

## INFOLAB RXM PORTAL - OPERATIONS

Before you read this Document, you should have seen the **OVERVIEW**.

### Purpose of this Manual

The purpose of this User Manual is to –

- a) Explain the Basics of Operating the INFOLAB<sub>rxm</sub> Portal
- b) Explain the Steps to Maintain the INFOLAB<sub>rxm</sub> dependency with the INFOLAB Enterprise Server

### Operating the INFOLAB<sub>rxm</sub> Portal

#### a. Starting the Portal Service



For the Portal to operate, it should be started either at Services, or by selecting 'Start the Portal as an Application' (from Start, All Programs). When you use the latter option, INFOLAB<sub>rxm</sub> will open a Window as shown below, and which you may minimize, but which is useful for viewing status messages.



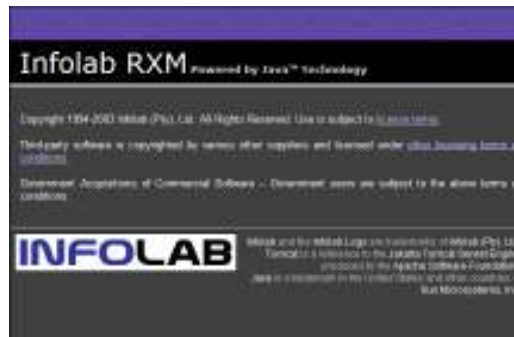
Hint: Although this step starts the INFOLAB<sub>rxm</sub> Portal, it cannot do much unless the INFOLAB Enterprise Service is also running, and Connection(s) can be established.

#### b. Checking the Status of the Portal Service



To check the status of the Portal after it has been started, select 'INFOLAB<sub>rxm</sub> Status'. You may do this initially (and optionally) to

verify that the system is working, or at any time when there is a need to. You should see something like what is shown below, if the Portal is functioning properly –



The page shown above indicates that the Portal is functioning, and you can also check 'License Terms' on it.

Another topic that relates to the status of the service is that of the 'Log File'. By default, your Portal is installed with a configuration that automatically maintains a Log of all Portal Events on the Portal Processing Side (not to be confused with another Log on the INFOLAB Enterprise Server side). This Log is most useful for viewing Portal Events.

