LIBERO MANUFACTURING

Created by

E-EVOLUTION

TECHNICAL GUIDE

Written by

REDHUAN D. OON

Version 0.1

sponsored by

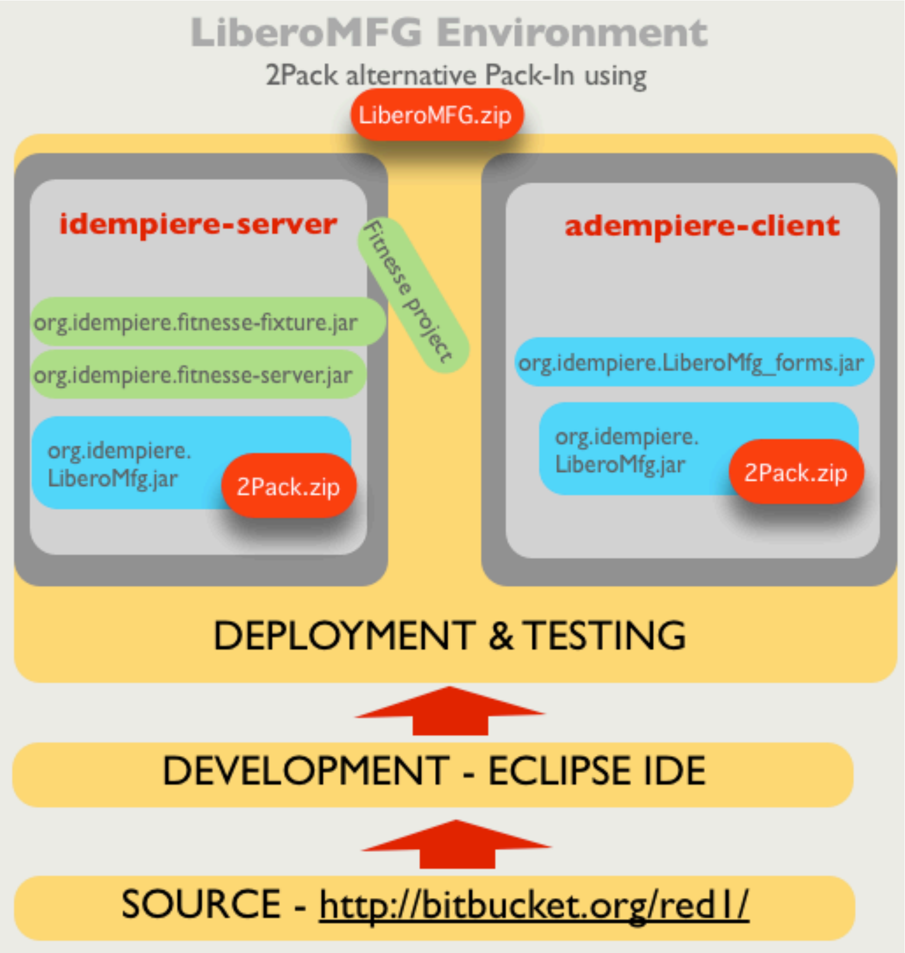
sysnova

BANGLADESH

Libero Manufacturing

Technical Guide

A comprehensive Libero Manufacturing technical guide on the conversion, upgrading, deploying and testing of LiberoMFG plugin in latest iDempiere.


LiberoMFG Environment
2Pack alternative Pack-In using
LiberoMFG.zip
idempiere-server
Fitnesse project
adempiere-client
org.idempiere.fitnesse-fixture.jar
org.idempiere.fitnesse-server.jar
org.idempiere.LiberoMfg_forms.jar
org.idempiere.
LiberoMfg.jar
2Pack.zip
org.idempiere.
LiberoMfg.jar
2Pack.zip
DEPLOYMENT & TESTING
DEVELOPMENT - ECLIPSE IDE
SOURCE - http://bitbucket.org/red1/

red1@red1.org
October 16, 2013
written in MALAYSIA

Version	Items	Dated
0.1	Full Plugin with FitNesse MakeToOrder Testing	October 16th, 2013

Item	Description	Status
Base plugin	LiberoMfg_	Deployable to both
Swing Plugin	LiberoMfg_forms	Client only
2Pack	LiberoMfg.zip	Pack-In OK both and embedded
FitNesse	org.idempiere.fitnesse.mfg	Server: fitnesse-server/fitnesse-fixture
Production Status	No full test, document or production reference	Alpha
Document	Technical guide for developers	0.1
Developer's forum	http://red1.org/adempiere/viewtopic.php?f=45&t=1775	
Tester's forum	http://red1.org/adempiere/viewtopic.php?f=45&t=1777	
SourceForge	http://sourceforge.net/projects/red1/files/p2/LiberoMFG	Binary repository
BitBucket	http://bitbucket.org/red1/	Source repository

RED1's THREE LAWS:

Information is Free

YOU HAVE TO KNOW

People are Not

YOU HAVE TO PAY

Contributors are Priceless

YOU HAVE TO BE

October 19, 2013. My capital city, Kuala Lumpur has the highest twin towers in the world and here at the SetiaSky Residences, belonging to a friend, my family get to spend some days enjoying the skyline while I put the finishing touches to this document.

TABLE OF CONTENTS

BACKGROUND

Libero Manufacturing (LiberoMFG) is another important and huge contribution from Victor Perez of e-Evolution, Mexico during the ADempiere project. Begining 2007 I supported and tried to make it mainstream in ADempiere and helped to edit the English instruction guide. Sponsored by SYSNOVA, I first reviewed and tested LiberoMFG and announced it as alpha as it destabilised the core trunk of the project. Nevertheless the global community has keen interest in such a domain area of manufacturing and thus in iDempiere, SYSNOVA again sponsored for this conversion as a free OSGi plugin, which minimise the impact it can bring to the core.

There were some code changes on Libero's part since then which has to be upgraded into my last reviewed base code. There is also the deprecation of M_Storage for OnHand and Reservation models by iDempiere. The later carries the biggest change and impact to the former Libero MFG requiring considerable class change.

Converting to OSGi plugin means the use of Extensions for Callouts, Processes, and Forms. The ModelValidator is also wrapped in an OSGi service and together with the ModelFactory registered for a Spring like performance.

I have ported into the plugin all of eevolution.model's content and this allows that package to be dropped soon from iDempiere core. However there are still some entangled classes of Libero in the core classes such as DocumentEngine that has to be solved in LiberoMFG plugin itself.

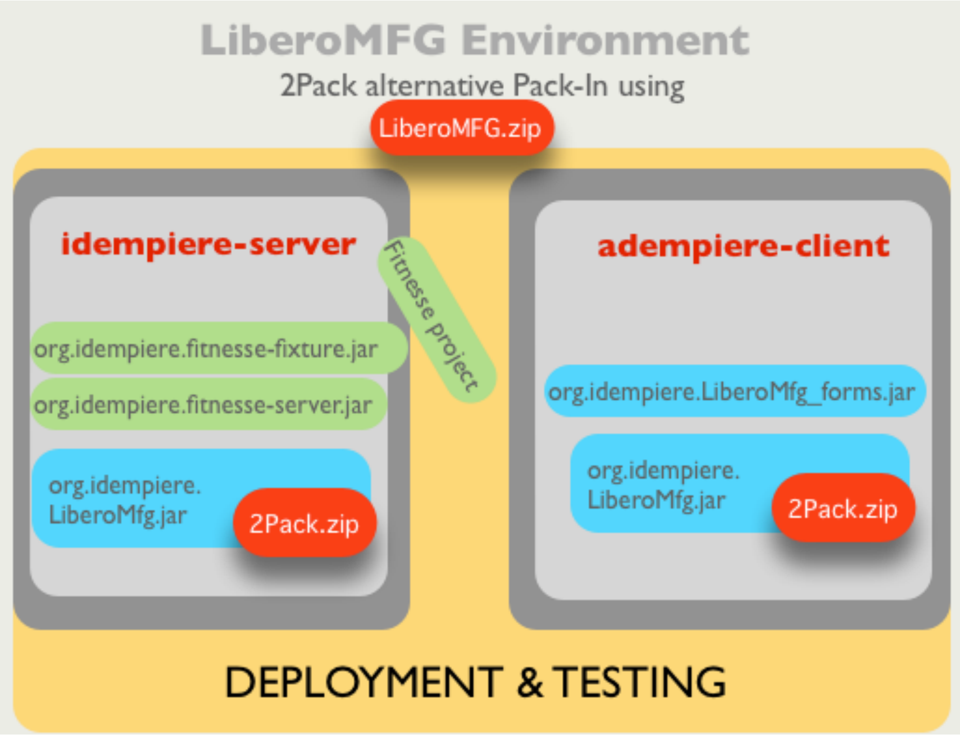
Fortunately I am able to port over the last FitNesse SLiM testing as an iDempiere plugin and confirmed that the basic functionality behaves as expected.

This guide will first detail for normal users to deploy LiberoMFG into their iDempiere instances. Then it will detail the sources of all artifacts used and how they are compiled from source for a full working environment.

Users who read and use this project should be technical proficient in software and database handling. The author will not entertain any basic question. Only competent peer review feedback is appreciated.

I hope users and developers alike can appreciate this technical expose which attempts to document as much detail the journey of this big module and collaborate further to raise it as a globally used choice for the production and manufacturing sector.

Setup



Overview of Libero Plugins Framework - Deployment and Testing side

Main Artifacts

org.idempiere.LiberoMFG.jar - base plugin that is used at both idempiere-server and adempiere-client containers.

org.idempiere.LiberoMFG_forms.jar - add-on plugin to adempiere-client container with the base plugin above.

LiberoMfg.zip - 2pack file for Pack In into iDempiere 1.0.c database. It is also embedded in the base plugin above.

Links:

http://sourceforge.net/projects/red1/files/p2/LiberoMFG/org.idempiere.LiberoMfg_1.0.0.201310152358.jar/download

sourceforge.net/projects/red1/files/p2/LiberoMFG/org.idempiere.LiberoMfg_forms_1.0.0.201310071230.jar/download

http://sourceforge.net/projects/red1/files/p2/LiberoMFG/LiberoMfg.zip/download

Testing Plugins

These are supplementary plugins for the testing part. Please read Testing section for details.

org.idempiere.fitnesse-server.jar - maps SLiM fixture management within OSGi framework.

org.idempiere.fitnesse-fixture.jar - contains reusable utilities for iDempiere model handling.

org.idempiere.fitnesse.mfg.jar - is the testing plugin that runs with the 2 plugins above.

Links: are embedded in the above jars. Just click on them when viewing this PDF in your reader.

Simple Setup

There are two ways to deploy the artifacts. Using the amazing Equinox OSGi features you can first download from the above links and then easily install into iDempiere while it is running, or you can do it without downloading but remotely updated. That is shown numerous times in my previous PDF guides. The next section 'Deploying' will tell you how to get your instances launched before you can apply the install scripts below.

Once you have the artifacts, you can launch your adempiere-client or idempiere-server. At the OSGi console do an ss to see the full stack of plugins.

Super Setup

This is the remote setup that is quite magical as it does not require you to download anything at all. Just been online and pasting the script below into your OSGi container will do the trick.

```
install http://downloads.sourceforge.net/project/red1/p2/LiberoMFG/
org.idempiere.LiberoMfg_1.0.0.201310152358.jar
```

```
install http://downloads.sourceforge.net/project/red1/p2/LiberoMFG/
org.idempiere.LiberoMfg_forms_1.0.0.201310071230.jar
```

For the two optional FitNesse plugins, use the links above to download them, unzip and install manually in your OSGi console. Then you can exceute the last optional script below.

```
install http://downloads.sourceforge.net/project/red1/p2/LiberoMFG/
org.idempiere.fitnesse.Mfg_1.0.0.201310160013.jar
```

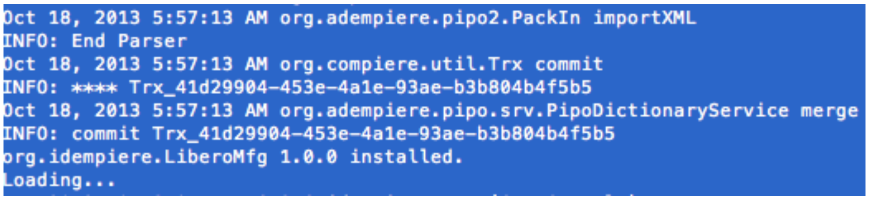
Note： these artifacts timestamps may change once they get updated. I will try to update this PDF where possible. But in the event that you are referring to an old PDF, you can check the locations of those links to get the latest timestamps and replace them in the scripts.

Deploying

Initial 2Pack Pack-In

In your latest iDempiere instance v1.0.c (as of October, the date of this publication. So if anything breaks after this please report it to me in my forum), when you installed the first plugin (LiberoMfg), it contains a 2Pack that will automatically Pack-In when started. It can work in both Swing and ZK Web UI instances. If your instance is in *Fine* logging preferences mode, you will see the prompts rushing off as seen in this testing video - http://youtu.be/YhOvEvDFoQk.

