Aptean

C-MCS Technical Guide

CALIDUS MCS

20th November 2019 - 1.0
Reference: FS 320722

Contents

1 Development Notes

Note: This is a developer guide.

1.1 Pre-Requisites

- Visual Studio 2017
- .NET Framework 4.7 Runtime
- Oracle Client 11.0.2 or 11.0.3
- Oracle ODT with ODAC 11.2.0.3

Installers are available here: See here:

- \\scansrv1\Global\OBS Logistics\Installs\ODTwithODAC112030.zip
- \\scansrv1\Global\OBS Logistics\Installs\win32_11gR2_client.zip

1.2 Oracle Set-up

Install the Oracle Client.

Note: Ensure that the client is installed into a folder within the C:\app structure, for example:

```
C:\app\UserName\product\11.2.0\client_1
```

Install Oracle ODT into same base folder.

Set up TNS Names.

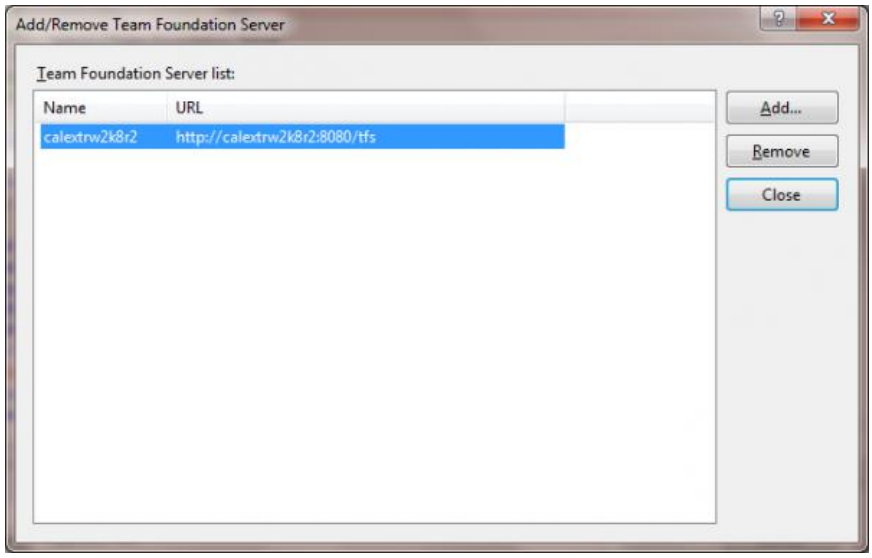
1.3 Code Set-up

Note: You must be provided access to the correct source control library - contact your .NET development manager for details. The following instructions show an example - TFS settings may vary.

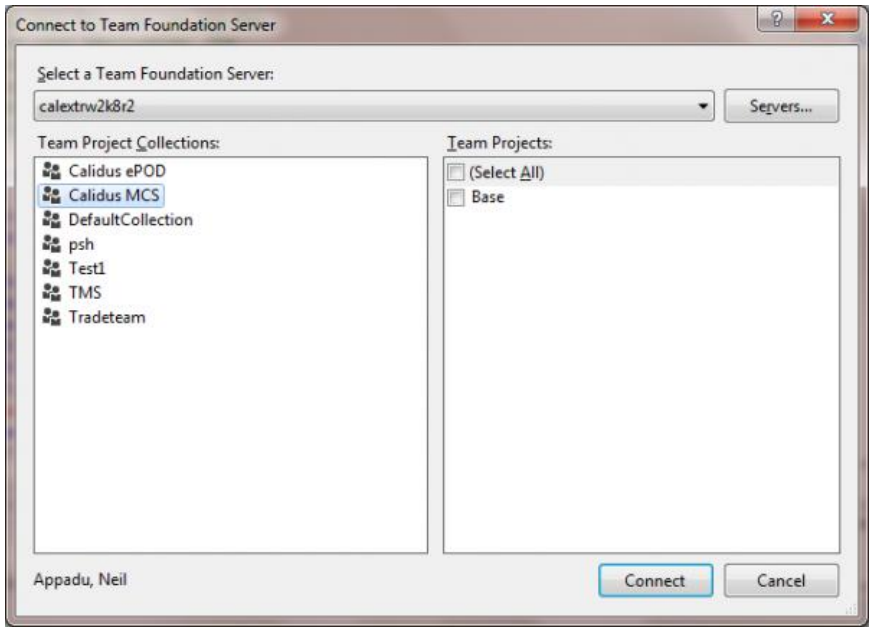
1) Copy \\spekefs2012\Projects\Development\MCS\# Source\Start\Calidus MCS to where you want your project setup locally. **Note:** This was correct at the time of document creation. There may be updated projects to copy, depending on the version - check the folder \\spekefs2012\Projects\Development\MCS\# Source for updated folders or zip files

2) Startup Visual Studio.