On the Portal Installation Path, e.g. C:\Infolab\WebPortal\webapps\INFOLABrxm\WEB-INF you will find a file called 'web.xml', and we show a picture of it below.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN" "h
<web-app>
  <servlet>
    <servlet-name>InfolabRXM</servlet-name>
    <servlet-class>Azure.InfolabRXM.Broker</servlet-class>
    <!-- uncomment this to enable logging -->
    <init-param>
      <param-name>log_file</param-name>
      <param-value>rxm.log</param-value>
    </init-param>
    <!--
  -->
    <init-param>
      <param-name>verbose</param-name>
      <param-value>>true</param-value>
    </init-param>
  </servlet>
</web-app>
```

The 1<sup>st</sup> <init-param> tag specifies the use of 'rxm.log'. If you do not wish to use a Log File, you may remove this tag.

Hint: '.xml' Files may be safely edited with Notepad or Wordpad if you do not have an XML Editor.

Important: If you continue to use this Log File 'rxm.log' (found on the same path as specified above), it should be a regular operational procedure to delete this File when it gets too big, in which case INFOLAB<sub>rxm</sub> will create a new log file when the service is started again.

c. Stopping the Portal Service

INFOLAB<sub>rxm</sub> Portal is stopped either at SERVICES, or by closing the Application Window, depending on how it was started. When the service is stopped, any open Connections to INFOLAB Enterprise Server is automatically and tidily wrapped up.



d. Uninstall the Portal Service



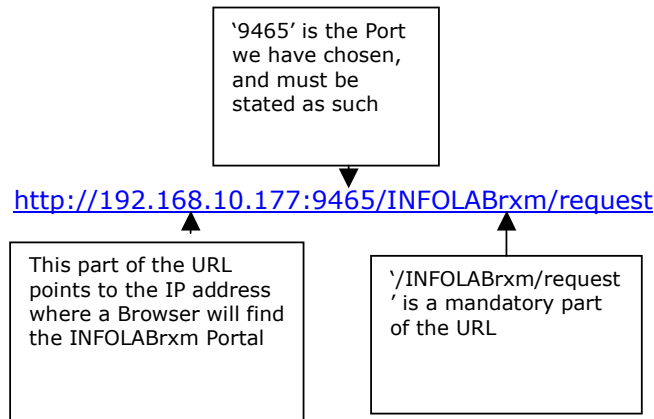
You may select 'Uninstall the INFOLAB Web Portal' to uninstall this Service. If you are particularly averse to editing '.xml' files to change the Portal Operations Parameters, then you may use this step to Uninstall, then Install again and specify the parameters in the new version of your choice.

e. Adapting Operations Parameters

The steps listed below, i.e. to change Operations Parameters for the Portal, may be achieved by editing the specified Files, or by using step (4) above, i.e. by Uninstalling the Portal, and then installing it again with the preferred parameters. INFOLAB<sub>rxm</sub> requires less than 60 seconds to Uninstall and Re-Install. In fact, you can even skip the Uninstall part, and simply INSTALL again with your chosen parameters. Simply allow the system to overwrite existing Files when it prompts you about this.

A: Changing the 'Listen' Port

In our example of the Installation (see the Installation Manual), we have used port 9465 is the HTTP Port where Browsers will connect to the Portal. This means that any link from our standard Website to the INFOLAB<sub>rxm</sub> Portal must look something like the following (not what the User will see, but the URL related to any link that might be displayed) :-



It is sensible to make a sound decision as to which Port will be used for HTTP. If you wish to change it afterwards you can do so, but it will be necessary to then change any Links from your Website, as well as the Portal Configuration file.

On the Installation Path, e.g. at C:\Infolab\WebPortal\conf, you will find a File called 'server.xml'. Line 4 in this File defines the HTTP Port, which you can discern in the example below as '9465'. This value needs to be edited if you want to change the HTTP Port.

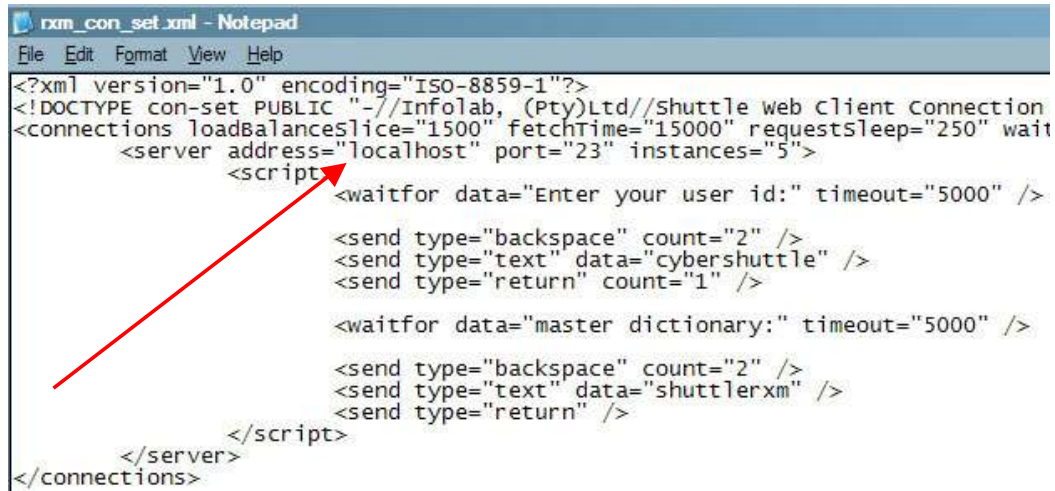
```

