



## General Setup

1. Read the developer guidance document found here.
2. Transaction Processor and Host Server:
  1. Run the Voyager Admin tool on the desktop.
    1. NOTE: Only the Valet tool will work on Windows 7.
  2. Make sure your Transaction Processor is green before performing any customizations.
  3. Remove the 2nd Host Server using the Valet tool. Only one is required for development.
  4. Under the Host Server properties, ensure that the directory specified in the ScriptDir property exists, create if necessary.
    1. No files need to be copied here, the directory just has to exist for the TP to start.
3. Get latest code from TFS:
  1. `$/OLB Voyager Upgrade 4.0/Dev/UI`
4. Copy Voyager templates:
  1. Get the latest templates from `$/OLB Voyager Upgrade 4.0/Dev/UI/src/Common/Templates`.
  2. Right-click on the Transaction Processor node in the Valet tool, choose Properties... and note the TemplatePath value.
  3. Copy the entire Templates folder over the Templates folder specified in the Voyager properties.
  4. Restart the TP at some point after copying the templates.
5. Set up custom Voyager script shortcuts:
  1. Copy this folder: `$/OLB Voyager Upgrade 4.0/Dev/UI/Config/Voyager/DEV/TPScripts`
  2. To: `C:\Source`.
  3. Make the new copied folder and all it's contents read-write (clear read-only flag).
  4. Create a "TP" folder under `C:\Source\TPScripts`.
  5. Move all the shortcut (\*.lnk) files from `C:\Source\TPScripts` to `C:\Source\TPScripts\TP`.
  6. Right-click on the task bar, Toolbars, New Toolbar...
  7. Browse to and select the new TP folder (`C:\Source\TPScripts\TP`).
6. Prepare for database deployment
  1. Read up on Visual Studio database projects here.
  2. Read up on Visual Studio database project deployment here.
  3. If you have previously deployed any database project, or manually created any databases:
    1. run the `DropVoyagerTestUser.sql` script from here: `$/OLB Voyager Upgrade 4.0/Dev/UI/Config/Voyager/DEV/TPScripts`
7. Deploy Voyager OOTB databases:
  1. Open Voyager database solution found here: `$/OLB Voyager Upgrade 4.0/Dev/UI/Builds/OLB_Voyager_4.x_Databases.sln`
  2. Deploy the entire solution, per the instructions above.
  3. If you get "User, group, or role 'HBIENT\srsvoyagertst' already exists in the current database" errors, you will need to delete this user from your local database and re-add them.
    1. Run the `DropVoyagerTestUser.sql` script from here: `$/OLB Voyager Upgrade 4.0/Dev/UI/Config/Voyager/DEV/TPScripts`.
    2. Re-deploy the entire database solution, per the instructions above.
8. Deploy Huntington Custom databases:
  1. Open Huntington Custom solution found here: `$/OLB Voyager Upgrade 4.0/Dev/UI/Builds/OLB_Huntington_Custom_Databases.sln`
  2. Deploy the entire solution, per the instructions above.
    1. You will likely receive errors deploying CommonInfo, see this wiki entry for getting a local CommonInfo database that mirrors QA.
9. Clean up after database deployment:
  1. Run the `DropCreateVoyagerTestUser.sql` script from here `$/OLB Voyager Upgrade 4.0/Dev/UI/Config/Voyager/DEV/TPScripts`
  2. If running Voyager from C:\, run the "CleanseDColon" database script to change all D:\ to C:\.
    1. If you created the TP shortcuts as instructed above, click the TP toolbar from the taskbar, click "CleanseDColon".
  3. Using the Valet tool, right-click on Host Server, Properties, change Timeout to 1200 (or some other large value so that your plugin debugging sessions don't timeout).
  4. Update the MQ database values for local development. NOTE: This step may not be needed if the TEST MQ queue manager is changed to the queue service.
    1. `UPDATE [Voyager].[dbo].[CCComponentParameter] SET value = 'EB.ESL.RQ' WHERE Value = 'EB.BREQ.IB'`  
`UPDATE [Voyager].[dbo].[CCComponentParameter] SET value = 'EB.ESL.RS' WHERE Value = 'EB.BREQ.IB'`
10. Grant execute permissions on the Voyager service account (srsvoyagertst for TEST environments)
  1. For TEST: `GrantExecuteOnAllStoredProcsFor_srvvoyagertst.sql`
11. Build the source code:
  1. Open the All Solutions solution found here: `$/OLB Voyager Upgrade 4.0/Dev/UI/Builds/OnlineBanking_All_Solutions.sln`
  2. Build the solution.
    1. Each plugin project in the solution is configured to automatically copy its DLLs to `<root>\OnlineBanking\Plugins`
  3. If you get a "...you must define a system environment variable named RMCustWeb..." build error in the RMAAdapter project:
    1. Follow the instructions to create the missing environment variable.
    2. Rebuild.
    3. Failure to successfully build this project will leave Relationship Manager, Bank Admin, and Retail Admin in an unconfigured state.
12. Deploy plugin DLLs to the appropriate folder:
  1. You may need to manually create the "Registration" folder under `C:\OnlineBanking\Plugins`.
  2. Run the "Copy" script from the custom TP toolbar.
    1. This script will copy all the plugin DLLs from `C:\OnlineBanking\Plugins` to `C:\OnlineBanking\Plugins\Registration`.
    2. The reason for the "Registration" sub folder is so that Voyager can load and lock the plugin DLLs from that folder, allowing the plugin projects to build and copy their DLLs to `C:\OnlineBanking\Plugins` without failing because the DLLs are locked.
  3. Run the "Stop" script from the custom TP toolbar.
    1. If the TP is already stopped, you will get an error. This is ok, ignore it.
  4. Run the "RegisterVoyagerPlugins" script from the custom TP toolbar.
    1. This script will register all plugin DLLs that are in the Registration folder.
  5. Run the "Start" script from the custom TP toolbar.
  6. If you get a "HS Has Not Started Successfully" error, you will need to figure out why.
    1. Make sure that the Voyager database has the "HBIENT\srsvoyagertst" user with "db\_owner" access.
    2. Check [CCLogging].[dbo].[CCLogging] for a mis-behaving plugin.