If it is not in *Fine* mode but *Info* mode those prompts will not be displayed but the instance will seem to hold for 2 minutes depending on your computer capacity to let the Pack-In process finished off. Once that is done you will see the 'installed' prompt at the end of the Pack-In logs as seen in the screen-shot below.



```
Oct 18, 2013 5:57:13 AM org.adempiere.pipo2.PackIn importXML
INFO: End Parser
Oct 18, 2013 5:57:13 AM org.compiere.util.Trx commit
INFO: **** Trx_41d29904-453e-4a1e-93ae-b3b804b4f5b5
Oct 18, 2013 5:57:13 AM org.adempiere.pipo.srv.PipoDictionaryService merge
INFO: commit Trx_41d29904-453e-4a1e-93ae-b3b804b4f5b5
org.idempiere.LiberoMfg 1.0.0 installed.
Loading...
```

In case your instance suffers a 'java.lang.OutOfMemoryError: Java heap space' use the -Xmx argument during your launch : ./adempiere-client2.sh -Xmx200m (swing client example)

In the case your iDempiere has corrupted data which you were working on and the Pack-In does not work, ensure it is the latest 1.0.c DailyBuild version by running these scripts under

https://sourceforge.net/projects/red1/files/p2/

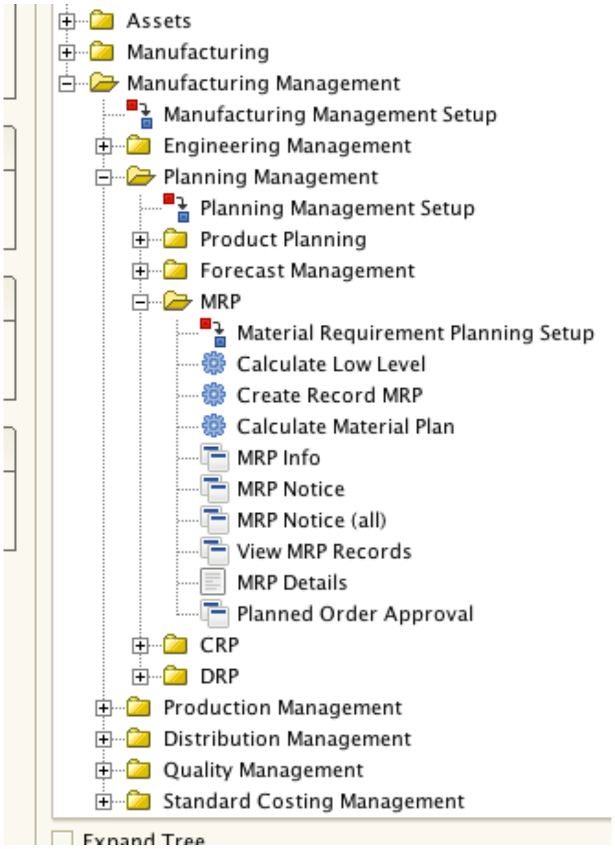
ServerUpdateFromDaily (for server instance) and ClientUpdateFromDaily (for swing instance)

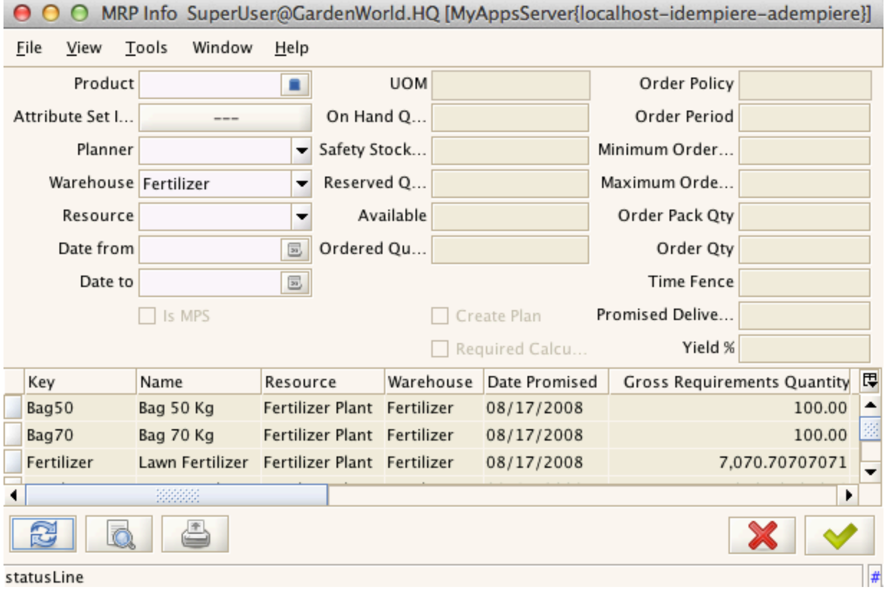
Then in the idempiere-server/utils folder, run ./RUN_ImportIdempiere to get the latest build data seed imported. (Of course you are doing all this on a trial instance and not a production instance s this will over-ride your present database).

Note: In the Developer's section, I will give full details how the 2Pack and plugin is created so that developers can stand on the shoulders of it and reach higher. Also note that if for some reason that your embedded 2Pack does not self-deploy and Pack-In into your database, you can still do so manually with the LiberoMfg.zip which is the same 2Pack. You then go to the System Menu and attach it in a new Pack-In window.

Launching Manufacturing Menu

You can then login into the *GardenAdmin* client menu to check out the new *Manufacturing Management* Menu. Click on the *Planning Management* item to expand it and you can see *MRP* which when expanded you can find an item called *MRP Info.* Click on that and it should open up a customised dialog form. Failure to open that form means the plugin is not working. If you see the screen below, then pat yourself on the back as you have arrived in the awesome world of pluggable ERP with Manufacturing functionality.





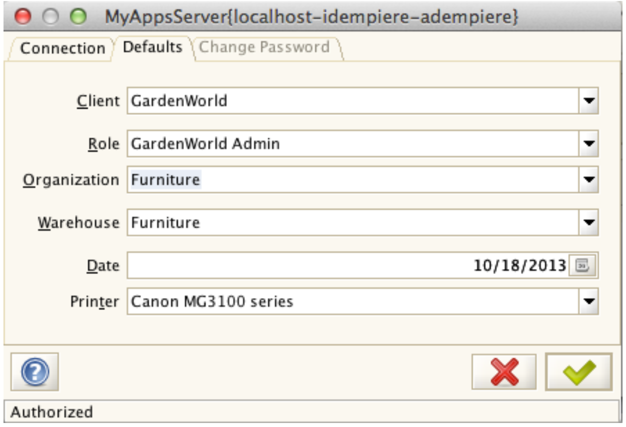
Warehouse: Fertilizer

Key	Name	Resource	Warehouse	Date Promised	Gross Requirements Quantity
Bag50	Bag 50 Kg	Fertilizer Plant	Fertilizer	08/17/2008	100.00
Bag70	Bag 70 Kg	Fertilizer Plant	Fertilizer	08/17/2008	100.00
Fertilizer	Lawn Fertilizer	Fertilizer Plant	Fertilizer	08/17/2008	7,070.70707071

Issuing Sales Order to ShopFloor

Now we jump right into the exciting part - creating a sales order for a BOM product that will trigger a production schedule to the shop-floor to produce that product on time!

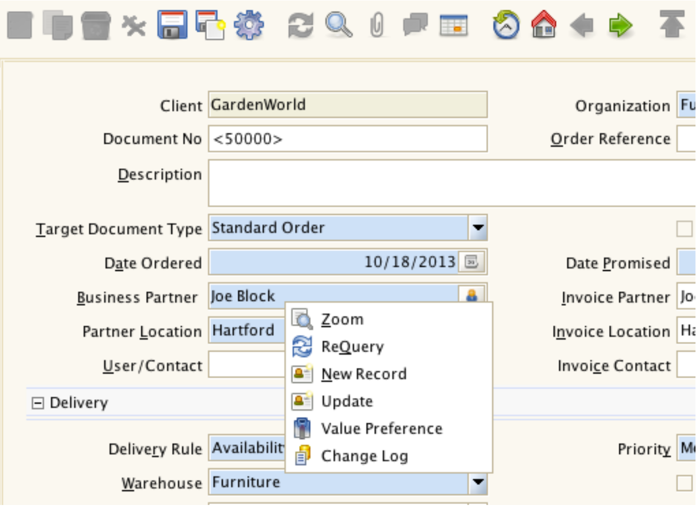
First logout and login again under Furniture Warehouse.



Client	GardenWorld
Role	GardenWorld Admin
Organization	Furniture
Warehouse	Furniture
Date	10/18/2013
Printer	Canon MG3100 series

Authorized

Then open up a new *Sales Order* and set the *DocumentType* to Standard Sales Order. Enter Joe into *Business Partner* field and press enter key. which will extract the full name and other details from the database. Then right-click on Joe Block and click on Zoom from the pop-up menu.

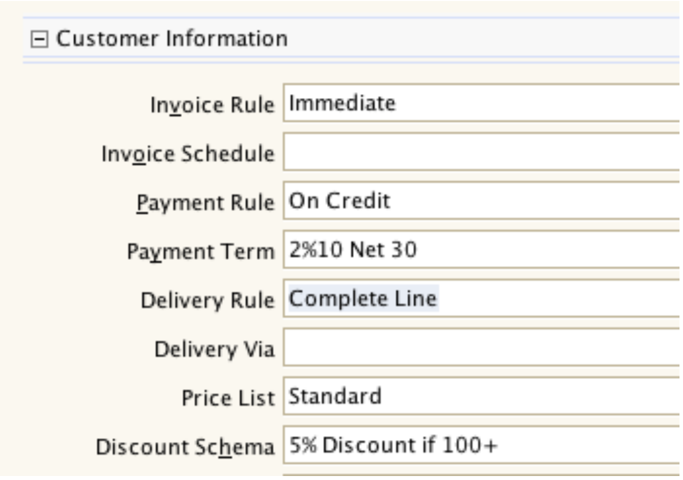


Client	GardenWorld	Organization	Fu
Document No	<50000>	Order Reference	
Description			
Target Document Type	Standard Order		□
Date Ordered	10/18/2013	Date Promised	
Business Partner	Joe Block	Invoice Partner	Jo
Partner Location	Hartford	Invoice Location	Ha
User/Contact		Invoice Contact	

⊟ Delivery

Delivery Rule	Availabilit	Priority	M
Warehouse	Furniture		□

In the opened *Business Partner* window, expand the *Customer Information* group info panel and select Complete Line for the Delivery Rule.



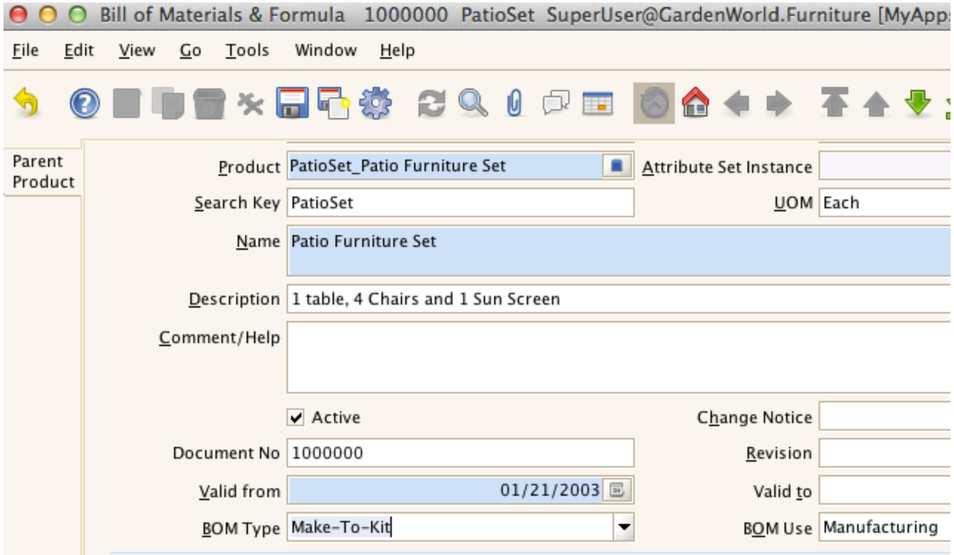
⊟ Customer Information

Invoice Rule	Immediate
Invoice Schedule	
Payment Rule	On Credit
Payment Term	2%10 Net 30
Delivery Rule	Complete Line
Delivery Via	
Price List	Standard
Discount Schema	5% Discount if 100+

Next we will set the BOM or Build of Materials information.

BOM Product Setting

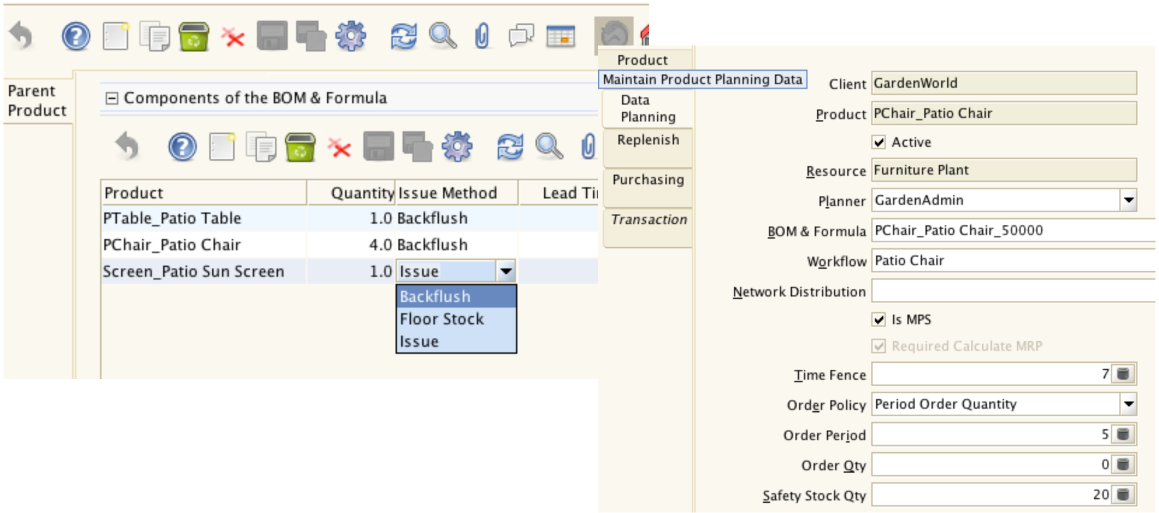
You have to set the BOM product to Make-To-Kit. You do that by calling out the Bill of Materials & Formula window, under the BomType:



Bill of Materials & Formula 1000000 PatioSet SuperUser@GardenWorld.Furniture [MyApp
File Edit View Go Tools Window Help
Parent Product

Product	PatioSet_Patio Furniture Set	Attribute Set Instance	
Search Key	PatioSet	UOM	Each
Name	Patio Furniture Set		
Description	1 table, 4 Chairs and 1 Sun Screen		
Comment/Help			
	☑ Active	Change Notice	
Document No	1000000	Revision	
Valid from	01/21/2003	Valid to	
BOM Type	Make-To-Kit	BOM Use	Manufacturing

Then at the detail tab for Components of the BOM & Formula, you set the BOM children's Issue Method to BackFlush.



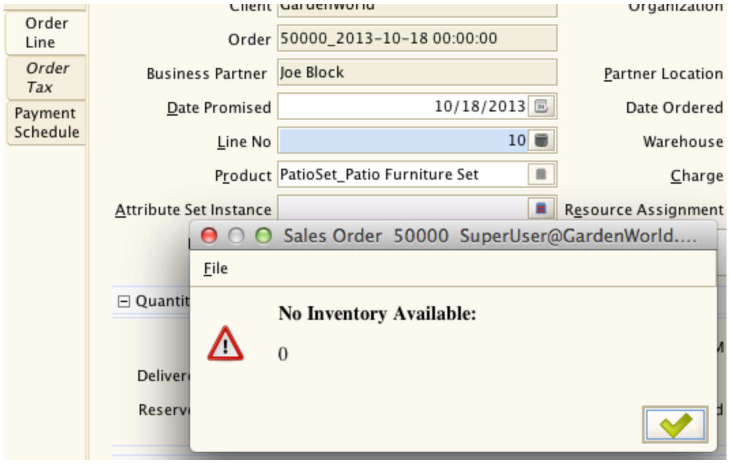
Product	Quantity	Issue Method	Lead Ti
PTable_Patio Table	1.0	Backflush	
PChair_Patio Chair	4.0	Backflush	
Screen_Patio Sun Screen	1.0	Issue	

Client	GardenWorld
Product	PChair_Patio Chair
	☑ Active
Resource	Furniture Plant
Planner	GardenAdmin
BOM & Formula	PChair_Patio Chair_50000
Workflow	Patio Chair
Network Distribution	
	☑ Is MPS
	☑ Required Calculate MRP
Time Fence	7
Order Policy	Period Order Quantity
Order Period	5
Order Qty	0
Safety Stock Qty	20

Then we go to the *Product Planning Data* window and at the second tab, *Data Planning* for Patio Chair, set its Safety Stock Qty to 20.

Now we are ready to complete the Sales Order. Go back there and at the Sales Order Line tab, get the Patio Furniture Set as the Product, save it and return to the main tab to press complete.