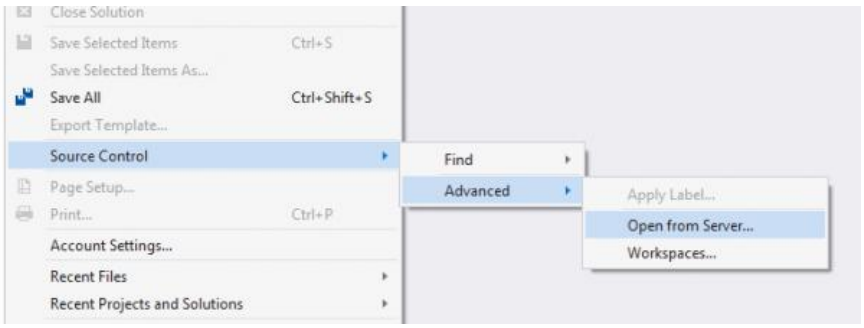
3) Configure TFS server.

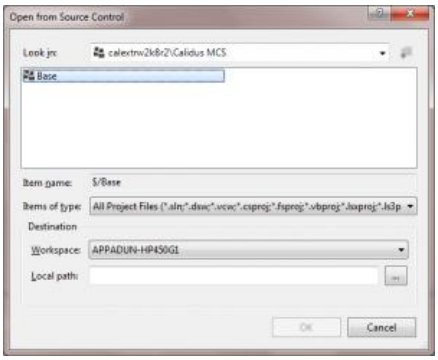


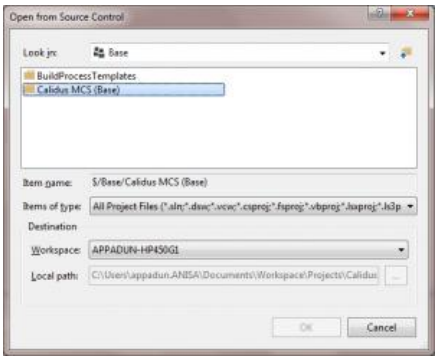
4) *CALIDUS* MCS source can be found underneath Calidus MCS.

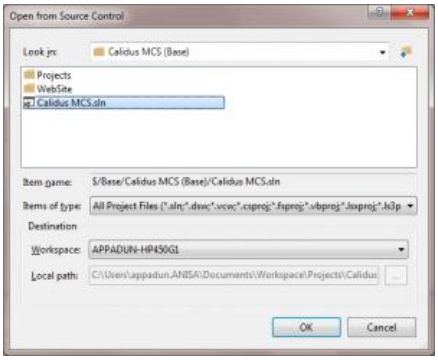


5) Once that is setup, open the project from source control.









(You might need to specify a local path at this point - match it up to where you put you local copy of MCS earlier). The system will prompt if you should overwrite existing files - elect to overwrite.

6) Once that?s done, you should have access to all the code.

1.4 Project Set-up

7) Click on the CustomServerControls project and open References.

8) Remove OracleDataAccess.

9) Right-click on References and add reference to the installed Oracle.DataAccess.dll. This will be located within the Oracle home directory, usually here:

a. C:\app\UserName\product\11.2.0\client_1\odp.net\bin\4\Oracle.DataAccess.dll

10) Repeat these steps for the DatabaseClassLibrary project.

11) Build the solution.

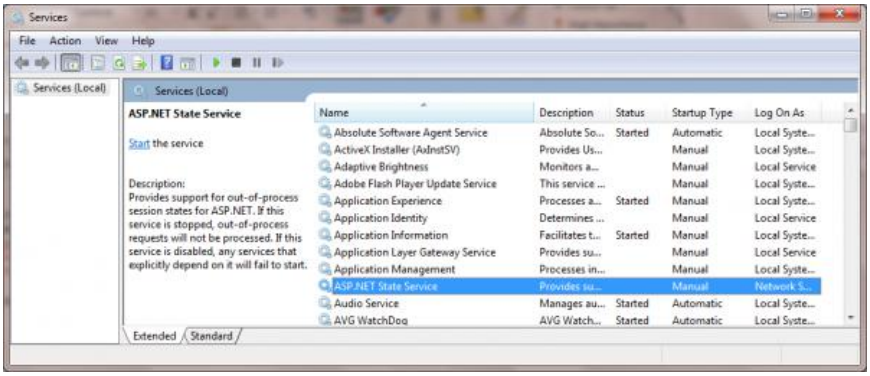
1.5 Runtime Set-up

12) Configure your web.config.

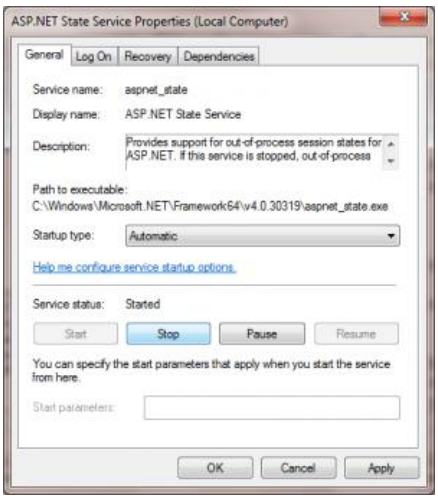
These 2 settings need to point to the files under App_Data.

```
<appSettings>
  <add key="MessageFile" value="C:\Users\appadun.ANISA\Documents\Workspace\Projects\Calidus MCS\WebSite\Ap
  <add key="KnownBadListFile" value="C:\Users\appadun.ANISA\Documents\Workspace\Projects\Calidus MCS\WebSi
```

13) The application uses the ASP session state service to handle sessions. You?ll need to ensure that the session state service is enabled and started on your PC.



Advise that you change it to automatic startup if it is currently set to "Manual".