## Relationship Manager

1. Double-check that the DefaultAppPool and RMAAppPool are set to .NET 2.0.
2. Clean Relationship Manager config file:
  1. Open file `C:\Program Files\Corillian\RelationshipManager\Web\web.config`.
  2. Remove line 49 (duplicate appsetting key): `<add key="WireReport.ShowCountrySearch" value="false" />`
  3. Remove file `C:\Program Files\Corillian\RelationshipManager\Web\Custom\Web.config`, if it exists. Not doing so will result in duplicate app settings key errors.
3. Relationship Manager Validation <Note: Be patient. Relationship Manager can be slow, particularly on the first load.>
  1. Make sure to build the RMAAdapter, BankAdmin, and RetailAdmin projects.
  2. Browse to <http://localhost/support>.
  3. UserID: admin
  4. PIN: 123456
  5. Reporting -> Reports
    1. Should see a list of the default SSRS reports
  6. Relationship -> Customer Support
    1. Should see the customer support dialog
4. Relationship Manager Configuration:
  1. Log in to Relationship Manager using the admin account
  2. System Setup
  3. System Configuration
  4. CUSTOM = Yes
  5. ACHWIRE = No
  6. CORPORATE = NO

## Business Online

1. Create Business Online App Pool:
  1. IIS/Application Pools/Add Application Pool...
  2. Name: "BusinessOnline"
  3. Framework Version: 4.0
  4. OK
2. Set Business Online App Pool identity and other settings:

1. Click "BusinessOnline" app pool, Advanced Settings...
2. Identity: hbienvt\srsvoyagertst, pwd: [v0y@g3r!](#)
3. Ping Enabled: False
4. Ping Maximum Response Time: 9000
5. Shutdown Time Limit: 9000
3. Create "BankAdmin" app pool following the steps for creating a new app pool above.
4. Set up the Business Online web application:
  1. Open the Business Online solution found here: "\$/OLB Voyager Upgrade 4.0/Dev/UI/Builds/OnlineBanking\_Business\_UI\_Build.sln"
  2. Opening the solution should automatically create the "BusinessOnline" web application.
  3. Go to the "BusinessOnline" web application, Advanced Settings...
  4. Application Pool: BusinessOnline
5. Set up the Bank Admin web application:
  1. Open the Bank Admin solution found here: "\$/OLB Voyager Upgrade 4.0/Dev/UI/Builds/OnlineBanking\_BankAdmin\_UI\_Build.sln"
  2. Opening the solution should automatically create the "BankAdmin" web application.
  3. Go to the "BankAdmin" web application, Advanced Settings...
  4. Application Pool: BankAdmin
6. Import CommonInfo data:
  1. See this wiki entry for getting a local CommonInfo database that mirrors QA.
  2. After the backup is restored, remove the user HBIENVT\srsvoyagertst from the CommonInfo database.
  3. Re-deploy CommonInfo .dbproj.
7. Import TokenManagement data:
  1. Drop the existing TokenManagement database (right-click, delete from Management Studio).
  2. Create new blank TokenManagement database.
  3. Right-click new TokenManagement, Tasks, Import Data...
  4. Next
  5. Server name: TSWOLBDB03
  6. Database: TokenManagement
  7. Next
  8. Make sure Destination Server name is "." or "(local)", or equivalent.
  9. Next
  10. Next
  11. Select all tables
  12. Click Edit Mappings..., check "Enable identity insert"
  13. Next
  14. Finish
  15. Re-deploy the TokenManagement database from the .dbproj.
8. Enroll a new user through BankAdmin:
  1. Run the BankAdmin solution.
  2. Company Search
  3. Company ID: 999900, click Search
  4. Click 999900 to select the company.
  5. Click User List, find username: admin, first name: admin1, last name: user
  6. Click Enroll in Voyager
  7. Should get a temp password, use this to log into BOL.

## Retail Online

1. Create Retail Online App Pool:
  1. IIS/Application Pools/Add Application Pool...
  2. Name: "RetailOnline"
  3. Framework Version: 4.0
  4. OK
2. Set Retail Online App Pool identity and other settings:
  1. Click "RetailOnline" app pool, Advanced Settings...
  2. Identity: hbienvt\srsvoyagertst, pwd: [v0y@g3r!](#)
  3. Ping Enabled: False
  4. Ping Maximum Response Time: 9000
  5. Shutdown Time Limit: 9000
3. Repeat 1 and 2 above for the "RetailAdmin" app pool.
4. Set up the Retail Online web application:
  1. Opening the All Solutions solution should automatically create the "RetailOnline" and "RetailAdmin" web applications.
  2. Go to the "RetailOnline" web application, Advanced Settings...
    1. Application Pool: RetailOnline
  3. Go to the "RetailAdmin" web application, Advanced Settings...
    1. Application Pool: RetailAdmin
5. Set up the ASPState database
  1. Execute [Create\\_aspstate\\_database.sql](#)
  2. Execute [Create\\_aspstate\\_job.sql](#)
6. Enroll your first Retail user:
  1. Run RM from <http://localhost/support>.
  2. Login using admin | 123456.
  3. Navigate to Custom/Retail Online Banking/Delete Customer.
  4. Enter a known good retail user (mittens0, kriswear1, roltest123, roltest456, etc.)
  5. Click Add User.
  6. You should get a "0:OK" message, or a "user already exists" error.
7. (Optional) Import HNBContentMgmt table to get hints on Retail pages:  
Follow similar instructions to importing the TokenManagement database in BOL.