You will get an exception message and that is what we want as we want to produce it. We want the system to issue out production orders and generate a workflow to handle what is needed to product the product set.



Client GardenWorld

Order 50000_2013-10-18 00:00:00

Business Partner Joe Block

Date Promised 10/18/2013

Line No 10

Product PatioSet_Patio Furniture Set

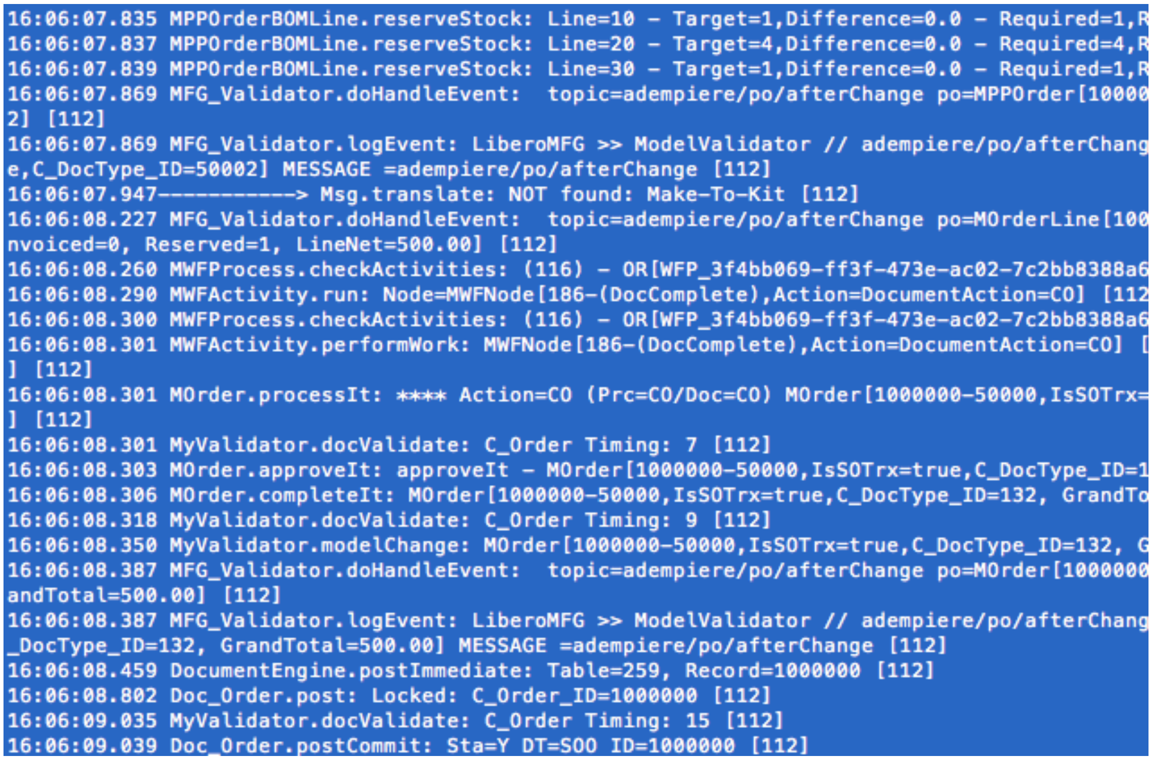
Attribute Set Instance

Sales Order 50000 SuperUser@GardenWorld....

No Inventory Available:

0

When the complete is clicked, the system will process it according to the MFG_Validator class and you can see its logging prompts in the console:

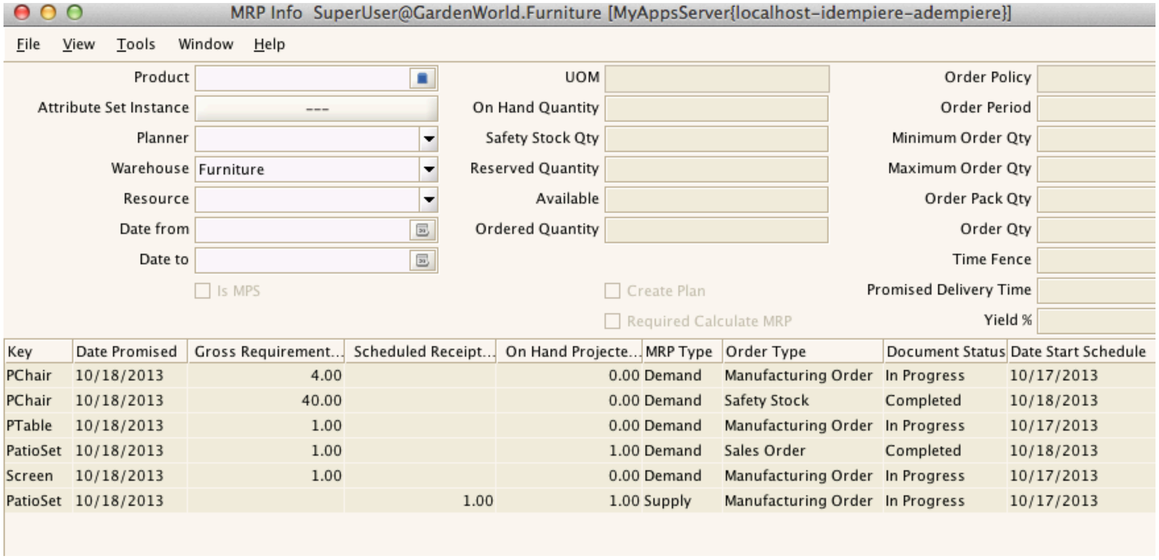


```
16:06:07.835 MPPOrderBOMLine.reserveStock: Line=10 - Target=1,Difference=0.0 - Required=1,R
16:06:07.837 MPPOrderBOMLine.reserveStock: Line=20 - Target=4,Difference=0.0 - Required=4,R
16:06:07.839 MPPOrderBOMLine.reserveStock: Line=30 - Target=1,Difference=0.0 - Required=1,R
16:06:07.869 MFG_Validator.doHandleEvent:  topic=adempiere/po/afterChange po=MPPOrder[10000
2] [112]
16:06:07.869 MFG_Validator.logEvent: LiberoMFG >> ModelValidator // adempiere/po/afterChang
e,C_DocType_ID=50002] MESSAGE =adempiere/po/afterChange [112]
16:06:07.947-----------> Msg.translate: NOT found: Make-To-Kit [112]
16:06:08.227 MFG_Validator.doHandleEvent:  topic=adempiere/po/afterChange po=MOrderLine[100
nvoiced=0, Reserved=1, LineNet=500.00] [112]
16:06:08.260 MWFProcess.checkActivities: (116) - OR[WFP_3f4bb069-ff3f-473e-ac02-7c2bb8388a6
16:06:08.290 MWFActivity.run: Node=MWFNode[186-(DocComplete),Action=DocumentAction=CO] [112
16:06:08.300 MWFProcess.checkActivities: (116) - OR[WFP_3f4bb069-ff3f-473e-ac02-7c2bb8388a6
16:06:08.301 MWFActivity.performWork: MWFNode[186-(DocComplete),Action=DocumentAction=CO] [
] [112]
16:06:08.301 MOrder.processIt: **** Action=CO (Prc=CO/Doc=CO) MOrder[1000000-50000,IsSOTrx=
] [112]
16:06:08.301 MyValidator.docValidate: C_Order Timing: 7 [112]
16:06:08.303 MOrder.approveIt: approveIt - MOrder[1000000-50000,IsSOTrx=true,C_DocType_ID=1
16:06:08.306 MOrder.completeIt: MOrder[1000000-50000,IsSOTrx=true,C_DocType_ID=132, GrandTo
16:06:08.318 MyValidator.docValidate: C_Order Timing: 9 [112]
16:06:08.350 MyValidator.modelChange: MOrder[1000000-50000,IsSOTrx=true,C_DocType_ID=132, G
16:06:08.387 MFG_Validator.doHandleEvent:  topic=adempiere/po/afterChange po=MOrder[1000000
andTotal=500.00] [112]
16:06:08.387 MFG_Validator.logEvent: LiberoMFG >> ModelValidator // adempiere/po/afterChang
_DocType_ID=132, GrandTotal=500.00] MESSAGE =adempiere/po/afterChange [112]
16:06:08.459 DocumentEngine.postImmediate: Table=259, Record=1000000 [112]
16:06:08.802 Doc_Order.post: Locked: C_Order_ID=1000000 [112]
16:06:09.035 MyValidator.docValidate: C_Order Timing: 15 [112]
16:06:09.039 Doc_Order.postCommit: Sta=Y DT=SOO ID=1000000 [112]
```

You can see the top of the screen above about the MPPOrderBOMLine class doing some reserveStock and further below are few MFG_Validator lines handling the caught events. More details on what is caught in the validator at the Developer's section.

Do note that all this can happen because we are in the GardenWorld sample client where most of the pre-requisite information are preset and configured. Thus it is good to go through them in finer detail when learning how to setup in a new client.

From here we will examine what has happened. We go to the MRP Info window to take a look as shown below:


MRP Info SuperUser@GardenWorld.Furniture [MyAppsServer{localhost-idempiere-adempiere}]
File
View
Tools
Window
Help

Product
Attribute Set Instance ---
Planner
Warehouse Furniture
Resource
Date from
Date to
□ Is MPS

UOM
On Hand Quantity
Safety Stock Qty
Reserved Quantity
Available
Ordered Quantity
□ Create Plan
□ Required Calculate MRP

Order Policy
Order Period
Minimum Order Qty
Maximum Order Qty
Order Pack Qty
Order Qty
Time Fence
Promised Delivery Time
Yield %

Key	Date Promised	Gross Requirement...	Scheduled Receipt...	On Hand Projecte...	MRP Type	Order Type	Document Status	Date Start Schedule
PChair	10/18/2013	4.00		0.00	Demand	Manufacturing Order	In Progress	10/17/2013
PChair	10/18/2013	40.00		0.00	Demand	Safety Stock	Completed	10/18/2013
PTable	10/18/2013	1.00		0.00	Demand	Manufacturing Order	In Progress	10/17/2013
PatioSet	10/18/2013	1.00		1.00	Demand	Sales Order	Completed	10/18/2013
Screen	10/18/2013	1.00		0.00	Demand	Manufacturing Order	In Progress	10/17/2013
PatioSet	10/18/2013		1.00	1.00	Supply	Manufacturing Order	In Progress	10/17/2013

You can see the new very interesting information up there. It seems that this factory is really working after all! There is a listing of items to be ordered and what OrderType it is about. There is classification if it is a *Demand* or *Supply* MRPType. The work StartSchedule is also stated together with the stated requirements and projections of items and their quantities.

You can try following the ADempiere wiki to walk through the other functionalities here http://www.adempiere.com/A_Step_by_Step_Guide_to_Libero_Manufacturing.

Please report back in my forum (link on top pages index) and I can quickly resolve it and commit the fixes.

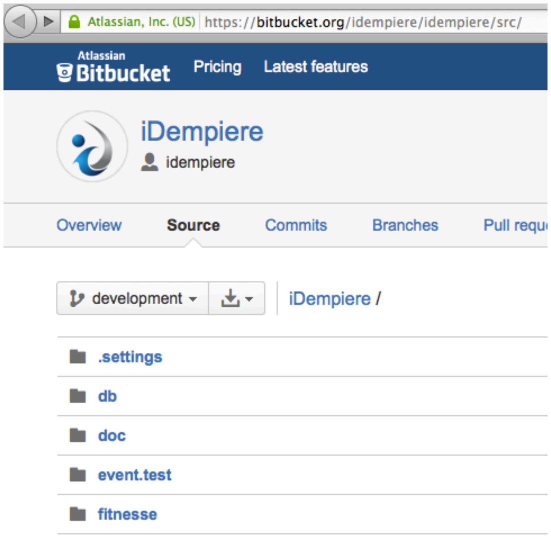
Testing

Using Fitnesse

Testing is not only mandatory but also crucial in debugging and developing further any software. FitNesse is a framework for testing in a more visible and user-friendly manner. The nice thing about iDempiere is that it has FitNesse testing integrated into it in the form of plugins. Thus the goodness of loosely coupling happens there. You need not bundle your iDempiere with extra plugins but can easily activate them when needed.

From the top section on Setup, you should be able to install the 3 plugins for FitNesse testing of the LiberoMFG.

Once they are started and active you look for a separate project called 'fitnesse' inside your iDempiere source from bitbucket.

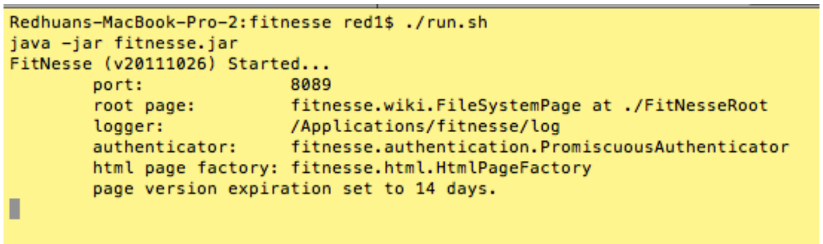


If not or if you are confused as I was, go to http://bitbucket.org/idempiere/idempiere/src and click on **fitnesse** there to view the source tree or if you want to check it out with

hg clone https://bitbucket.org/idempiere/idempiere

Put this project into another directory (in my Macbook case it is /Applications/ to run it.

Ensure you have idempiereEnv.properties placed at your home folder. If not, copy it from your <IDEMPIERE-HOME>. However this is not mandatory as the FitNesse still works. You should see the following in your console or terminal window that executes the run batch or shell file.

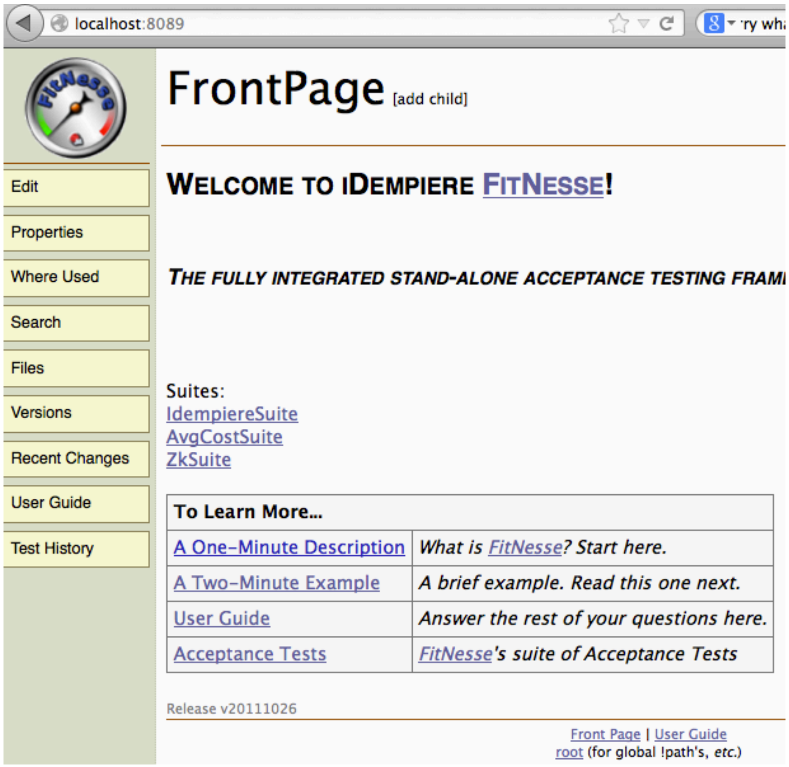


```
Redhuans-MacBook-Pro-2:fitnesse red1$ ./run.sh
java -jar fitnesse.jar
FitNesse (v20111026) Started...
        port:              8089
        root page:         fitnesse.wiki.FileSystemPage at ./FitNesseRoot
        logger:            /Applications/fitnesse/log
        authenticator:     fitnesse.authentication.PromiscuousAuthenticator
        html page factory: fitnesse.html.HtmlPageFactory
        page version expiration set to 14 days.
```

Then note the port number shown in the console and use that in the browser: http://localhost:8089.