<Server port="8005" shutdown="SHUTDOWN" debug="0">
  <Service name="swP-HTTP">
    <Connector className="org.apache.catalina.connector.http.HttpConnector"
      port="9465" minProcessors="5" maxProcessors="75"
      enableLookups="true" redirectPort="8443"
      acceptCount="10" debug="0" connectionTimeout="60000"/>
    <Engine name="Standalone" defaultHost="localhost" debug="0">
      <Logger className="org.apache.catalina.logger.FileLogger"
        prefix="catalina_log." suffix=".txt"
        timeStamp="true"/>
      <Realm className="org.apache.catalina.realm.MemoryRealm" />
      <Host name="localhost" debug="0" appBase="webapps" unpackWARs="true">
        <Valve className="org.apache.catalina.valves.AccessLogValve"
          directory="logs" prefix="localhost_access_log." suffix=".txt"
          pattern="common"/>
        <Logger className="org.apache.catalina.logger.FileLogger"
          directory="logs" prefix="localhost_log." suffix=".txt"
          timeStamp="true"/>
        <Context path="/manager" docBase="manager" debug="0" privileged="true"/>
      </Host>
    </Engine>
  </Service>
  <Service name="swP-WARP">
    <Connector className="org.apache.catalina.connector.warp.WarpConnector"
      port="8008" minProcessors="5" maxProcessors="75"
      enableLookups="true"
      acceptCount="10" debug="0"/>
    <Engine className="org.apache.catalina.connector.warp.WarpEngine"
      name="Apache" defaultHost="localhost" debug="0" appBase="webapps">
      <Logger className="org.apache.catalina.logger.FileLogger"
        prefix="apache_log." suffix=".txt"
        timeStamp="true"/>
      <Realm className="org.apache.catalina.realm.MemoryRealm" />
    </Engine>
  </Service>
</Server>

```

**B: Changing the INFOLAB Enterprise Server IP Address**

On the Installation Path, e.g. at C:\Infolab\WebPortal\webapps\INFOLABrxm\WEB-INF you will find the file 'rxm\_con\_set.xml'. If you wish to change the IP address that INFOLAB<sub>rxm</sub> should use to find the INFOLAB Enterprise Server, then you need to edit the value in this File. In our example, 'localhost' is specified.



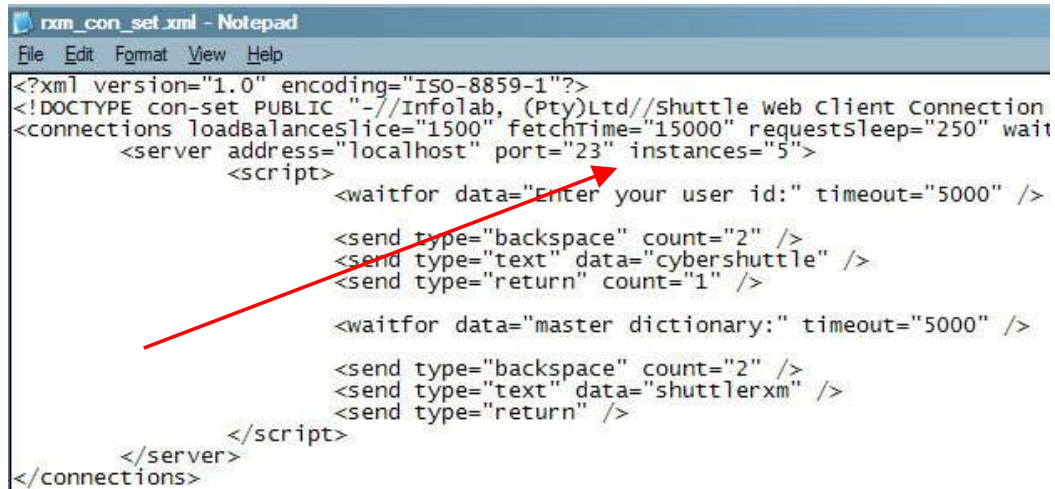
```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE con-set PUBLIC "-//Infolab, (Pty)Ltd//Shuttle web Client Connection
<connections loadBalanceslice="1500" fetchTime="15000" requestSleep="250" wait
  <server address="localhost" port="23" instances="5">
    <script>
      <waitFor data="Enter your user id:" timeout="5000" />
      <send type="backspace" count="2" />
      <send type="text" data="cybershuttle" />
      <send type="return" count="1" />
      <waitFor data="master dictionary:" timeout="5000" />
      <send type="backspace" count="2" />
      <send type="text" data="shuttlerxm" />
      <send type="return" />
    </script>
  </server>
</connections>

```

**C: Adjusting number of Allowed Connections**

The number of Connections allowed between INFOLAB<sub>rxm</sub> and INFOLAB Enterprise Server is specified in the same file 'rxm\_con\_set.xml' as in the previous step. Our example allows 5 Connections.



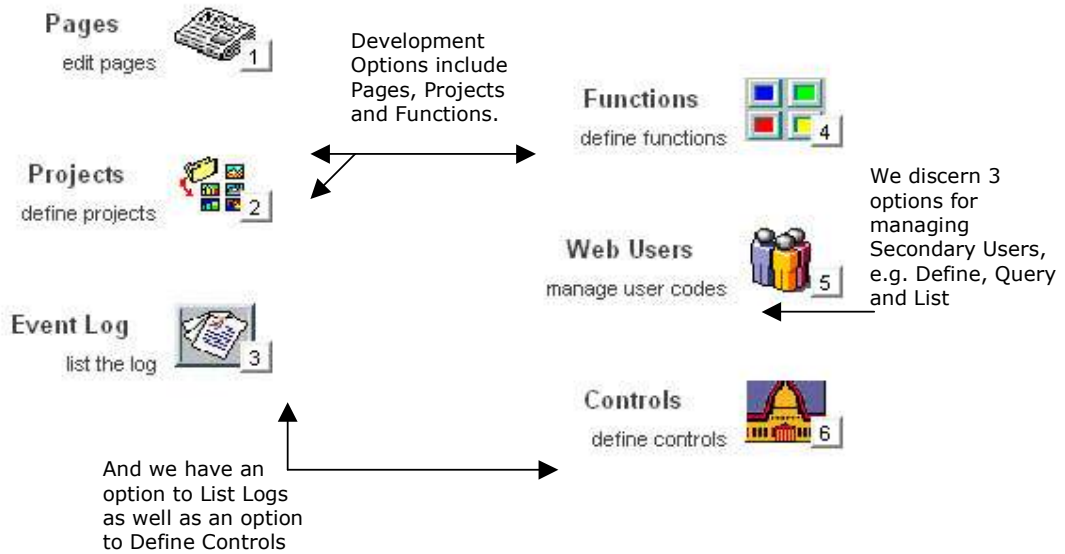
```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE con-set PUBLIC "-//Infolab, (Pty)Ltd//Shuttle web Client Connection
<connections loadBalanceslice="1500" fetchTime="15000" requestSleep="250" wait
  <server address="localhost" port="23" instances="5">
    <script>
      <waitFor data="Enter your user id:" timeout="5000" />
      <send type="backspace" count="2" />
      <send type="text" data="cybershuttle" />
      <send type="return" count="1" />
      <waitFor data="master dictionary:" timeout="5000" />
      <send type="backspace" count="2" />
      <send type="text" data="shuttlerxm" />
      <send type="return" />
    </script>
  </server>
</connections>

```

**Maintaining Dependencies with INFOLAB Enterprise Server**

Below is a picture of the Portal Wizard in INFOLAB.



1a: Secondary User Master –

