# User's Manual for Yuhana 0.01.1 bate

Last update By Merlin Lain August 15, 2003

## License of this document



## Target Readers

The supposed reader of this documentation are the developers of Yuhana project; software developers who are going to use the releases of Yuhana to work with their own; or people who is interested in it.

## Purpose of this document

The main purpose of this documentation is to explain the structure of Yuhana binary release and source release, and explain how to use this release to work as an stand alone Application sever or software library can be linked to other programs, as well as running those demonstrations included.

## About project Yuhana

This software is to provide an open source, simple but featured, pure .Net implements, portable .Net application server software. It provides an object persistence mechanism; web service host engine; enable to deploy your exist .Net component as a web service component.

## Table of Contents



## System Requirement

Microsoft .Net Framework 1.1 , and a computer can run it. (We still have some problems with running the hole release under other runtime.)

## Structure of release

Table of Contents



The structure of binary release 0.01.1 b

```
<release root>
    |--<lib>   (directory for libs )
    |--<exttools>   (directory for tools )
        |--<storageviewer>
    |--<StorageRoot>   (root directory for all persistent data )
    |--<testprogram>   (directory for programs test the Demos work with this release )
        |-- DemoCalculatorTestClient.exe   (Test program for Democalculator)
        |-- DemoChatter.exe   (Test program for Democalculator)
    |--<webroot>   (root directory for all web resource host with this release)
        |--< YUHANAService > (Directory for Web services of Yuhana system function)
            |--< MonYuhanaManageService > (Host the System manages service)
            |--< UserManageService >(Host the Users manage service)
            |-- < ObjectStoreService > (Host the object store service)
        |--< DemoCalculator > (Host the Calculator web service)
        |--<Demo> (Host all Demo web services)
            |--<PetShop> (aspx files of the Yuhana.net petshop)
            |--<WebServicePetShop> (web service of the petshop)
            |--<EMailService> (Nuked)
```

The structure of source release 0.01.1 b

```
< release root >
    |--< core > core modules
        |--< ObjectComponentRuntime >
        |--< ObjectView > object management module
        |--< Storage > storage management module
        |--< WebServiceEngine >
    |--< exttool >
        |--< shutdowntool >
        |--< storageviewer>
    |--< launcher >application server launcher program
    |--< server > Application server module
    |--< utility > utilities
    |--< webservice > webservice interface for Yuhana system functions
```

The source of demos is also included.

**Use as a Stand alone Application Server**

Use as a Stand alone Application Server means you can distribute you webservice with Yuhana without aspx platform support, for example you can host you webservice one win9x with MS .net FX 1.1 or MONO runtime(on schedule).

To distribute you webservice, following this steps:

1. Place you file for webservice under the <webroot> directory . The wsdl(Web Service Definition Language ) file of your web service must be place with them.

   For example you file are place at <webroot>\mywebservice, the wsdl file must be at <webroot>\mywebservice\myservice.wsdl .

2. The binary lib of you webservice must be palce at <yourserviceroot>\bin. For example at: <webroot>\mywebservice\bin\myservice.dll .

3. Modify the wsdl file , make the web service bind address to the proper host web address. As the example , make it like:

```
<service name="mywebservice">
    <port name="mywebserviceSoap" binding="s0:MyServiceSoap">
      <soap:address location="http://127.0.0.1/mywebservice/MyService.asmx" />
    </port>
</service>
```

4. Place the .asmx file at the <yourserviceroot>. Notice that the format of .asmx file we needed is not same as MS's format. For convenient , it is not a xml file, just include the full type name of the web-service class. We will change it in the future.

   See the <webroot>\DemoCaculator\DemoCaculator.asmx for an example.

5. Launch the app server, use you webservice tool to test it.

**Note:**

You may need to config the App server, open the .config file of the launcher, modify the <YuhanaWSServer> section. You can modify the serve IP and port, like this:

<YuhanaWSServer ServeIP="0.0.0.0" ServePort="8000" WebRoot=".\webroot" />

Cause our implement does not depend on ASP.Net , your web service do not need to inherit from System.Web.WebService, but the WebServiceAtrribute and WebMehtodAtrribute is needed currently. The proper wsdl file is essential for current implement , in the future release the wsdl file will be optional.

Please reference to the included DemoCalculator for a real host example. ( <webroot>\DemoCaculator).

**Use as Lib**

Use as a lib means dynamic link or static link you program with Yuhana libs.

See the "API documentation of Yuhana" for api specification .