14) Run the app.

2 MCS Build and Release Guide

Note: This is a developer guide.

2.1 Pre-Requisites

See Development Notes for details of development environment set-up.

Start with the QA TFS stream.

Create a new TFS branch for the new version.

Ensure that all code has been brought in or created in this stream.

Change the JavaScript and Build versions:

- app.js - change MCS.App.Version to the new version number.
- Web.config - change JSVersion to the new version number.

Ensure that the code builds.

Check in all code changes to the new TFS branch.

2.2 Build the Release

Right-click on *WebSite*.

Select *Publish Web App*.

- Connection:
  - File System - specify a local target location.
- Settings:
  - Configuration: Release.

Click **Publish**.

2.3 Prep for Release

Copy the local release folder and contents to \\spekefs2012\Projects\Releases\MCS\Release\.

Rename *Web.config* to *Web.config.release*.

Rename this copied folder to the version name e.g. MCS_v.vv.vv.vvv.

Zip the folder to the same name.

2.4 Release Documentation

Create a new release note in this Assist system:

- From the navigation menu, choose *Help/Creating new documents from templates*.
- Choose *Create release note*.
- Enter a release note name in the format "REL (System) (Version)", for example "REL C-MCS v2.00.00.000".
- Click **Create Release note**.
- In the page that opens, change the 4 lines here to reflect the release note details:

```
{{#vardefine:System|''CALIDUS'' Mobile/TTM/Vision/ePOD, etc}}
{{#vardefine:SystemCode|WCS/MCS/EPOD, etc}}
{{#vardefine:Version|Release version no}}
{{#vardefine:Date|7th February 2020}}
```

- For example:

```
{{#vardefine:System|''CALIDUS'' MCS}}
{{#vardefine:SystemCode|MCS}}
{{#vardefine:Version|2.00.00.000}}
{{#vardefine:Date|7th February 2020}}
```

- Update *Platforms/Database* section with any restrictions that the product has regarding releases, for example operating system, browser, etc.
- Update *Related Documents* section to note any controlled documents that might be of use here. For example, if there is an Assist page that identifies the functionality changes, reference that.
- Update *User Notes* section to update site, or remove this line if inapplicable.
- Update *Released Incidents / Changes* section to include a line for each change included in the release notes. Add a line with the log references and title of the log, plus a description of the functionality added. Describe in user terms, not technical terms. For example "Fixed a bug where printer cannot be selected at Pallet Build" is a good comment, whereas "Fixed a bug in ui103lt3b_001.cs to add onLoad event" is not.
- Update *Setup* section to include any configuration steps that are required by the user. Be detailed with the individual flags and parameters, identifying system if necessary.
- Update *User Instructions* section, explaining where process has changed because of the changes included in this release, including the log reference.
- Update *Performance Notes* section, to indicate where performance of the system or the operation might be impacted by the changes in this release.
- Update *Related Releases* section, to show related releases, for this system or any other system. For example, if an MCS release is dependent on a specific C-TMS release, reference that here.
- Update *Technical Changes & Special Notes* section, for any other technical information that the customer might need to know. For example, if the changes require specific hardware or hardware configuration, for example a scanner, a signature pad, a specific monitor size, list this here. Any system tasks or changes that must be made to the application that cannot be made by the customer should also be listed here.

Save the release note in \\spekefs2012\Projects\Releases\MCS\Release Notes\.

Update the release log in \\spekefs2012\Projects\Releases\MCS\.

2.5 Release

On the customer machine, locate the customer MCS installation to be updated.

Delete any prior back-ups if appropriate.

Back-up the existing folder and zip it.

Copy the new release into the MCS installation folder.

Extract the contents of the new release zip over the existing folder.

Check and compare *Web.config* to *Web.config.release* - make any changes to the customer MCS installation version to reflect new keys added to the file.

Restart the system through IIS.

Inform the customer through email with the release note attached.

3 MCS Server Installation

3.1 Info

Latest builds can be found in the following directory on the projects drive

```
P:\Development\MCS\Release Logs\Release
```

The release notes and change log can be found at the following location, it will be the developers responsibility to fill in these documents

```
P:\Development\MCS\Release Logs
```

LFS is currently the only client which uses MCS the QA box can be accessed from IP 172.151.45.173

3.2 Dependencies

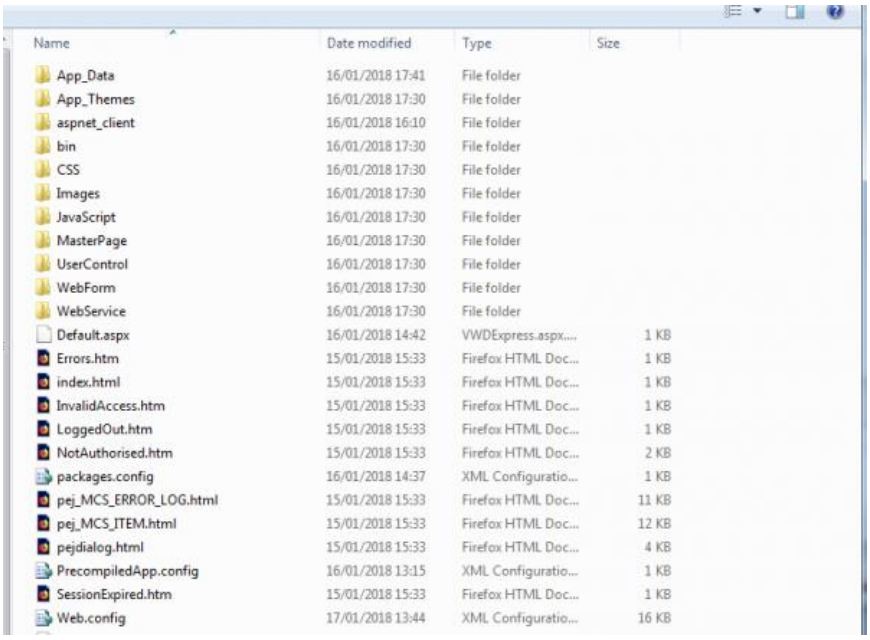
1. .NET Framework 4.7 Runtime
2. Oracle Client 11.0.2 or 11.0.3
3. Oracle ODT with ODAC 11.2.0.3