You will see the Front page as displayed here on the right.

Click on the bottom root link and paste the following into it:



FrontPage [add child]

WELCOME TO IDEMPIERE FITNESSE!

THE FULLY INTEGRATED STAND-ALONE ACCEPTANCE TESTING FRAMI

Suites:
IdempiereSuite
AvgCostSuite
ZkSuite

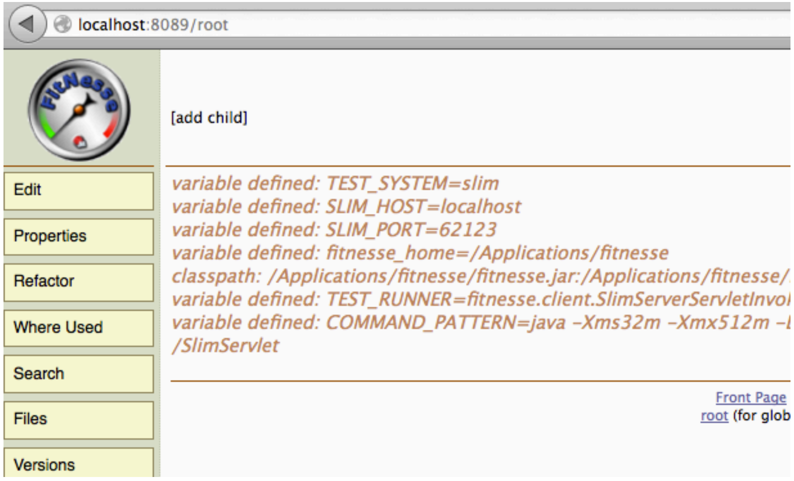
To Learn More...	
A One-Minute Description	*What is FitNesse? Start here.*
A Two-Minute Example	*A brief example. Read this one next.*
User Guide	*Answer the rest of your questions here.*
Acceptance Tests	*FitNesse's suite of Acceptance Tests*

Release v20111026

Front Page | User Guide
root (for global !path's, *etc.*)

```
!define TEST_SYSTEM {slim}
!define SLIM_HOST (localhost)
!define SLIM_PORT (62123)
!define fitnesse_home {/Applications/fitnesse}
!path ${fitnesse_home}/fitnesse.jar:${fitnesse_home}/lib/*.jar:${fitnesse_home}/bin
!define TEST_RUNNER {fitnesse.client.SlimServerServletInvoker}
!define COMMAND_PATTERN {java -Xms32m -Xmx512m -DLOG4J_LEVEL=CONFIG -cp %p %m http://localhost:8080/fitnesse/SlimServlet}
```

Edit it to fit your own environment, then save it and you should get:



[add child]

```
variable defined: TEST_SYSTEM=slim
variable defined: SLIM_HOST=localhost
variable defined: SLIM_PORT=62123
variable defined: fitnesse_home=/Applications/fitnesse
classpath: /Applications/fitnesse/fitnesse.jar:/Applications/fitnesse/
variable defined: TEST_RUNNER=fitnesse.client.SlimServerServletInvo
variable defined: COMMAND_PATTERN=java -Xms32m -Xmx512m -
/SlimServlet
```

Front Page
root (for glob

Test Script

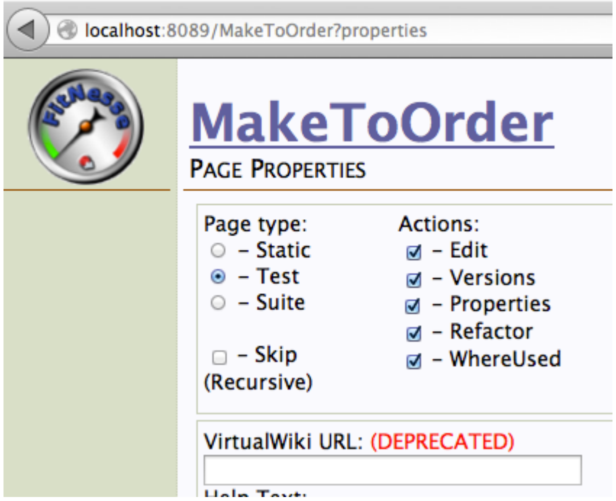
Return to the Front page and click on Add Child. Give it a name say “MakeToOrder” and make sure the properties is set to Test. OK it to see its empty page, press Edit and paste the following into its empty space (you can copy from http://red1.org/adempiere/viewtopic.php?f=45&t=1777#p8481. However it may give you extra blanks on the left side, so just copying the text below should be better.) :

```
!|ADempiere Login|
|User|Password|Client|Role|execute?|
|GardenAdmin|GardenAdmin|GardenWorld|GardenWorld Admin|true|

!|User Setup|
|Business Partner|Organisation|Warehouse|Product|Forward Backward|Ac-
counting Schema|Costing Method|Master Data?|Accounts Posting?|
|Joe Block | Furniture|Furniture | PatioSet |F|GardenWorld US/A/US Dol-
lar |Standard |true |I|

!|User Setup|
|PatioSet Bom Type|Chair Safety Stock|Issue Method|Customer Delivery
Rule|
|Make To Kit|20|Back Flush|Complete Line|

Check Storage
!|Manufacturing Steps For|
|Product|Storage On Hand?|Storage Reserved?|Available Storage?|
|PChair|30|0|30|
|PTable|12|0|12|
|Screen|15|0|15|
|PatioSet|0|0|0|

ISSUE SALES ORDER FOR A BOM - PATIO CHAIR
!|Manufacturing Steps For|
|Promise Days|Qty|Price|Transaction Type|Activity Measure|Activity Du-
ration|Execute Transaction?|
|10|1|500|Sales Order|Day|3|true|

!|Manufacturing Steps For|
|Product|Storage On Hand?|Storage Reserved?|Available Storage?|
|PChair|30 |0 |30|
|PTable|12 |0 |12|
|Screen|15 |0 |15|
|PatioSet|0 |1 |-1|

!|Query:Check MRP|
|Name|PP_MRP_ID|Doc Status|Order Type|Warehouse|Product|Qty|Start
Schedule|Finish Schedule|OrderBomLine|PP_Order|
|MRP||CO|SOO|||||||
|80000||IP|MOP|||||||
|80000||IP|MOP|||||||
|80000||IP|MOP|||||||
|80000||IP|MOP|||||||
!|Process Handling|
|Calculate Material Plan?|
|true|

!|Query:Check Manufacturing Order|
|Document No|Product|Doc Status|Qty Ordered|Qty Reserved|Qty Delivered|Date Promised|Date Start Schedule|Date Finish Schedule|
|80000|MProduct[145-PatioSet]|IP|1|1|0||||

!|Query:Check MRP|
|Name|PP_MRP_ID|Doc Status|Order Type|Warehouse|Product|Qty|Start Schedule|Finish Schedule|OrderBomLine|PP_Order|
|MRP||CO|||||||||
|80000||IP|||||||||
|80000||IP|||||||||
|80000||IP|||||||||
|80000||IP|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||

!|Query:Check Requisition|
|Product|Qty|Price|Org ID|Doc Status|
|MProduct[133-PChair]|||||
|MProduct[134-PTable]|||||
|MProduct[135-Screen]|||||
|MProduct[133-PChair]|||||

!|Query:Check Manufacturing Order BOM|
|Name|Document No|Product|Mfg Order|
|Patio Furniture Set||||

!|Query:Check Manufacturing Order Node|
|Name|Doc Status|Node|Workflow|Qty Delivered|Setup Time|
|Assembly|DR|||||

!|Query:Check Manufacturing Order WorkFlow|
|Name|Workflow|Resource|
|Patio Furniture Set|||

!|Process Handling|
|Calculate Requirements Plan?|
|true|

!|Query:Check Manufacturing Order|
|Document No|Product|Doc Status|Qty Ordered|Qty Reserved|Qty Delivered|Date Promised|Date Start Schedule|Date Finish Schedule|OrderLine|
|80000|MProduct[145-PatioSet]|IP|1|1|0|||||

!|Manufacturing Steps For|
|Product|Qty|Transaction Type|Location From|Location To|Execute Transaction?|
|PChair|29|Inventory Movement|Default HQ Locator|Furniture|true|
|PTable|2|Inventory Movement|Default HQ Locator|Furniture|true|
|Screen|2|Inventory Movement|Default HQ Locator|Furniture|true|

!|Manufacturing Steps For|
|Transaction Type|Execute Transaction?|Get Account?|
|Create Issue|true  |true|

!|Query:Check Cost Collector|
|Product |Type |Doc Status |Locator |Serial No |Movement Date |Finish
|Movement Qty|
|MProduct[145-PatioSet] |100 |CO |Furniture |   ||               |1|
|MProduct[134-PTable] |110 |CO |Furniture |X_M_AttributeSetInstance[0]
|||1|
|MProduct[133-PChair] |110 |CO |Furniture |X_M_AttributeSetInstance[0]
|||4|
|MProduct[135-Screen] |110 |CO |Furniture |X_M_AttributeSetInstance[0]
|||1|
|MProduct[134-PTable] |140 |CO |Furniture |X_M_AttributeSetInstance[0]
|||1|
|MProduct[133-PChair] |140 |CO |Furniture |X_M_AttributeSetInstance[0]
|||4|
|MProduct[135-Screen] |140 |CO |Furniture |X_M_AttributeSetInstance[0]
|||1|

!|Manufacturing Steps For|
|Product|Storage On Hand?|Storage Reserved?|Available Storage?|
|PChair|26||26|
|PTable|11||11|
|Screen|14||14|
|PatioSet|1|1||

!|Manufacturing Steps For|
|Transaction Type|Action|Execute Transaction?|
|Process Mfg Order|Complete|true  |

!|Manufacturing Steps For|
|Product|Storage On Hand?|Storage Reserved?|Available Storage?|
|PChair|26||26|
|PTable|11||11|
|Screen|14||14|
|PatioSet|1|1||

!|Query:Check Manufacturing Order|
|Document No|Product|Doc Status|Qty Ordered|Qty Reserved|Qty Deliv-
ered|Date Promised|Date Start Schedule|Date Finish Schedule|OrderLine|
|80000|MProduct[145-PatioSet]|CO|1|0|1|||||

!|Manufacturing Steps For|
|Transaction Type|Action|Execute Transaction?|
|Process Mfg Order|Close|true|

!|Query:Check Manufacturing Order|
|Document No|Product|Doc Status|Qty Ordered|Qty Reserved|Qty Deliv-
ered|Date Promised|Date Start Schedule|Date Finish Schedule|OrderLine|
|80000|MProduct[145-PatioSet]|CL|1|0|1|||||

!|Query:Check Manufacturing Order Node|
|Name|Doc Status|Node|Workflow|Qty Delivered|Setup Time|
|Assembly|CL|||||

!|Query:Check MRP|
|Name|PP_MRP_ID|Doc Status|Order Type|Warehouse|Product|Qty|Start
Schedule|Finish Schedule|OrderBomLine|PP_Order|
|MRP||CO|||||||||
|80000||CL|||||||||
|80000||CL|||||||||
|80000||CL|||||||||
|80000||CL|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||
|MRP||DR|||||||||

!|Manufacturing Steps For|
|Product|Storage On Hand?|Storage Reserved?|Available Storage?|
|PChair  |26||26|
|PTable |11||11|
|Screen |14||14|
|PatioSet |1|1||

!|Query:Check Notes|
|Reference|Note ID|Message|Other Info|
|PChair Patio Chair||||
|PChair Patio Chair||||
|PatioSet Patio Furniture Set||||

!|User Setup|
|Roll Back?|
|true|
```

After pasting, and if there are errors during testing, you may have to edit or make sure that there are no trailing spaces before and after a value in each column.

Launch Test

After saving the page, check that the test button on the left panel does appear. If not, then click on the properties link and set it to test:



localhost:8089/MakeToOrder?properties

MakeToOrder

PAGE PROPERTIES

Page type:
○ – Static
● – Test
○ – Suite

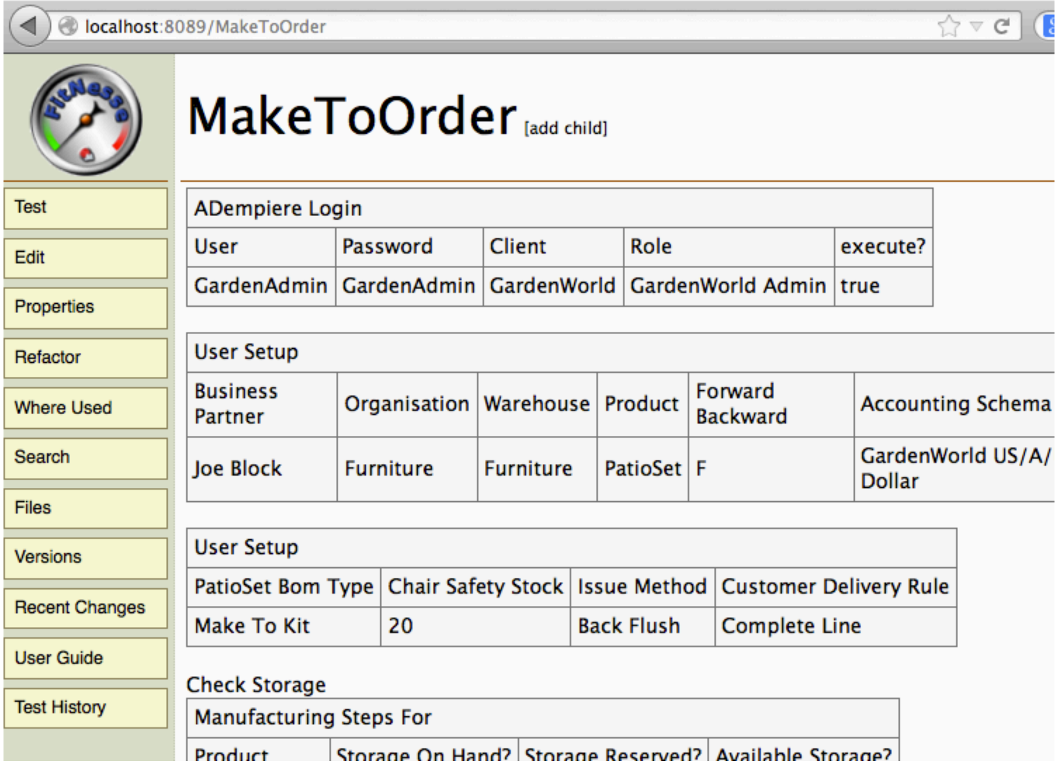
□ – Skip (Recursive)

Actions:
☑ – Edit
☑ – Versions
☑ – Properties
☑ – Refactor
☑ – WhereUsed

VirtualWiki URL: (DEPRECATED)

Help Text:

Save it and you should see the page displayed below.



localhost:8089/MakeToOrder

MakeToOrder [add child]

ADempiere Login				
User	Password	Client	Role	execute?
GardenAdmin	GardenAdmin	GardenWorld	GardenWorld Admin	true

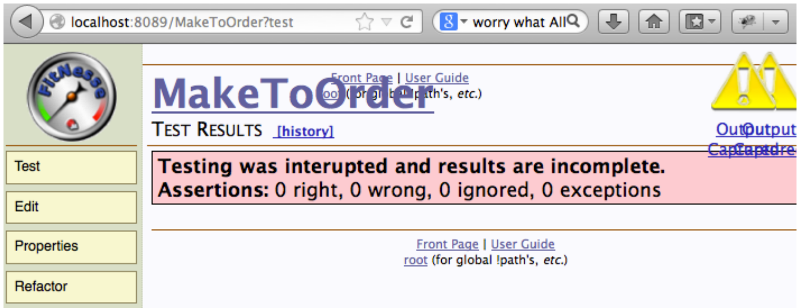
User Setup					
Business Partner	Organisation	Warehouse	Product	Forward Backward	Accounting Schema
Joe Block	Furniture	Furniture	PatioSet	F	GardenWorld US/A/ Dollar

User Setup			
PatioSet Bom Type	Chair Safety Stock	Issue Method	Customer Delivery Rule
Make To Kit	20	Back Flush	Complete Line

Check Storage

Manufacturing Steps For			
Product	Storage On Hand?	Storage Reserved?	Available Storage?