Check Out the source files of those demos included.

See the "Developer's Guide" for detail information. (This document is still under

construction)

### Use the source

Use the source means modify the source code of Yuhana and make it to fit your requirement .

To build the source, we do not ship any build script or project file yet.

See the "Developer's Guide" for detail information. (This document is still under construction) If you meet some problems, do not grudge you question.

You can also get the those source via anonymous CVS like this:

cvs -d:pserver:anonymous@cvs.sourceforge.net:/cvsroot/yuhana login

### Tools

There are 2 tools included in current release:

Storageviewer: A simple tool to view the content of storage.

Launch the Storageviewer.exe, click the button to load a .csg cluster file under the <storageroot> dir, click the list component on the lest to see the content of the cluster file.

Shutdowntool: a tool to shutdown the service in the legal wait.

Open the config file of this tool , edit the value of "ShutdownServiceURL" ,make it point to the proper URL of Yuhana system webservice MonYuhanaManageService, here is a example:

<add key="ShutdownServiceURL" value="http://127.0.0.1:8000/YUHANAService/MonYuhanaManageService/MonYuhanaManageService.asmx"/>

Then execute it to shutdown the app server.

NOTE:

To shut down the server you must use this tool or use the MonYuhanaManageService to make the server do some finalizing work, such as finish transaction , write cache to storage. Close the app server rudely may lose data.

### Running Demo

There are 3 demonstrations ship with this release.

**Yuhana.Net PetShop**: Yuhana.Net PetShop is a modified version of .Net PetShop 2.x which can run with Yuhana system.

**Demo Calculator**: This Calculator is a web service calculator.

**Demo Chatter**: It is a stand alone P2P Instant-Message demo program based on web service.

Running the Yuhana.Net PetShop

1. you need a IIS server to host the representation tier written in asp.net. Those apsx file are located at <webroot>\Demo\PetShop, you may a virtual directory for it in you IIS. For example the vd is http://localhost/PetShop. make sure you can see the welcome page of the pet shop.
2. Modify the web.config file of the PetShop at <webroot>\Demo\PetShop\web.config. make the "webserviceurl" item point to the proper web service url of the pet shop service host on the Yuhana server. For an example:

```
<appSettings>
    <add key="ConnectionString" value="server=MERMAID;database=petshop;integrated security=SSPI;" />
    <add key="webserviceurl" value="http://127.0.0.1:8000/Demo/WebServicePetShop/WebServicePetShop.asmx" />
  </appSettings>
```

Note that, in this example, the Yuhana server running on the same machine but use port 8000.(Check out the config file of the launcher)

3. start up the Yuhana server with the launcher. Open you browser and type the Petshop Url, in the case is:
   http://localhost/PetShop

Running the Demo Calculator

This demo is located at <releaseroo>\ testprogram\ DemoCaculatorTestClient.exe

Check out the config file of the program, make sure the "CaculatorServiceURL" is correct:

```
<add key="CaculatorServiceURL" value="http://127.0.0.1:8000/DemoCaculator/DemoCaculator.asmx"/>
```

Run the server, then run the calculator.

Running the Demo Calculator

- The program itself include a web service host function which provide service for other chatter to send their messages, as well as a web service caller which can use the other chatter services.

1.Check out the config file of the program DemoChatter.exe.config :

```
<appSettings>
    <add key="ListenURL" value="http://127.0.0.1:8001/DemoChatter.asmx" />
    <add key="ListenPort" value="8001" />
    <add key="ClientURL" value="http://127.0.0.1:8002/DemoChatter.asmx" />
</appSettings>
```

ListenURL: The chat web service of this program.(It is not used.)
ListenPort: The listen port of the chat web service.
ClientURL: The service URL of another chatter you chat with.

2. Run the program, check out those configure values in the textboxes. Click the "Reset chatter Service" button to enable receive messages. Write your words in the text-box at the left-down corner, and click the "send" button to send them.

3. Modify the ClientURL value in textbox to send messages to another chatter if you want.

## How to get help

If you meet some problem, please check out this document and the FQA carefully first. If the problem still can not be done, Please post your problem at our help forum ( http://sourceforge.net/forum/forum.php?forum_id=280186 ) , you are welcome.
No FQA currently, we need your questions to make it **J**

## More Information

This is project is host at sourceforge as an open source project, the code name for it on the sf.net is yuhana. Home page: http://yuhana.sourceforge.net/ . Visit our page on sf.net for lasted update, discussion, help, and your contribution.