3.3 General Release Process

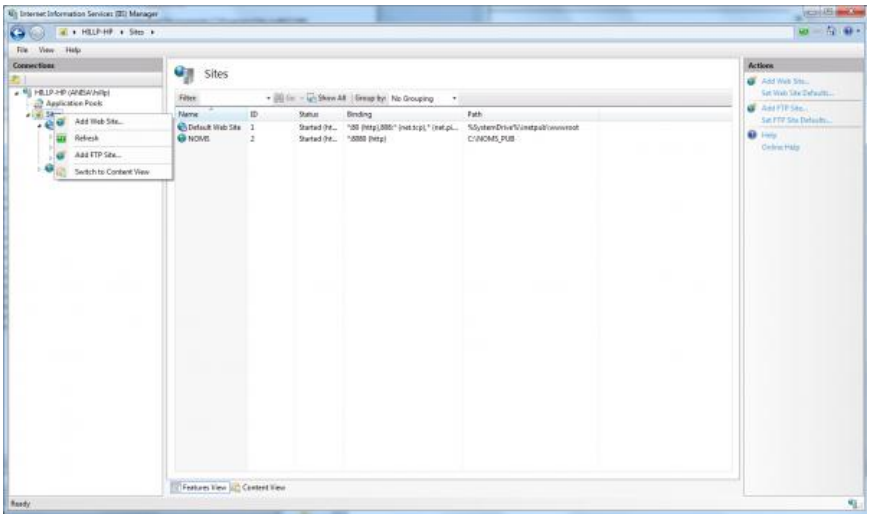
Copy the latest build zip file to server(I usually do to the desktop for convenience, then delete when I?m finished)

Extract zip file to a new directory somewhere on the box(for example C:\Program Files (x86)\OBS Logistics\MCS)