Now you can run it simply by clicking on the Test button. You should have the 3 *fitnesse* plugins already running at this time or else you can hit the following screen:



localhost:8089/MakeToOrder?test

MakeToOrder

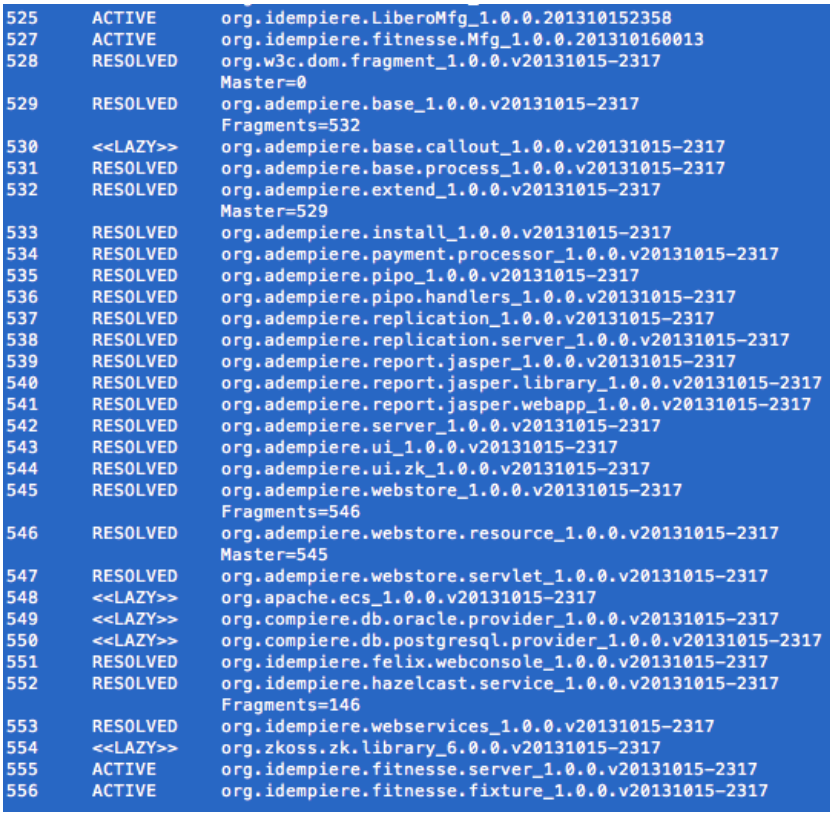
TEST RESULTS [history]

Testing was interupted and results are incomplete.
Assertions: 0 right, 0 wrong, 0 ignored, 0 exceptions

Front Page | User Guide
root (for global !path's, *etc.*)

So launch your idempiere-server with ./idempiere-server.sh -Xmx800m to provide enough memory and avoid Java Heap Size error during Pack-In. Alternatively you can use the adempiere-client to do Pack-In as shown early on. Ensure your plugins are working as shown on the screen-shot below. Note that bundle 527, 555 and 556 are active. Otherwise just type in start 555 and then start 556 527 (actual bundle numbers depend on your console display).

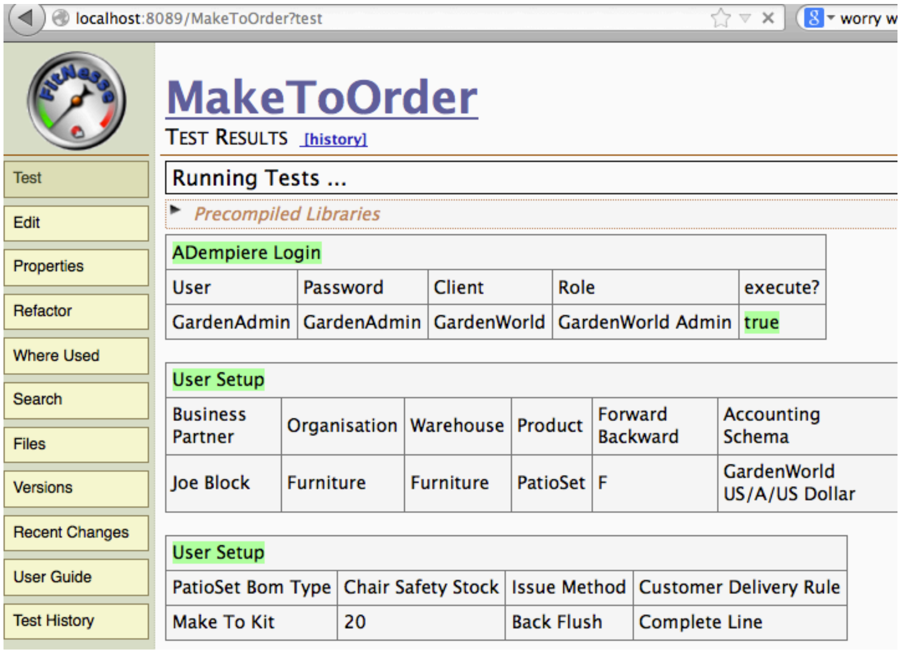
Note also at the top of the screen that LiberoMfg bundle number 525 is active and the 2pack has fully deployed its Pack-In process.



```
525     ACTIVE      org.idempiere.LiberoMfg_1.0.0.201310152358
527     ACTIVE      org.idempiere.fitnesse.Mfg_1.0.0.201310160013
528     RESOLVED    org.w3c.dom.fragment_1.0.0.v20131015-2317
                    Master=0
529     RESOLVED    org.adempiere.base_1.0.0.v20131015-2317
                    Fragments=532
530     <<LAZY>>    org.adempiere.base.callout_1.0.0.v20131015-2317
531     RESOLVED    org.adempiere.base.process_1.0.0.v20131015-2317
532     RESOLVED    org.adempiere.extend_1.0.0.v20131015-2317
                    Master=529
533     RESOLVED    org.adempiere.install_1.0.0.v20131015-2317
534     RESOLVED    org.adempiere.payment.processor_1.0.0.v20131015-2317
535     RESOLVED    org.adempiere.pipo_1.0.0.v20131015-2317
536     RESOLVED    org.adempiere.pipo.handlers_1.0.0.v20131015-2317
537     RESOLVED    org.adempiere.replication_1.0.0.v20131015-2317
538     RESOLVED    org.adempiere.replication.server_1.0.0.v20131015-2317
539     RESOLVED    org.adempiere.report.jasper_1.0.0.v20131015-2317
540     RESOLVED    org.adempiere.report.jasper.library_1.0.0.v20131015-2317
541     RESOLVED    org.adempiere.report.jasper.webapp_1.0.0.v20131015-2317
542     RESOLVED    org.adempiere.server_1.0.0.v20131015-2317
543     RESOLVED    org.adempiere.ui_1.0.0.v20131015-2317
544     RESOLVED    org.adempiere.ui.zk_1.0.0.v20131015-2317
545     RESOLVED    org.adempiere.webstore_1.0.0.v20131015-2317
                    Fragments=546
546     RESOLVED    org.adempiere.webstore.resource_1.0.0.v20131015-2317
                    Master=545
547     RESOLVED    org.adempiere.webstore.servlet_1.0.0.v20131015-2317
548     <<LAZY>>    org.apache.ecs_1.0.0.v20131015-2317
549     <<LAZY>>    org.compiere.db.oracle.provider_1.0.0.v20131015-2317
550     <<LAZY>>    org.compiere.db.postgresql.provider_1.0.0.v20131015-2317
551     RESOLVED    org.idempiere.felix.webconsole_1.0.0.v20131015-2317
552     RESOLVED    org.idempiere.hazelcast.service_1.0.0.v20131015-2317
                    Fragments=146
553     RESOLVED    org.idempiere.webservices_1.0.0.v20131015-2317
554     <<LAZY>>    org.zkoss.zk.library_6.0.0.v20131015-2317
555     ACTIVE      org.idempiere.fitnesse.server_1.0.0.v20131015-2317
556     ACTIVE      org.idempiere.fitnesse.fixture_1.0.0.v20131015-2317
```

Now pressing the test button should give the following in-progress screen:

After some furious minutes of behind the scenes processing you should end up with flying colours, shown below:



MakeToOrder

TEST RESULTS [history]

Running Tests ...

▸ Precompiled Libraries

ADempiere Login

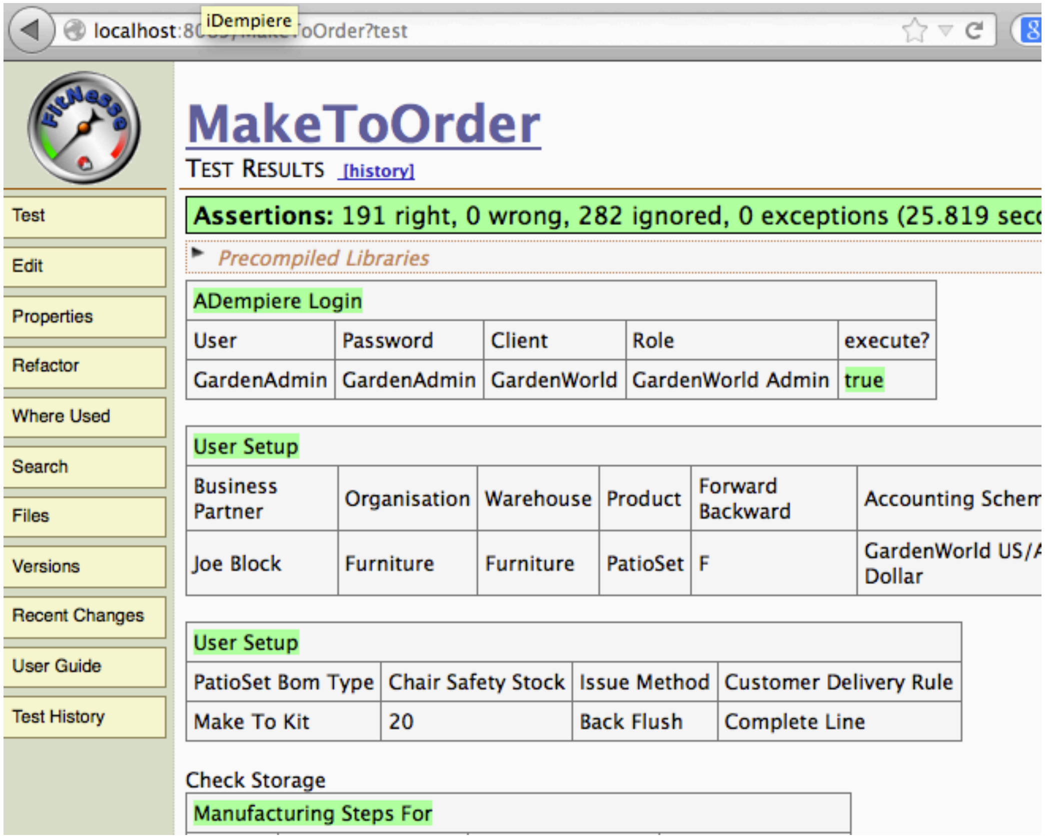
User	Password	Client	Role	execute?
GardenAdmin	GardenAdmin	GardenWorld	GardenWorld Admin	true

User Setup

Business Partner	Organisation	Warehouse	Product	Forward Backward	Accounting Schema
Joe Block	Furniture	Furniture	PatioSet	F	GardenWorld US/A/US Dollar

User Setup

PatioSet Bom Type	Chair Safety Stock	Issue Method	Customer Delivery Rule
Make To Kit	20	Back Flush	Complete Line



MakeToOrder

TEST RESULTS [history]

Assertions: 191 right, 0 wrong, 282 ignored, 0 exceptions (25.819 sec

Precompiled Libraries

ADempiere Login				
User	Password	Client	Role	execute?
GardenAdmin	GardenAdmin	GardenWorld	GardenWorld Admin	true

User Setup					
Business Partner	Organisation	Warehouse	Product	Forward Backward	Accounting Schem
Joe Block	Furniture	Furniture	PatioSet	F	GardenWorld US/A Dollar

User Setup			
PatioSet Bom Type	Chair Safety Stock	Issue Method	Customer Delivery Rule
Make To Kit	20	Back Flush	Complete Line