The Secondary User Master is used when there is a need for Users to identify themselves with a proper Logon, i.e. UserCode and Password. This User Type should not be stored on the normal INFOLAB User Master, which is reserved for 'internal' Users, i.e. those that use the INFOLAB Applications with Menu Access. A typical use for Secondary Users would be a case where your Customers who access their Accounts, Orders or other Transactions with your Company have Secondary User Codes so that they may be properly identified before gaining access to their Records with your Company. There are specific routines provided for storing and retrieving these User Credentials during the course of a Portal Session (see PORTAL Code when writing programs). For example, as soon as a Secondary User requests access to a Page that requires User Credentials, you may link the session to a Page with a Logon. Once the Logon credentials have been verified, you call the appropriate routine to store the credentials with the SID (Session ID). After that, for the remainder of the Session, the credentials may be retrieved whenever a Page requires it, as logged with the unique SID.

The Secondary User Master provides 'custom' attributes that are reserved for the Developer(s) of your local Portal Services to store any additional data or parameters required by your Application Services, and that will not be affected when you upgrade your system to new releases of IES / INFOLAB<sub>rxm</sub>.

1b: Portal Control Parameters –

On the Control Parameter Screen you need to specify a Startup and Default Call, for handling new Connects and 'No Page' links, respectively. The Event Log may be switched on or off, and is usually only switched on in order to debug problems.

### Startup Call

You need to specify a Subroutine Call here that handles all new Sessions, i.e. determines the 1st Page to serve to such a new Session. The Subroutine is called with 1 parameter(`nex_page_key_to_serve`), and you may access the attendant Values and Parameters through `INFOLABrxm` calls from within your Subroutine.

If there are multiple entry points to the Service, i.e. from different links on the Website, then it is prudent to attach a Variable Name to such links, in order that this variable may be inspected in your Subroutine to determine where the Entry was made from, i.e. if different logic is required dependent on where entry started from.

**NOTE: THE SUBROUTINE SPECIFIED HERE MUST BE CATALOGED IN THE 'INFOLAB' DATAMART, AND WILL BE PROCESSED THERE.**

### Default Call

It is mandatory to provide a Default Subroutine Call to be made whenever `INFOLABrxm` is left without a 'navigator' Parameter.

All Pages, Submit and Hyperlink requests are expected to pass a Variable Name called 'navigator' to the Server. The value of 'navigator' is the next Page to reference, which will determine which Subroutine will be called to handle the processing. If the 'navigator' value is absent for whatever reason, OR invalid, i.e. the Page cannot be found, then the Server will call your Default Subroutine to deal with the next step for the active Session Request.

This Default Subroutine is always called with 1 parameter (`nex_page`), and should return the Key for the next Page to serve after processing of the current request (by the same subroutine).

**NOTE: THE SUBROUTINE SPECIFIED HERE MUST BE CATALOGED IN THE 'INFOLAB' DATAMART, AND WILL BE PROCESSED THERE.**

1c: Event Log –

The Event Log may be switched on or off, and the Log Report will list all steps encountered by the Server in dealing with Sessions (SIDS) while the Log is in the ON state. The information in this Log is sufficient for analysis of what Page Requests have been attended to, Subroutines Called, Values passed, and so on.

© Infolab, 2006

This Documentation is copyrighted by Infolab (Pty) Ltd. [ [www.infolab.cc](http://www.infolab.cc) ] All rights are reserved. Licensed INFOLAB Users are granted permission, for internal use ONLY, to reproduce the Documentation, and to include amendments dealing with specific instructions local to your installation.