Contents of directory should looks as follows

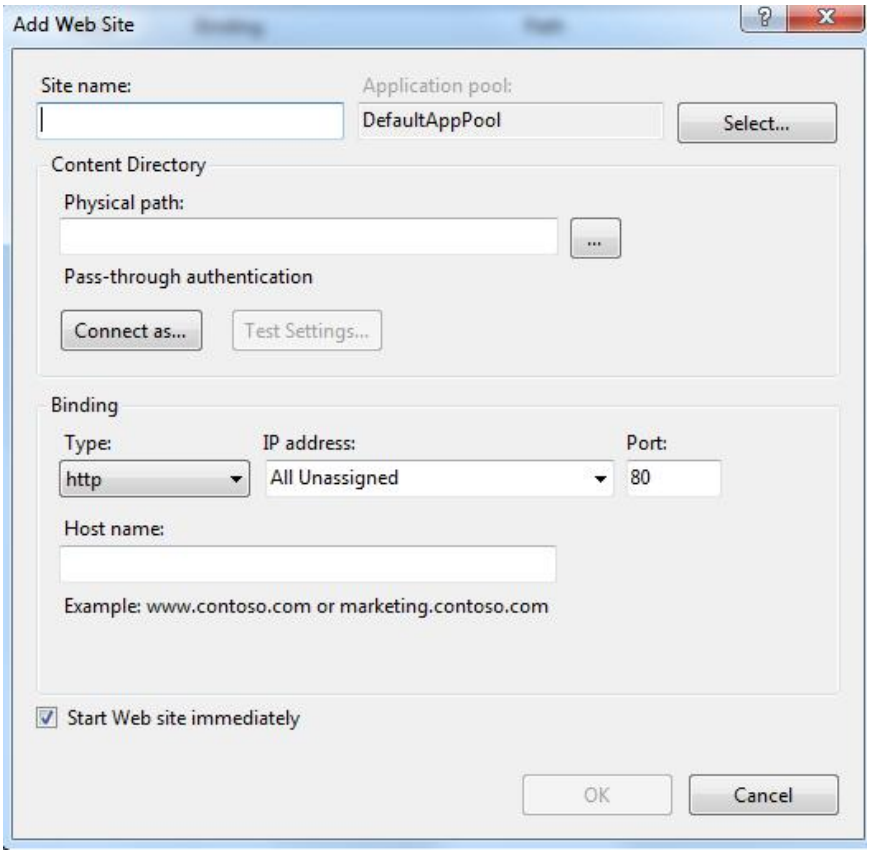


Open the "inetmgr" program



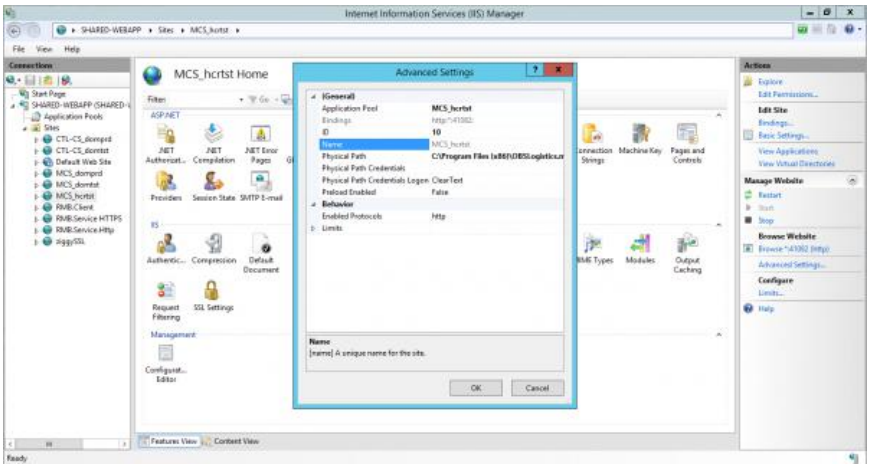
Right click sites, then "Add Web Site.." Enter the site information in the following popup,

- Site name - internal reference(product name and client e.g. "MCS_NOMS")
- Physical path should be the directory the MCS source files where extracted to
- The "connect as" button is used for configuring security settings, this should be left at the default "pass-through authentication"
- The port will need to be unique for the box
- I believe https certificates are usually applied using a reverse-proxy apache server, so leave the type on "http"

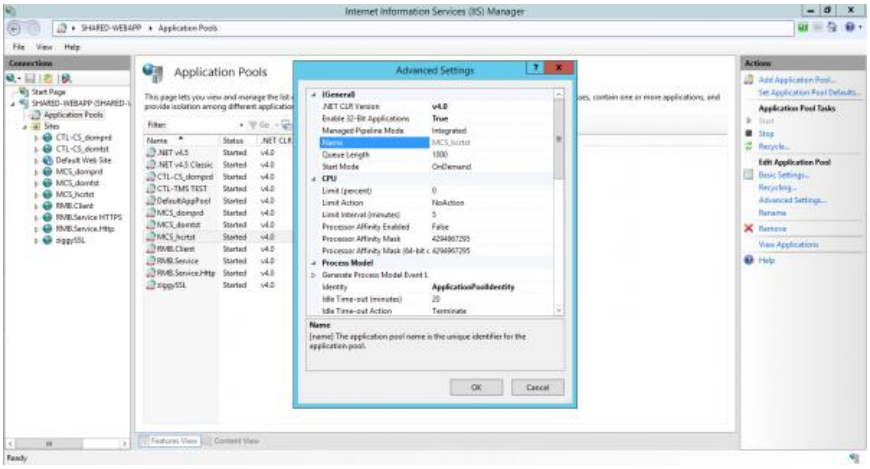


By default, creation of a new site in IIS will create a new Application Pool of the same name. This must be capable of running 32bit apps. To achieve this:

Note the Application Pool created, or find by clicking the site in the Connections explorer, and clicking Advanced Settings in the action pane.



Click Application Pools in the Connections explorer, click the application pool and click Advanced Settings in the Actions pane. Double-click Enable 32-Bit Applications to set the value to True and click OK.



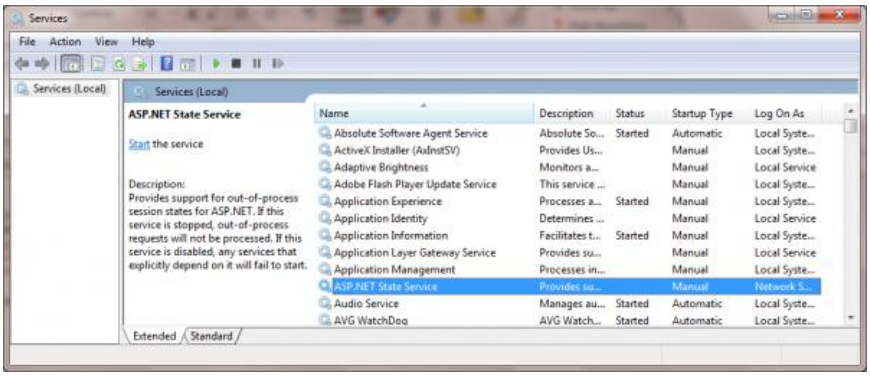
The app configuration is done through the web config file, the "MessageFile" and "knownBadListFile" will need their location updating, while the email addresses and the "connectionString" will need updating for the specific client.

```
<configuration xmlns="http://schemas.microsoft.com/.NetConfiguration/v2.0">
  <appSettings>
    <add key="MessageFile" value="C:\inetpub\wwwroot\MCS\WebSite\App_Data\msg_file.txt" />
    <add key="KnownBadListFile" value="C:\inetpub\wwwroot\MCS\WebSite\App_Data\knownBadList.txt" />
    <add key="TestMode" value="false" />
    <add key="DebugConnections" value="false" />
    <add key="INV_EMAIL_TO" value="n.appadu@obs-logistics.com" />
    <add key="INV_EMAIL_FROM" value="n.appadu@obs-logistics.com" />
    <add key="autologin_user" value="obsdhl" />
    <add key="autologin_pass" value="obsdhl" />
    <add key="MasterStyleSheet" value="MMPS.css" />
    <add key="PersonalisationMessage" value="AUTO" />
    <add key="WelcomeMessage" value="Welcome to Calidus WMS" />
    <add key="WelcomeTitle" value="Welcome" />
    <add key="JSVersion" value="054" />
    <add key="MissingReasonCode" value="MI" />
    <add key="LateReasonCode" value="LATE" />
    <add key="FreightNotReceivedReasonCode" value="FNR" />
    <add key="SystemType" value="WMS" />
  </appSettings>
  <connectionStrings>
    <add name="calidus" connectionString="Data Source=wmsdev;" providerName="System.Data.OracleClient" />
  </connectionStrings>
```