Check Storage

Manufacturing Steps For

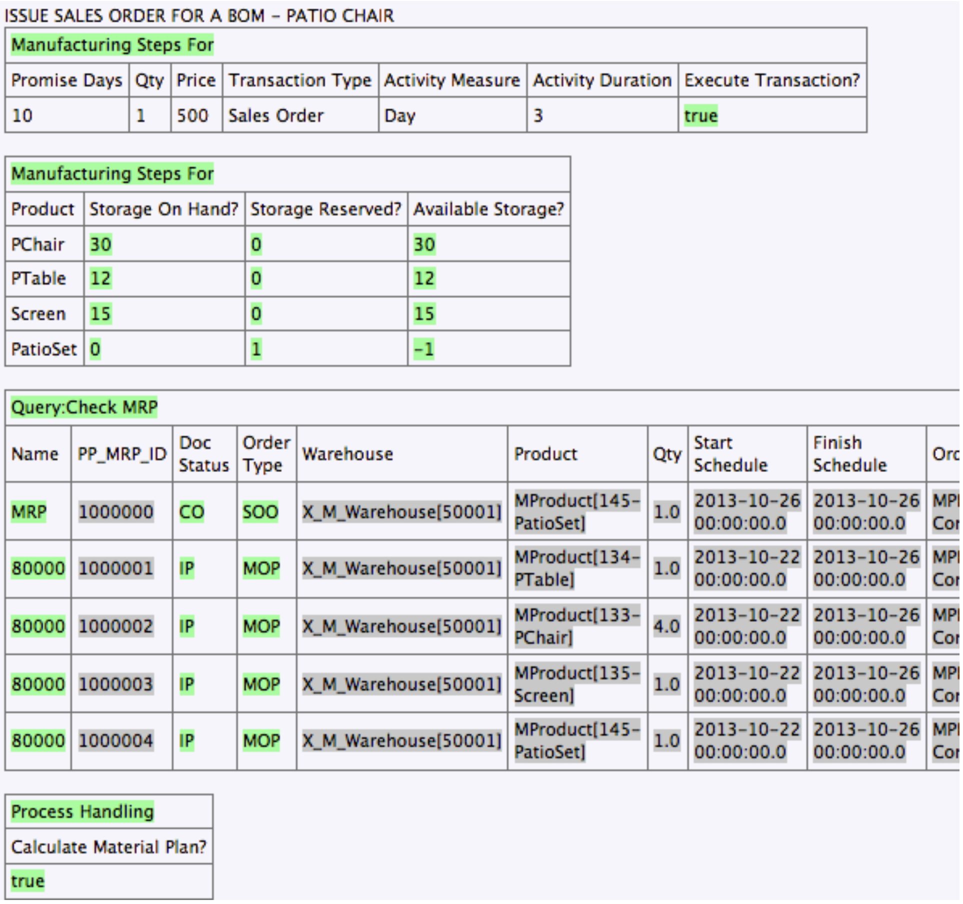
There are two videos I made to show how I set things up and carry out those tests:

http://www.youtube.com/watch?v=CtV9wS1C3rY

http://www.youtube.com/watch?v=YhOvEvDFoQk

New Storage Management

As noted, what really breaks the old ADempiere is in the new iDempiere way to manage Storage counts particularly as M_StorageOnHand and M_StorageReservation. Previously they were all managed in a single table M_Storage. Today breaking them into these two allow for accurate global use across different Warehouse Locators and scenarios. So likewise in my tests as in the code, much refactoring has been done. I also thought of showing this differentiation in the FitNesse tests themselves. I have made the extra columns to pick from those two tables which are used to work out the nett in between them. You can see this at parts along the FitNesse results:



ISSUE SALES ORDER FOR A BOM - PATIO CHAIR

Manufacturing Steps For

Promise Days	Qty	Price	Transaction Type	Activity Measure	Activity Duration	Execute Transaction?
10	1	500	Sales Order	Day	3	true

Manufacturing Steps For

Product	Storage On Hand?	Storage Reserved?	Available Storage?
PChair	30	0	30
PTable	12	0	12
Screen	15	0	15
PatioSet	0	1	-1

Query:Check MRP

Name	PP_MRP_ID	Doc Status	Order Type	Warehouse	Product	Qty	Start Schedule	Finish Schedule	Orc
MRP	1000000	CO	SOO	X_M_Warehouse[50001]	MProduct[145-PatioSet]	1.0	2013-10-26 00:00:00.0	2013-10-26 00:00:00.0	MPI Cor
80000	1000001	IP	MOP	X_M_Warehouse[50001]	MProduct[134-PTable]	1.0	2013-10-22 00:00:00.0	2013-10-26 00:00:00.0	MPI Cor
80000	1000002	IP	MOP	X_M_Warehouse[50001]	MProduct[133-PChair]	4.0	2013-10-22 00:00:00.0	2013-10-26 00:00:00.0	MPI Cor
80000	1000003	IP	MOP	X_M_Warehouse[50001]	MProduct[135-Screen]	1.0	2013-10-22 00:00:00.0	2013-10-26 00:00:00.0	MPI Cor
80000	1000004	IP	MOP	X_M_Warehouse[50001]	MProduct[145-PatioSet]	1.0	2013-10-22 00:00:00.0	2013-10-26 00:00:00.0	MPI Cor

Process Handling

Calculate Material Plan?
true

You can read further from the http://bitbucket.org/red1/org.idempiere.liberomfg source changes by looking at commits remarked with Storage or getAvailableQty comments

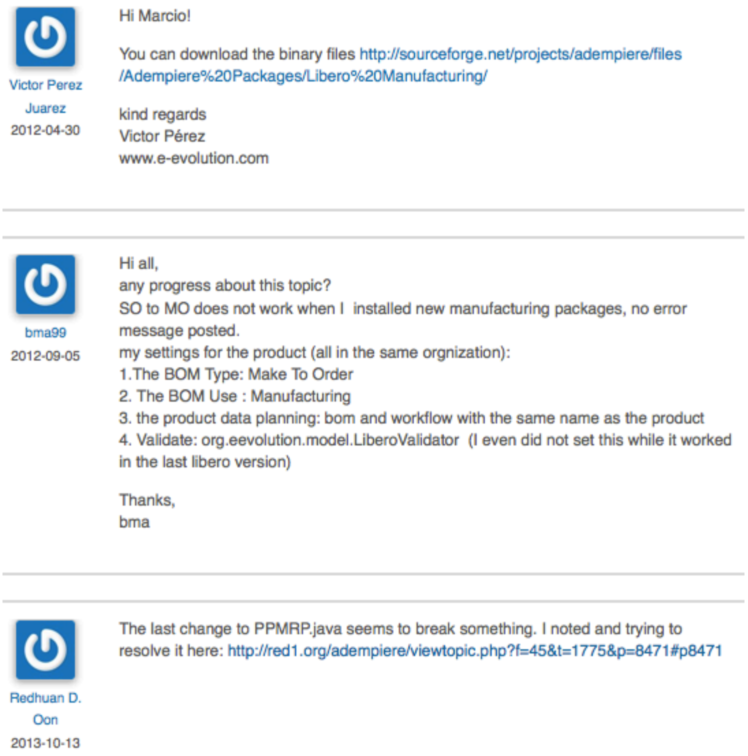
Bugs!

It is with these tests that I could discover bugs more easily and readily as it dispense off data preparation time and get to the code in question quite instantly. In the FitNesse which is not unlike how a developer will do it in JUnit testing, one can quickly focus on changing variables and code logic to look deeper into an occurring bug, quickly establishing the correct logic, before forgetting about it in the maze.

This is where I like to expose a bug that I think is solved, but it may also mean that my version of LiberoMFG is a forked one from the core project by Victor Perez of eEvolution! Thus it is important that this is made clear and subsequent developers do not reinvent the wheel and collaborate better on this fork. Of course I do not claim exclusivity other than on merit and I welcome if someone improves it without my constant participation and thus a better fork. Also I welcome that fork to announce itself to me for further review as I announce things always in a transparent and free manner.

The bug in question is at the bottom of this thread:
http://sourceforge.net/p/adempiere/discussion/639403/thread/a1f50cbd

I snapped the last three messages to start the grappling of it.



Victor Perez Juarez
2012-04-30

Hi Marcio!

You can download the binary files http://sourceforge.net/projects/adempiere/files/Adempiere%20Packages/Libero%20Manufacturing/

kind regards
Victor Pérez
www.e-evolution.com

bma99
2012-09-05

Hi all,
any progress about this topic?
SO to MO does not work when I installed new manufacturing packages, no error message posted.
my settings for the product (all in the same orgnization):
1.The BOM Type: Make To Order
2. The BOM Use : Manufacturing
3. the product data planning: bom and workflow with the same name as the product
4. Validate: org.eevolution.model.LiberoValidator (I even did not set this while it worked in the last libero version)

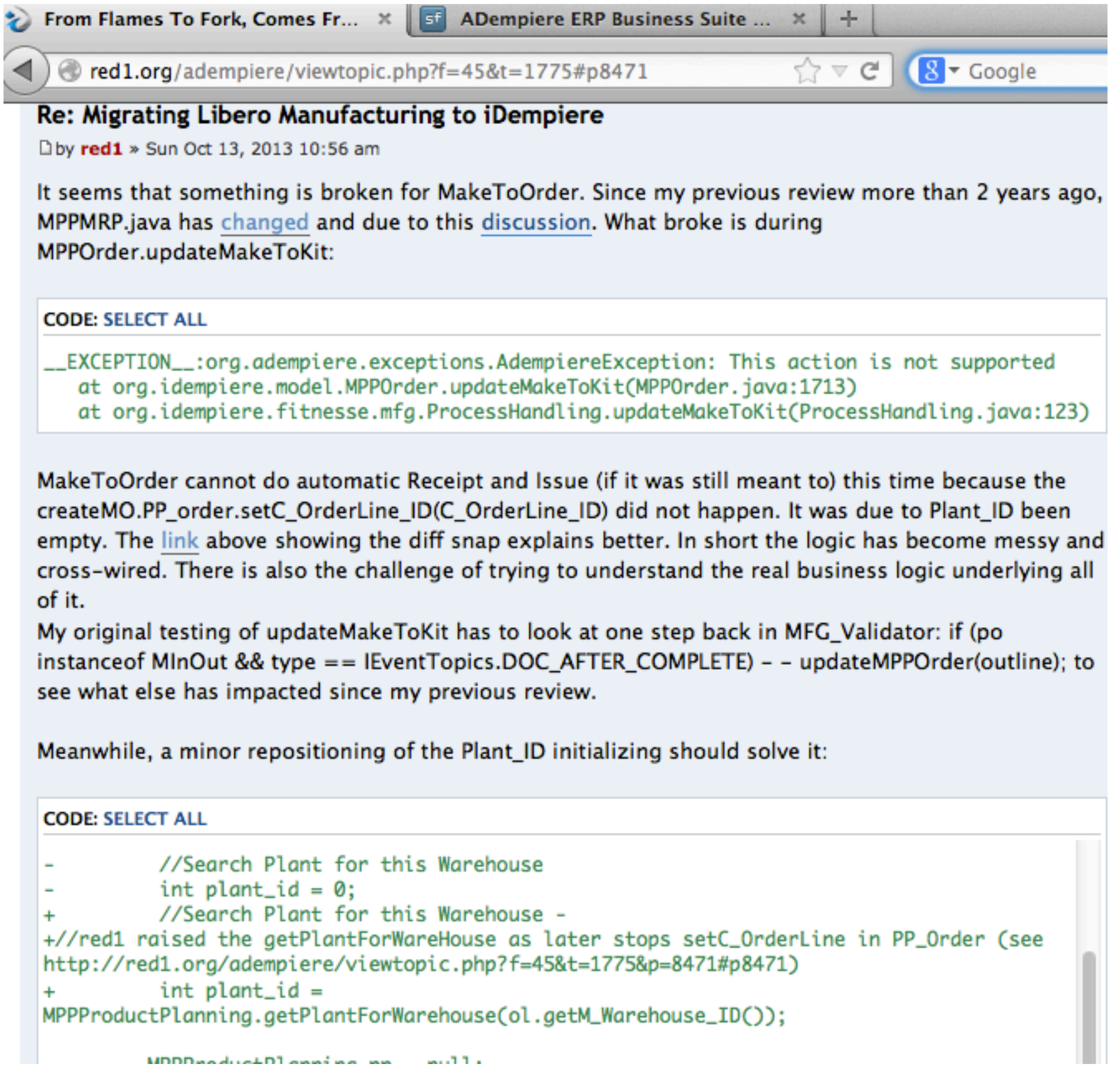
Thanks,
bma

Redhuan D. Oon
2013-10-13

The last change to PPMRP.java seems to break something. I noted and trying to resolve it here: http://red1.org/adempiere/viewtopic.php?f=45&t=1775&p=8471#p8471

The binary files referred to are from a fix done here: http://adempiere.atlassian.net/browse/MFG-10

As I hit the same error that **bma99** reported, which my 2 years' prior FitNesse tests did not encounter, I thus examined what has changed and suspect it must have broken the same case. I remarked in my forum thread above, http://red1.org/adempiere/viewtopic.php?f=45&t=1775#p8471, repeating below:


From Flames To Fork, Comes Fr...
ADempiere ERP Business Suite ...
red1.org/adempiere/viewtopic.php?f=45&t=1775#p8471
Google

Re: Migrating Libero Manufacturing to iDempiere

by **red1** » Sun Oct 13, 2013 10:56 am

It seems that something is broken for MakeToOrder. Since my previous review more than 2 years ago, MPPMRP.java has changed and due to this discussion. What broke is during MPPOrder.updateMakeToKit:

CODE: SELECT ALL

```
__EXCEPTION__:org.adempiere.exceptions.AdempiereException: This action is not supported
   at org.idempiere.model.MPPOrder.updateMakeToKit(MPPOrder.java:1713)
   at org.idempiere.fitnesse.mfg.ProcessHandling.updateMakeToKit(ProcessHandling.java:123)
```

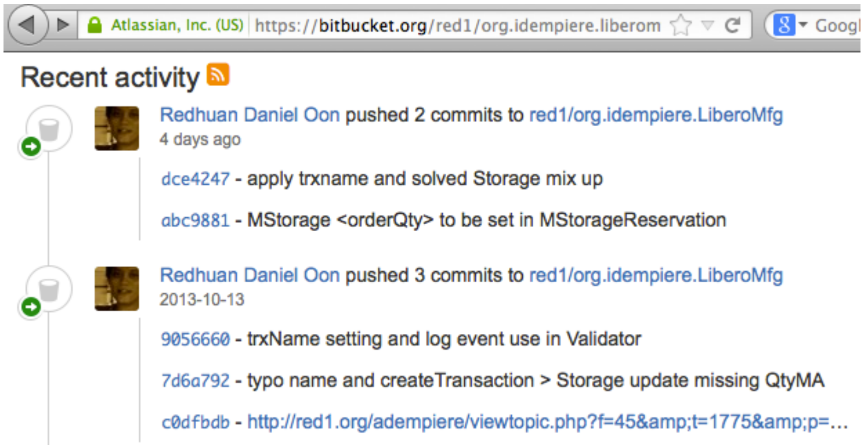
MakeToOrder cannot do automatic Receipt and Issue (if it was still meant to) this time because the createMO.PP_order.setC_OrderLine_ID(C_OrderLine_ID) did not happen. It was due to Plant_ID been empty. The link above showing the diff snap explains better. In short the logic has become messy and cross-wired. There is also the challenge of trying to understand the real business logic underlying all of it.
My original testing of updateMakeToKit has to look at one step back in MFG_Validator: if (po instanceof MInOut && type == IEventTopics.DOC_AFTER_COMPLETE) - - updateMPPOrder(outline); to see what else has impacted since my previous review.

