Disable any pin verification by deleting pin default value, or change it to current card pin value, Organ also automatically verify pin for any privileged access to some file contents.



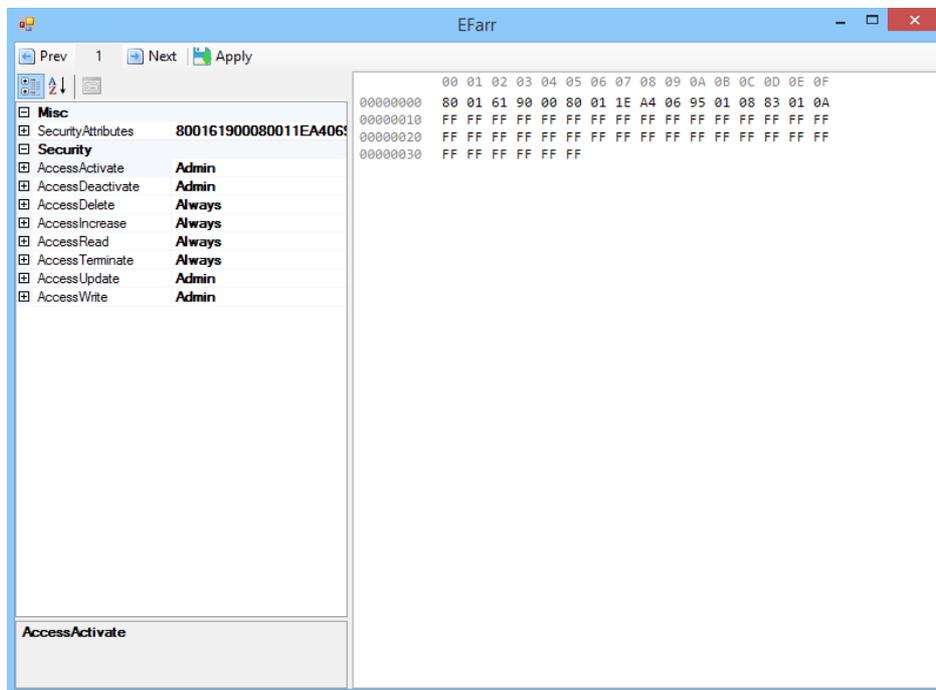
After the Organ completed the scan job, you can browse the structure of scanned card, or compose it to Orb-Weaver card.

3. Modifying File Contents

File content modification are done through extension, each might have different kind of format which will be handled by extension (similar with windows explorer can be used to open specific application for different kind of file format), extension are automatically registered during software updates.



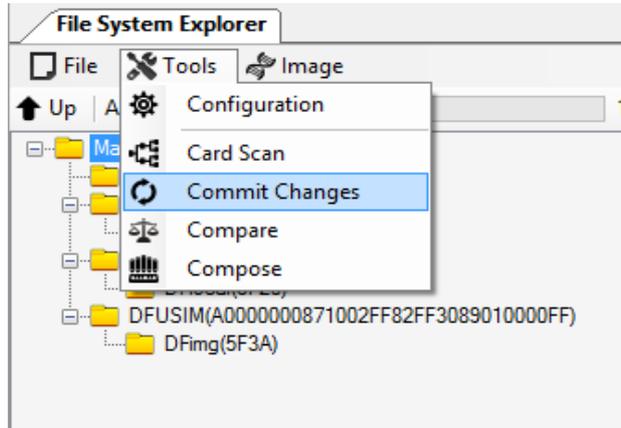
Screenshot for default file extension, editing record file, after editing content on each record, click Apply button to apply the update to current file.



Extension for opening EF_access_rules_references (EF_arr), editing content of EF_arr can be done easily through configuration property on left side panel, different kind of EF might have different kind of extension registered

4. Commit Changes

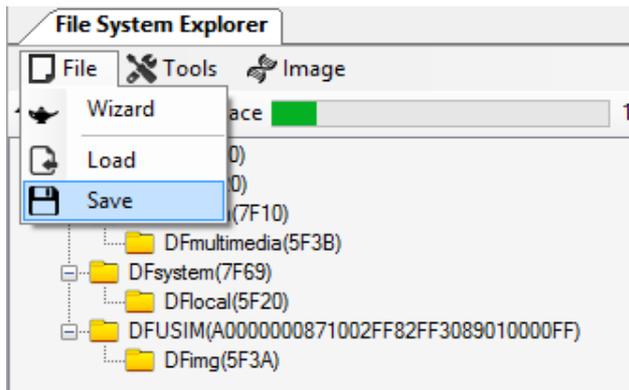
When user need to commit changes to card, use Tools->Commit Changes to flush all modified data to card (write access need to be verified beforehand).