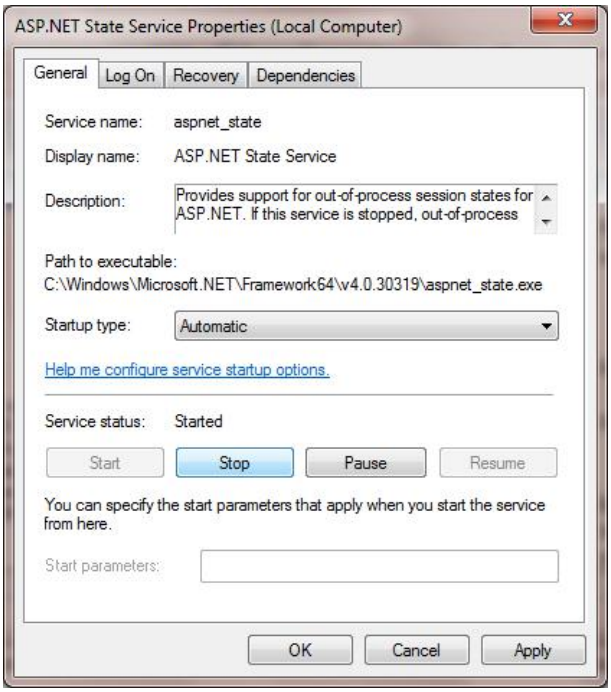
3.4 Session State Service

In addition to the site setup the following service may need some setup as well

The application uses the ASP session state service to handle sessions. You?ll need to ensure that the session state service is enabled and started on your pc.



Advise that you change it to automatic startup if it is currently set to "Manual"



A.3 Document History

Version	Date	Status	Reason	By
1.0	23/12/2019	Issue	Initial Version	ANW

A.4 Authorised By

Julie Scott	OBS Manager	_____
Tony Walker	OBS Consultant	_____

A.3 Document History

Version	Date	Status	Reason	By
1.0	20/11/2019	Issue	Initial Version	ANW

A.4 Authorised By

Julie Scott	OBS Manager	_____
Tony Walker	OBS Consultant	_____

A.3 Document History

Version	Date	Status	Reason	By
1.0	20/11/2019	Issue	Initial Version	ANW

A.4 Authorised By

Julie Scott	OBS Manager	_____
Tony Walker	OBS Consultant	_____

4 Technical Notes

Note: This is a technical working document and always under draft.

4.1 DOCUMENTATION

Existing Documentation:

- P:\Development\MCS\# Documentation\MCS Technical Details v1.0.docx
- P:\Development\MCS\# Guides\Calidus MCS Development Setup.docx
- P:\Development\MCS\# Guides\MCS Cleardown Instructions v1.0.docx
- P:\Development\MCS\# Guides\MCS Development Guide v1.0.doc

User Guides:

- P:\Development\MCS\# Guides\User Guide\CMCS User Guide - Refresh MCS v1.0.doc
- P:\Development\MCS\# Guides\User Guide\UG 320722 CALIDUS MCS C-TMS User Guide v1.0.pdf
- P:\Development\MCS\# Guides\User Guide\UG 320722 CALIDUS MCS Device User Guide v1.0.pdf

Also, on this guide:

- MCS Device User Guide
  - Category:UG 320722 CALIDUS MCS Device User Guide
- CTMS-MCS User Guide
  - Category:UG 320722 CALIDUS MCS C-TMS User Guide
- Developer Guides (including this document)
  - Category:Developer Guides

4.2 C-TMS NOTES

4.2.1 PARAMETERS

System-wide:

- MCS_PREVENT_UNPLANNED - Prevent unplanned items being scanned for pallet building, despatch and receipt although expected for the route (Y/N)?

System-wide parameters like this are maintained in CTMS.

Location-specific:

- MCS Active
- Receipt Only
- Despatch Only

4.2.2 PACKAGES

MCS is mentioned in several places in CTMS code.

The business logic is all based in DP_MCS_SCANNING, although there are several functions that contain some (duplicated?) code as well:

- fn_add_3rd_party_label
- fn_add_to_pallet
- fn_cancel_booking
- fn_create_pallet
- fn_damages
- fn_desp_comp
- fn_desp_conf

- fn_goods_rec
- fn_goods_rec_comp
- fn_goods_rec_tmp
- fn_open_pallets_data
- fn_package_data
- fn_pallet_swapping
- fn_print_pallet_label
- fn_remove_from_pallet
- fn_update_pallet_status
- fn_update_service
- fn_validate_piece

Most are duplicated in DP_MCS_SCANNING and these are the ones that are called from MCS (at least for all the ones that I have checked). Also the following package (not yet checked).