Meanwhile, a minor repositioning of the Plant_ID initializing should solve it:

CODE: SELECT ALL

```
-         //Search Plant for this Warehouse
-         int plant_id = 0;
+         //Search Plant for this Warehouse -
+//red1 raised the getPlantForWareHouse as later stops setC_OrderLine in PP_Order (see
http://red1.org/adempiere/viewtopic.php?f=45&t=1775&p=8471#p8471)
+         int plant_id =
MPPProductPlanning.getPlantForWarehouse(ol.getM_Warehouse_ID());

          MPPProductPlanning pp = null;
```

The last part shows my code patch, that results in my FitNesse turning green again. I then commit those changes into my bitbucket:


Atlassian, Inc. (US) https://bitbucket.org/red1/org.idempiere.liberom
Googl

Recent activity

Redhuan Daniel Oon pushed 2 commits to red1/org.idempiere.LiberoMfg
4 days ago

dce4247 - apply trxname and solved Storage mix up

abc9881 - MStorage <orderQty> to be set in MStorageReservation

Redhuan Daniel Oon pushed 3 commits to red1/org.idempiere.LiberoMfg
2013-10-13

9056660 - trxName setting and log event use in Validator

7d6a792 - typo name and createTransaction > Storage update missing QtyMA

c0dfbdb - http://red1.org/adempiere/viewtopic.php?f=45&amp;t=1775&amp;p=...

The commit at the bottom is back-linked to my same forum post as reference. I hope this helps those who find something amiss to quickly review this too. Understandably MRP for manufacturing is highly complex and the creators must have it hard to resolve and be perfect. Likewise, we in turn would not be able to fare better unless we collaborate and stand on whatever good works done by others.

In spite of the 'holy wars' we fought over the stability of the trunk, which I consider very valid and important to be fought, nevertheless I have always recognised priceless contribution, and this Libero suite of applications has always been the worthiest of candidates. The least we can do is to bring it further into the final holy grail of the loosely coupled and pluggable, scalable and faster, explosively growing and well- maintained iDempiere.

Developing

Checking Out The Code

In your Eclipse workspace, you can check out the LiberoMFG code from my bitbucket onto your iDempiere workspace. If you have not setup iDempiere, you have to look for the right tutorial in www.idempiere.com that can help you do that.

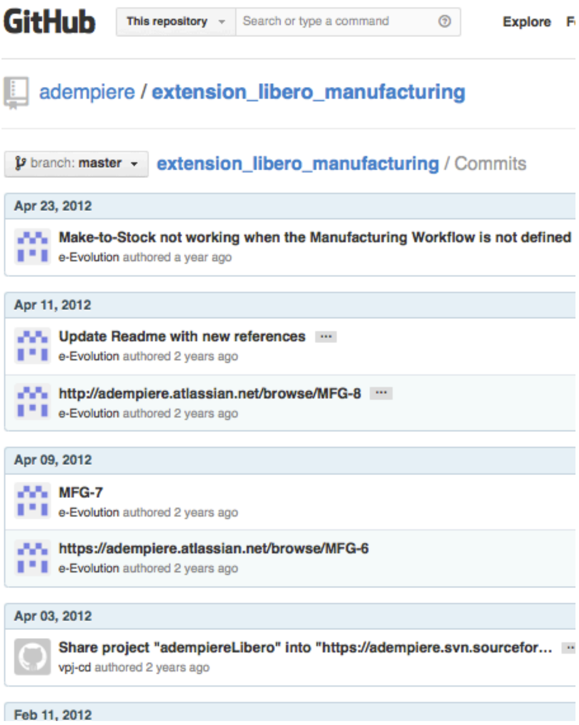
The link to the LiberoMFG module is http://bitbucket.org/red1/org.idempiere.liberomfg. This houses all the logic of the manufacturing module together with the embedded LiberoMfg.zip 2Pack.

Another plugin http://bitbucket.org/red1/org.idempiere.liberomfg_forms is for the Swing client customised panels or called forms in Compiere-speak.

Latest Code Source

Here we examine the origin of the LiberoMFG code. The original source that I reviewed 2 years ago as in this forum:

http://red1.org/adempiere/viewforum.php?f=26 which gives an earliest march 09, 2011 date for reference to track what has changed since then in the present Libero repository: http://github.com/adempiere/extension_libero_manufacturing, click on commits, and I found the latest ones and track those (see image on right) from my last review to know what happened to it since. I then apply those changes and you can see them in the bitbucket link above, by clicking on commits, you can see a list of those changes been reapplied on 2013-09-22 and 23. See the page below listing those changes to the bitbucket source.



adempiere / extension_libero_manufacturing

branch: master

extension_libero_manufacturing / Commits

Apr 23, 2012

Make-to-Stock not working when the Manufacturing Workflow is not defined
e-Evolution authored a year ago

Apr 11, 2012

Update Readme with new references …
e-Evolution authored 2 years ago

http://adempiere.atlassian.net/browse/MFG-8 …
e-Evolution authored 2 years ago

Apr 09, 2012

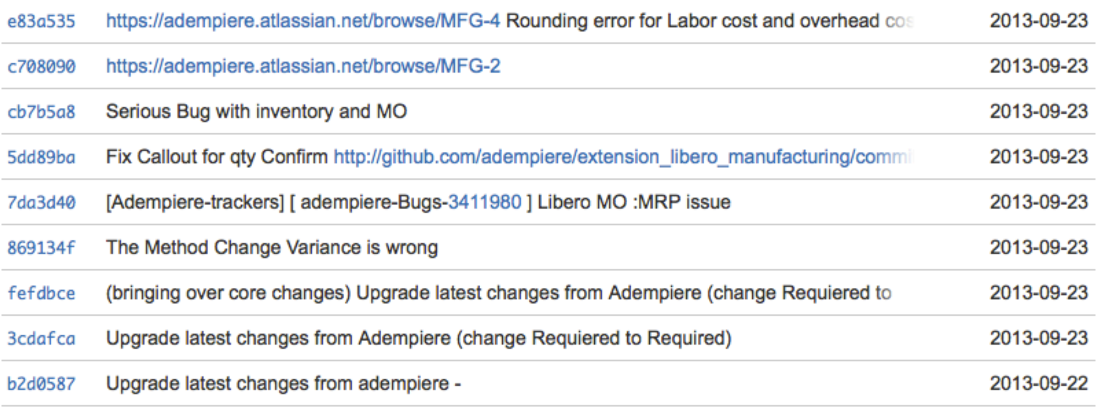
MFG-7
e-Evolution authored 2 years ago

https://adempiere.atlassian.net/browse/MFG-6
e-Evolution authored 2 years ago

Apr 03, 2012

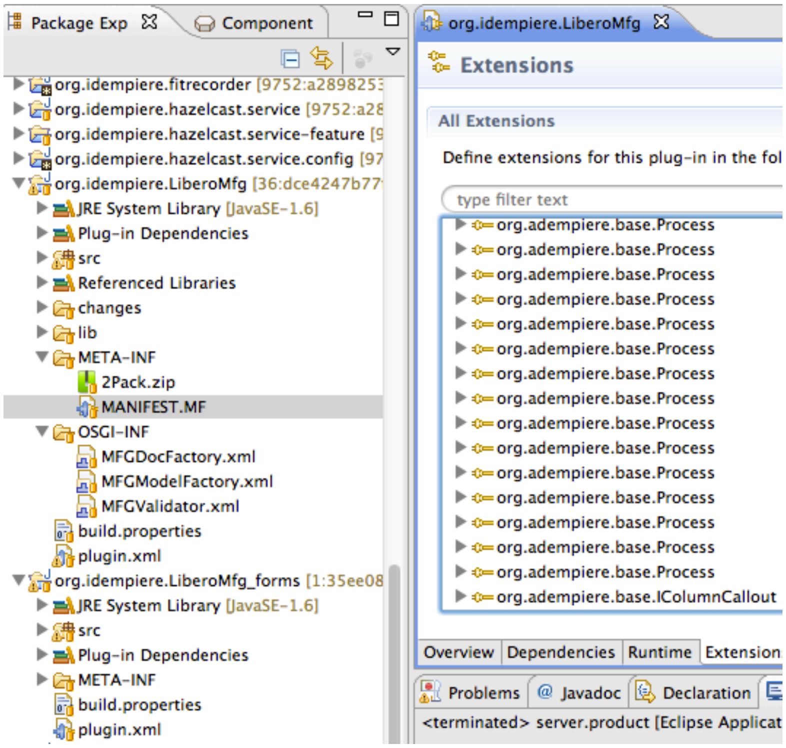
Share project "adempiereLibero" into "https://adempiere.svn.sourcefor… …
vpj-cd authored 2 years ago

Feb 11, 2012



e83a535	https://adempiere.atlassian.net/browse/MFG-4 Rounding error for Labor cost and overhead co	2013-09-23
c708090	https://adempiere.atlassian.net/browse/MFG-2	2013-09-23
cb7b5a8	Serious Bug with inventory and MO	2013-09-23
5dd89ba	Fix Callout for qty Confirm http://github.com/adempiere/extension_libero_manufacturing/comm	2013-09-23
7da3d40	[Adempiere-trackers] [adempiere-Bugs-3411980] Libero MO :MRP issue	2013-09-23
869134f	The Method Change Variance is wrong	2013-09-23
fefdbce	(bringing over core changes) Upgrade latest changes from Adempiere (change Requiered to	2013-09-23
3cdafca	Upgrade latest changes from Adempiere (change Requiered to Required)	2013-09-23
b2d0587	Upgrade latest changes from adempiere -	2013-09-22

Plugins Layout

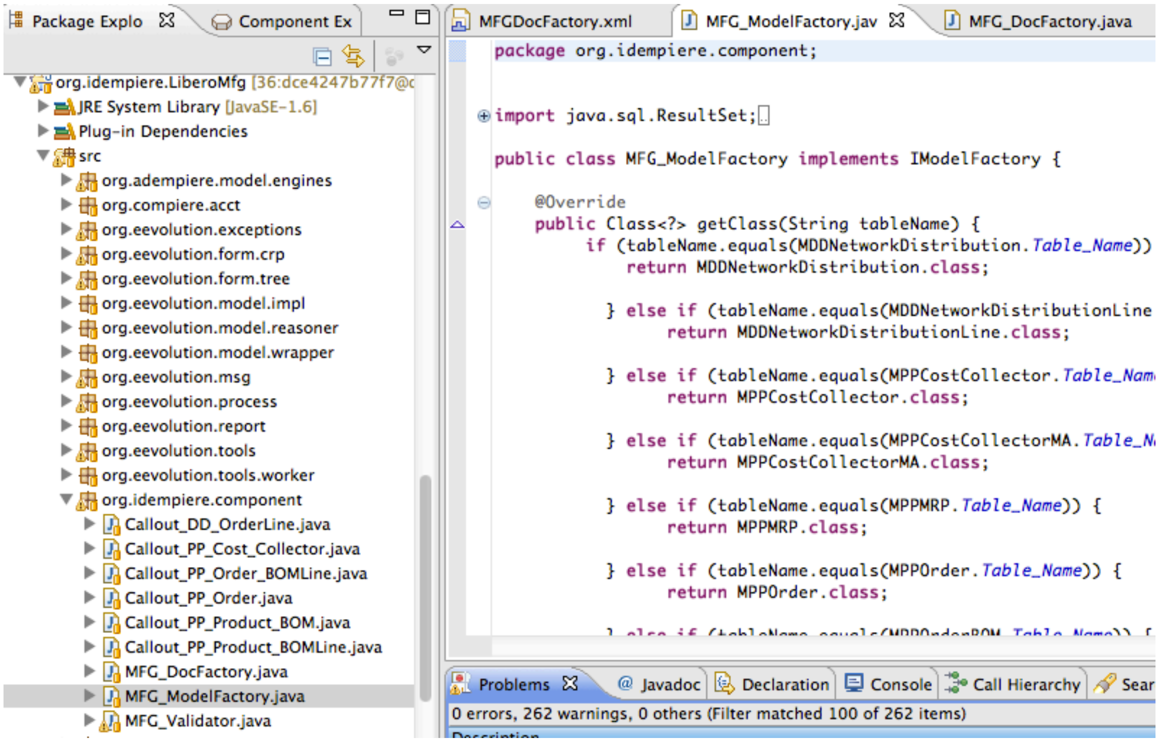

Package Exp
Component
org.idempiere.fitrecorder [9752:a289825
org.idempiere.hazelcast.service [9752:a28
org.idempiere.hazelcast.service-feature [9
org.idempiere.hazelcast.service.config [97
org.idempiere.LiberoMfg [36:dce4247b77
JRE System Library [JavaSE-1.6]
Plug-in Dependencies
src
Referenced Libraries
changes
lib
META-INF
2Pack.zip
MANIFEST.MF
OSGI-INF
MFGDocFactory.xml
MFGModelFactory.xml
MFGValidator.xml
build.properties
plugin.xml
org.idempiere.LiberoMfg_forms [1:35ee08
JRE System Library [JavaSE-1.6]
src
Plug-in Dependencies
META-INF
build.properties
plugin.xml
org.idempiere.LiberoMfg
Extensions
All Extensions
Define extensions for this plug-in in the fol
type filter text
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.Process
org.adempiere.base.IColumnCallout
Overview
Dependencies
Runtime
Extension
Problems
Javadoc
Declaration
<terminated> server.product [Eclipse Applicat

If you have followed my earlier PDFs such as Godfather III, OpenbravoPOSGuide, LiberoHR and SFAndroid, you would have been familiar with the OSGi layout style as shown here.

A plugin has a MANIFEST file that registers extensions and services. For LiberoMfg it has a 2Pack which is renamed from LiberoMFG.zip which is the PackOut of the Manufacturing Maintenance Menu and some functions and view tables specifically used by this module. Under the OSGi-INF are three XMLs or service registrants. MFGDocFactory.xml is for the FactAcct handling or Accounts posting activity. MFGModelFactory.xml is to call the MFG_ModelFactory class that declare all the Model classes in use by this module. MFGValidator.xml calls the MFG_Validator class that contains all model interception code.