Organ would automatically update all modified EF(s) on smart card.

5. Compare File System

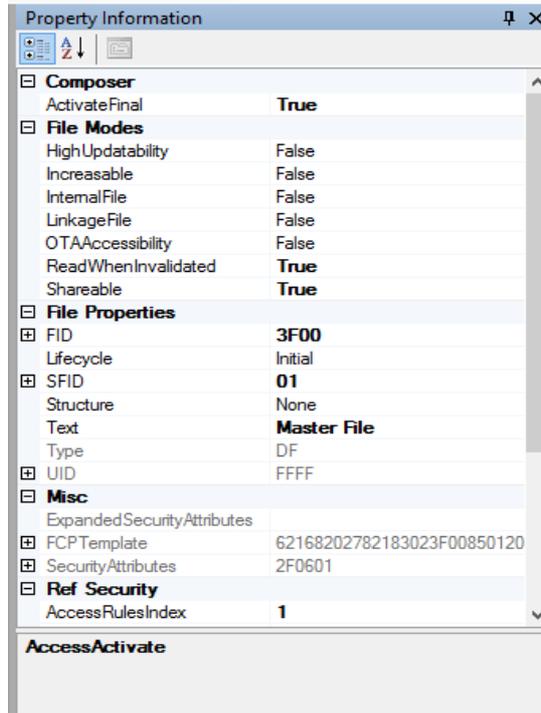
Compare file system useful when user need to compare two file system to check for different, the first file system to compare need to be load first before being compare, click Tools->Compare to select which profile to compare, in order to create profile file user must save the profile in XML format, click File->Save to save current file system to XML format.



Once file system has been save as XML format later it could be used as compare profile, user can compare scan result with compare profile, the result will be shown on Console window.

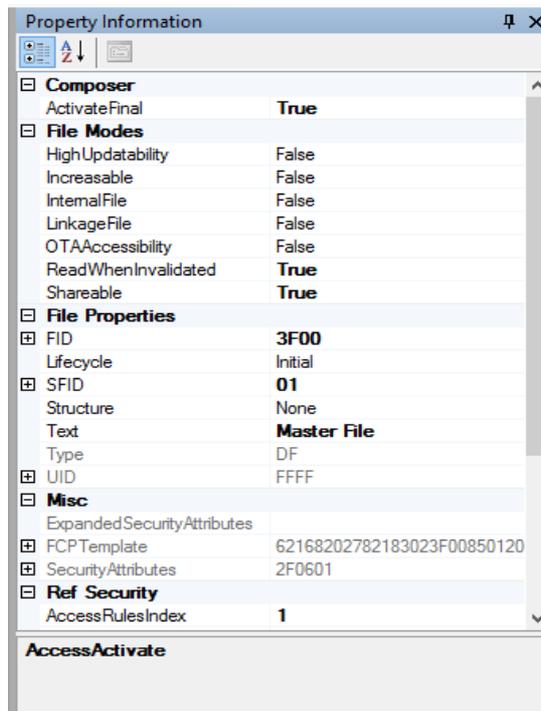
6. Changing Property

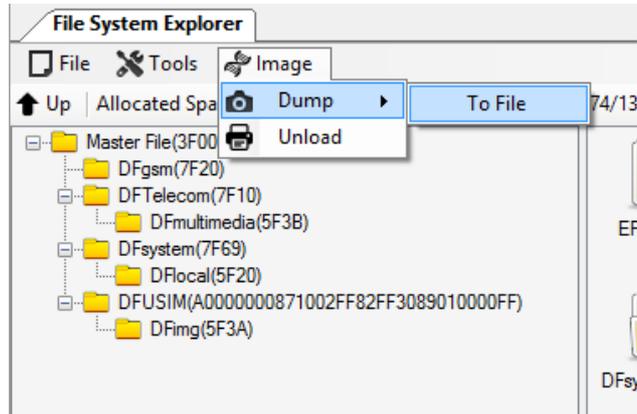
Changing configuration of file can be done through property window, the configuration updated in realtime, the property shown on property window belong to last item selected by user, it can be an EF, DF, ADF, MF or even APDU log shown Card Manager.



7. Image Operation

Image operation can be used when Orb-Weaver Card inserted on card reader, it automatically dump all file system image for duplication, image is useful for production purpose, as it will significantly increase speed of card personalization, current operation support dumping card image to file.





8. Extension

Extension are use to view, edit and update specific EF on file system explorer, it's available in form of plugin which will be invoked when specified EF with pre-assigned extension configuration are clicked. The purpose of extension is to view data in a more readable format compared with it's default viewer, through use of extension editing EF such as EF_arr is much more easier compared with editing in it's binary form.