- RECEIVED_OFF_PLAN

4.3 MCS PROCESSING AND VALIDATION IN C-TMS

4.3.1 Logging On

Logging on:

- User:
- Straight list of users, user_based_at plus others from ADMIN_USERS. Must have:
  - COST_CENTRE parameter set up
- Site:
  - must be a location on geo_location with DEPOT = "RDC" and MCS_ACTIVE = "Y".

4.3.2 Pallet Building

Pallet Building:

- Previously-configured pallets:
  - DP_MCS_SCANNING.FN_VALID_ITEM_FOR_PALL
  - Carrier must be set to allow pallets (res_carrier.allow_pallets = Y). If hazardous product, must also allow_dg_pallets=Y.
  - Package must be on a trip which matches locations for packages already on the pallet.
  - Class must be compatible (not on prd_non_compatible, comparing the packages SOI.class to each soi for sch_ship_pall_item on the shipment pallet)
  - Product type must be compatible (not on prd_prod_non_comp, comparing the packages SOI.prod_type_id to each soi for sch_ship_pall_item on the shipment pallet)
  - Counting hazardous classes, the count of classes on the pallet must not exceed the maximum amount (res_carrier_haz_qty.max_qty)
- New pallets:
  - DP_MCS_SCANNING.FN_CREATE_PALLET
  - The pallet is simply created with pallet status "OPEN" and location_id (the from location) set as the depot at which it was built.
  - IMP: We need a sequence setting in the database: SEQ_SCH_SHIP_PALL. It's quite obviously a global thing, so we would need to make sure that the pre-printed stationary are outside that range.
- Adding items to pallets:
  - DP_MCS_SCANNING.FN_ADD_TO_PALLET
  - Pallet must exist on sch_ship_pall.
  - Validation exists whereby items cannot be added to the pallet if the pallet status is not OPEN (sch_ship_pall.pallet_status).
  - Package must not already be on the pallet.
  - Calls FN_VALIDATE_PIECE (is this the one in the package or the one in FUNCTIONS? TBC)
  - Package and pallet must match trips.
  - The pallet locations are updated with the from location being the depot at which the pallet is built, and the to location being the next unload for that order at an RDC depot.
  - If an unload at a depot is not found, then the location returned is the destination location of the item's order

- Printers - printers must be set up on ADM_DFLT_PRINTER for the user in C-TMS
- Pallet Label:
    - DP_MCS_SCANNING.FN_PRINT_PALLET_LABEL
    - Report is fixed to "LFS_PALLET_LABEL"

4.3.3 Receipt

Receipt:

- Trips
    - SQL to validate is in V_MCS_TRIP_ITEM which is very complex - needs checking
- Items:
    - SQL to validate is in V_MCS_TRIP_ITEM which is very complex - needs checking
    - DP_MCS_SCANNING.FN_GOODS_REC
    - Updates items as reason code SU (Successfully Unloaded)
    - Sets SCH_ORDER_ITEMS.qty_delivered = 1 if this is the final destination for the order.
    - Checks system adm flag MCS_PREVENT_UNPLANNED. If set, will not allow receipt of items from another order not planned on this trip. If set, allows this and splits the original order and adds it to this trip.
    - Checks specific FNR (Freight Not Received) reason code. Sets qty_to_deliver to 0. If all items on that order are not received, the order is unscheduled from the trip and audited.
    - Note: The actual reason code used for FNR processing can be changed in C-MCS (in the web.config file) but this is fixed in C-TMS packages, so it should never be changed.
    - Checks reason.at_fault of reason code applied to an item. If "LATE", sets the late_order flag against the OMS ref. Automatically unschedules the order from the onward delivery trip.
- Completing:
    - DP_MCS_SCANNING.FN_GOODS_REC_COMP
    - Any items not received are marked with reason code NU (Not Unloaded) - needs to be set up as Item_Non_Con
    - Updates the trip/stop with actual arrive (now) and actual depart (now + 15) only for DL/CL stop types.
    - Updates any missing stop times (not sure how)
    - Completes any prior flight trips (which shouldn't fire - defined as immediately prior trip with AWB/Consignment ID)
    - Sets trip status to COMPLETED.

4.3.4 Despatch

Despatch:

- Trips
    - SQL to validate is in V_MCS_TRIP_ITEM which is very complex - needs checking
- Items
    - SQL to validate is in V_MCS_TRIP_ITEM which is very complex - needs checking
    - DP_MCS_SCANNING.FN_DESP_CONF
    - Updates items as reason code SL (Successfully Loaded)
- Completing:
    - DP_MCS_SCANNING.FN_DESP_COMP
    - Any items not despatched are marked with reason code NL - needs to be set up as Item_Non_Con
    - If gl.despatch_only = Y
        - ◊ Updates the trip/stop with actual arrive (now-15) and actual depart (now) only for SU stop types
        - ◊ Sets trip status to EN-ROUTE.
    - Else
        - ◊ Updates the trip/stop with actual arrive (now-15) and actual depart (now) only for SU/PK stop types
        - ◊ Sets trip status to COMPLETED.
    - Completes any onward flight trips (which shouldn't fire - defined as immediately following trip with AWB/Consignment ID)

4.3.5 Damages

Damages:

- Reason codes - item non-con codes from SCH_REASON_CODES for the default MCS cost centre, set up in MCS_CONTROL_PARAM.CTP_KEY = "DEFAULT_COST_CENTRE"
- Calls process FN_DAMAGES.
- Creates or updated SOIRC at the depot to reason code or 'MI' if not provided.

- Sets SOI to qty 0.
- This triggers an update of STI through trigger TRG_SCH_ORD_ITEMS_ACTUALS. This sets ALL STI records (that have a null delivered qty) to the DAMAGED reason code on the SOIRC (set above), or 'MI'.

4.4 PROCESSING ON MOBILE DEVICE

Sites are loaded initially into the local storage database in the browser. To remove or refresh them, the database cookie must be deleted.

4.4.1 MCS DESPATCH

- A loaded item may be unloaded by scanning it again, with confirmation
- A loaded item cannot be marked as an exception (damaged) - it must be unloaded first.
- Despatch cannot be completed (through the Comp button) unless all items have been scanned, manually loaded through the Load button or marked as exception through the EXC button or scanned as Exception using the EXC slider.
- To mark all items remaining on a trip with an exception code, click the EXC button. Select a Reason Code from the drop-down list and click Confirm. The device will ask you to confirm.
- Clicking on the list removes the floating header and fixes it to the top of the list. Clicking again restores it as a floating header.
- To show loaded items on a trip, click the Show button - they are shown at the end of the list. Note that, if the screen is limited, and there are more pending items to scan that can be shown on a single screen, the Loaded items are not shown. Use the LIM slider to show all items on a trip.
- To show all items remaining, click the LIM slider to ALL. This will show all items, including any loaded, if Showing all items including Loaded ones.

4.4.2 MCS RECEIPT

- Clicking on the list removes the floating header and fixes it to the top of the list. Clicking again restores it as a floating header.
- LIM slider controls whether only limited numbers of items are displayed on the screen. Click it to slide to ALL to show all items.
- Items may be scanned as Exception using the EXC slider.
- To mark all items remaining on a trip with an exception code, click the EXC button. Select a Reason Code from the drop-down list and click Confirm. The device will ask you to confirm.
- Receipt confirmation is through the Update button.
- If not all items have been received when the Update button is pressed, the application will confirm whether they will be marked as reason "Freight Not Received".
- To show received items on a trip, click the Show button - they are shown at the end of the list. Note that, if the screen is limited, and there are more pending items to scan that can be shown on a single screen, the received items are not shown. Use the LIM slider to show all items on a trip.
- To show all items remaining, click the LIM slider to ALL. This will show all items, including any received, if Showing all items including received ones.

4.4.3 MCS DAMAGES

To trigger the Damages (Exception) process

- At Despatch:
  - Enter an item then click EXC.
  - Switch the ITM/EXC slider to EXC, then scan an item or key an item and hit Enter.
- From the menu:
  - Select Damages

- Reason codes are ITEM_NON_CON reason codes from the default cost centre, defined in the MCS system parameters in CTMS.

4.5 Q&A

4.5.1 FAQ

Can we trigger SmartPOD journey from completion of MCS despatch?

- There is a SYSTEM flag MIC_SEND_ACCEPTED to determine the sending at ACCEPTED status.
- Used in TRG_SCH_TRIP_XML_INT, which fires when SCH_TRIP is updated.
- SCH_TRIP is updated for status EN-ROUTE by MCS DESPATCH, for loading at SU.

Can we change triggering of SmartPOD to be at EN ROUTE?

- There is a SYSTEM flag MIC_SEND_ACCEPTED to determine the sending at ACCEPTED status.
- Used in TRG_SCH_TRIP_XML_INT
- Set this to "N" (or unset) to get messages triggered from EN-ROUTE status.

What criteria is required against the locations to generate MCS items?

- Generates onto SCH_TRIP_ITEMS, per stop ID.
- Created in the following packages:
    - DP LOGIX_CONFIG - when adding, deleting or moving SCH_HAULAGE_ACTIVITY records (SHA).
    - TRM - when adding, deleting or moving SCH_HAULAGE_ACTIVITY records (SHA)
- Only on Load or Unload where the location is MCS_ACTIVE="Y"

Is there already a mechanism of MCS pallet closure?

- There is a function FN_UPDATE_PALLET_STATUS, but this is not called by any code in the C-TMS database or C-MCS. It may be called from the C-TMS Shipment Pallet screen itself.

Removing items from pallets?

- There is functionality to do this (FN_REMOVE_FROM_PALLET in .NET DPMcsScanningPKG1.cs and in C-TMS DP_MCS_SCANNING) but it seems that it's never called. We should bear this in mind for the future. For now, simply scan the package to another pallet if incorrectly placed.

What trip status are items available to be received/despatched/pallet Built?

- ACCEPTED

4.5.2 OUTSTANDING QUESTIONS

Can we generate pallets without AWB?

- Unknown

Can we use MCS at non-SU stops i.e. loading at warehouse?

- ??? It appears so, but requires testing.