Layout of Code

I try to go through some of the code layout to assist in the understanding of any developer on handling such code. I would not be able to go through in too much detail as this is a huge module and many details may be too overwhelming.



```
org.idempiere.LiberoMfg [36:dce4247b77f7@c
  JRE System Library [JavaSE-1.6]
  Plug-in Dependencies
  src
    org.adempiere.model.engines
    org.compiere.acct
    org.eevolution.exceptions
    org.eevolution.form.crp
    org.eevolution.form.tree
    org.eevolution.model.impl
    org.eevolution.model.reasoner
    org.eevolution.model.wrapper
    org.eevolution.msg
    org.eevolution.process
    org.eevolution.report
    org.eevolution.tools
    org.eevolution.tools.worker
    org.idempiere.component
      Callout_DD_OrderLine.java
      Callout_PP_Cost_Collector.java
      Callout_PP_Order_BOMLine.java
      Callout_PP_Order.java
      Callout_PP_Product_BOM.java
      Callout_PP_Product_BOMLine.java
      MFG_DocFactory.java
      MFG_ModelFactory.java
      MFG_Validator.java
```

```
package org.idempiere.component;

import java.sql.ResultSet;

public class MFG_ModelFactory implements IModelFactory {

    @Override
    public Class<?> getClass(String tableName) {
        if (tableName.equals(MDDNetworkDistribution.Table_Name))
            return MDDNetworkDistribution.class;

            } else if (tableName.equals(MDDNetworkDistributionLine
                return MDDNetworkDistributionLine.class;

            } else if (tableName.equals(MPPCostCollector.Table_Nam
                return MPPCostCollector.class;

            } else if (tableName.equals(MPPCostCollectorMA.Table_N
                return MPPCostCollectorMA.class;

            } else if (tableName.equals(MPPMRP.Table_Name)) {
                return MPPMRP.class;

            } else if (tableName.equals(MPPOrder.Table_Name)) {
                return MPPOrder.class;

            } else if (tableName.equals(MPPOrderBOM.Table_Name)) {
```

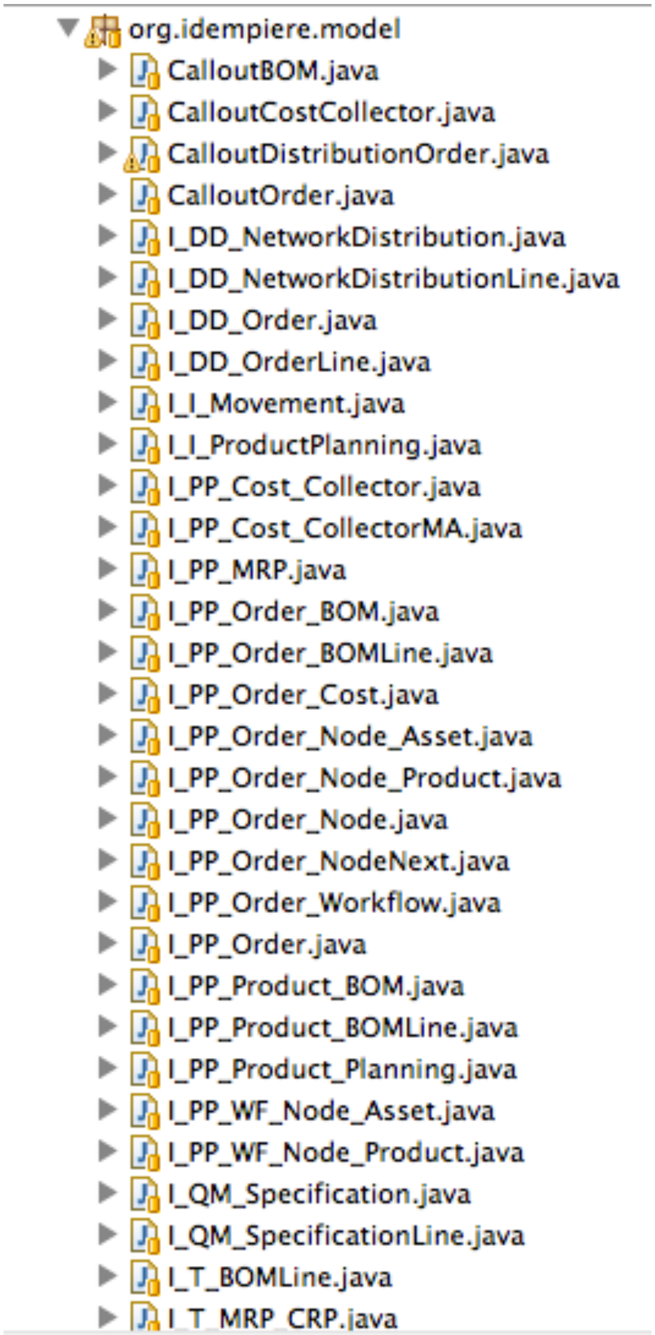
```
0 errors, 262 warnings, 0 others (Filter matched 100 of 262 items)
Description
```

The above shows the content of org.idempiere.component package which I set as a standard across all plugins I made this for putting in new code that interface with the legacy code and OSGi extensions or services. The legacy code is copied from org.eevolution.model in the core iDempiere to a new package org.idempiere.model (see next screen-shot below).

The new code examples are: The Callout classes implementing the CalloutFactory interface and refers to the legacy Callouts; The MFG_DocFactory interface with the FactAcct or Accounting consequence handling; The MFG_ModelFactory interface with the Model classes and lastly the MFG_Validator does all the *Libero Validator* event handlings.

The org.idempiere.model package is as shown on the right. I have made a copy of the org.eevolution.model in the org.adempiere.base plugin to here as the inter-dependencies among them are heavy. I have proposed to Carlos Ruiz to drop the org.eevolution.model from the base plugin. This is to make a clean separation and remove tight coupling within the code. However dropping that cannot be done without wrapping extensions by LiberoMFG on core classes such as DocumentEngine. According to Carlos, Heng Sin can advice on how to do that. This will mean another visit to my favorite guru and treating him to a good lunch. :)

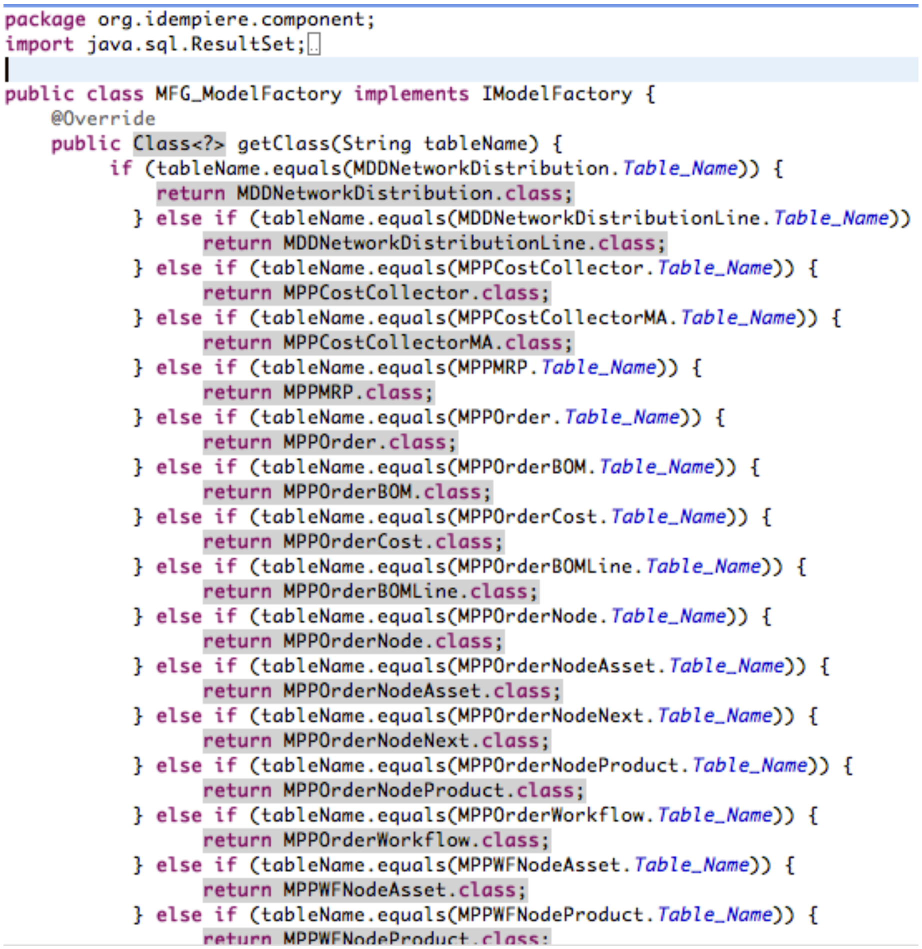
The advice I got before from Heng Sin is to use a differentiated package name for classes that may override core classes and thus I am using the name, org.idempiere.model. This will ensure no ambiguity when referring to the former legacy class, which may now be edited and not impose on the core base plugin.



```
▼ org.idempiere.model
  ▶ CalloutBOM.java
  ▶ CalloutCostCollector.java
  ▶ CalloutDistributionOrder.java
  ▶ CalloutOrder.java
  ▶ I_DD_NetworkDistribution.java
  ▶ I_DD_NetworkDistributionLine.java
  ▶ I_DD_Order.java
  ▶ I_DD_OrderLine.java
  ▶ I_I_Movement.java
  ▶ I_I_ProductPlanning.java
  ▶ I_PP_Cost_Collector.java
  ▶ I_PP_Cost_CollectorMA.java
  ▶ I_PP_MRP.java
  ▶ I_PP_Order_BOM.java
  ▶ I_PP_Order_BOMLine.java
  ▶ I_PP_Order_Cost.java
  ▶ I_PP_Order_Node_Asset.java
  ▶ I_PP_Order_Node_Product.java
  ▶ I_PP_Order_Node.java
  ▶ I_PP_Order_NodeNext.java
  ▶ I_PP_Order_Workflow.java
  ▶ I_PP_Order.java
  ▶ I_PP_Product_BOM.java
  ▶ I_PP_Product_BOMLine.java
  ▶ I_PP_Product_Planning.java
  ▶ I_PP_WF_Node_Asset.java
  ▶ I_PP_WF_Node_Product.java
  ▶ I_QM_Specification.java
  ▶ I_QM_SpecificationLine.java
  ▶ I_T_BOMLine.java
  ▶ I_T_MRP_CRP.java
```

Model Factory

As a follow-up of the above, the Model classes contained there and referred to by the code has to be registered under the MFG_ModelFactory class. You can see in the code link below that every model is declared. Otherwise there will be a ClassLoading error or casting error as that model was not loadable in the first place. You can see the class further at http://bitbucket.org/red1/org.idempiere.liberomfg/src, and go to src/org/idempiere/component/MFG_ModelFactory.java.



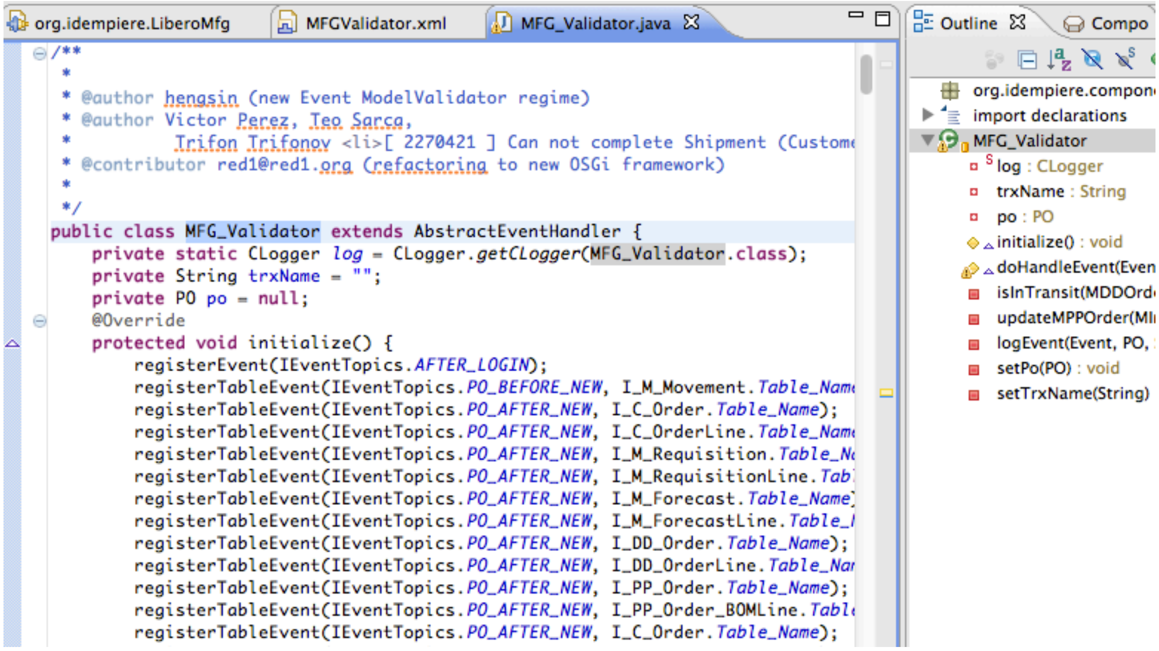
```
package org.idempiere.component;
import java.sql.ResultSet;

public class MFG_ModelFactory implements IModelFactory {
    @Override
    public Class<?> getClass(String tableName) {
         if (tableName.equals(MDDNetworkDistribution.Table_Name)) {
             return MDDNetworkDistribution.class;
           } else if (tableName.equals(MDDNetworkDistributionLine.Table_Name))
                 return MDDNetworkDistributionLine.class;
           } else if (tableName.equals(MPPCostCollector.Table_Name)) {
                 return MPPCostCollector.class;
           } else if (tableName.equals(MPPCostCollectorMA.Table_Name)) {
                 return MPPCostCollectorMA.class;
           } else if (tableName.equals(MPPMRP.Table_Name)) {
                 return MPPMRP.class;
           } else if (tableName.equals(MPPOrder.Table_Name)) {
                 return MPPOrder.class;
           } else if (tableName.equals(MPPOrderBOM.Table_Name)) {
                 return MPPOrderBOM.class;
           } else if (tableName.equals(MPPOrderCost.Table_Name)) {
                 return MPPOrderCost.class;
           } else if (tableName.equals(MPPOrderBOMLine.Table_Name)) {
                 return MPPOrderBOMLine.class;
           } else if (tableName.equals(MPPOrderNode.Table_Name)) {
                 return MPPOrderNode.class;
           } else if (tableName.equals(MPPOrderNodeAsset.Table_Name)) {
                 return MPPOrderNodeAsset.class;
           } else if (tableName.equals(MPPOrderNodeNext.Table_Name)) {
                 return MPPOrderNodeNext.class;
           } else if (tableName.equals(MPPOrderNodeProduct.Table_Name)) {
                 return MPPOrderNodeProduct.class;
           } else if (tableName.equals(MPPOrderWorkflow.Table_Name)) {
                 return MPPOrderWorkflow.class;
           } else if (tableName.equals(MPPWFNodeAsset.Table_Name)) {
                 return MPPWFNodeAsset.class;
           } else if (tableName.equals(MPPWFNodeProduct.Table_Name)) {
                 return MPPWFNodeProduct.class;
```

This class in turn is referenced in the MFGModelFactory.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<scr:component xmlns:scr="http://www.osgi.org/xmlns/scr/v1.1.0"
name="org.idempiere.component.MFG_ModelFactory">
   <implementation class="org.idempiere.component.MFG_ModelFactory"/>
   <property name="service.ranking" type="Integer" value="1"/>
   <service>
      <provide interface="org.adempiere.base.IModelFactory"/>
   </service>
</scr:component>
```

ModelValidator Code

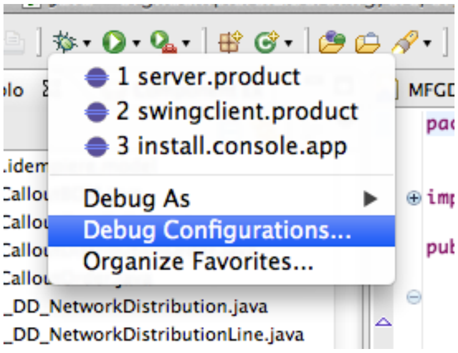
This is perhaps the most important part of any 3rd party external plugin module. It contains the different processing logic to happen in the model events or posting activity. It works on the old Compiere technology of ModelValidator where an official DocType such as C_Order or C_Invoice has their main events such as AfterNew, BeforeSave and AfterComplete hooked for extensions handling in a separate class. I have taken wholesale the legacy LiberoValidator class and adapted as a new MFG_Validator class which implements the new interface and naming convention. Below is a screen-shot off my Eclipse. This class is called from MFG_Validator.xml.

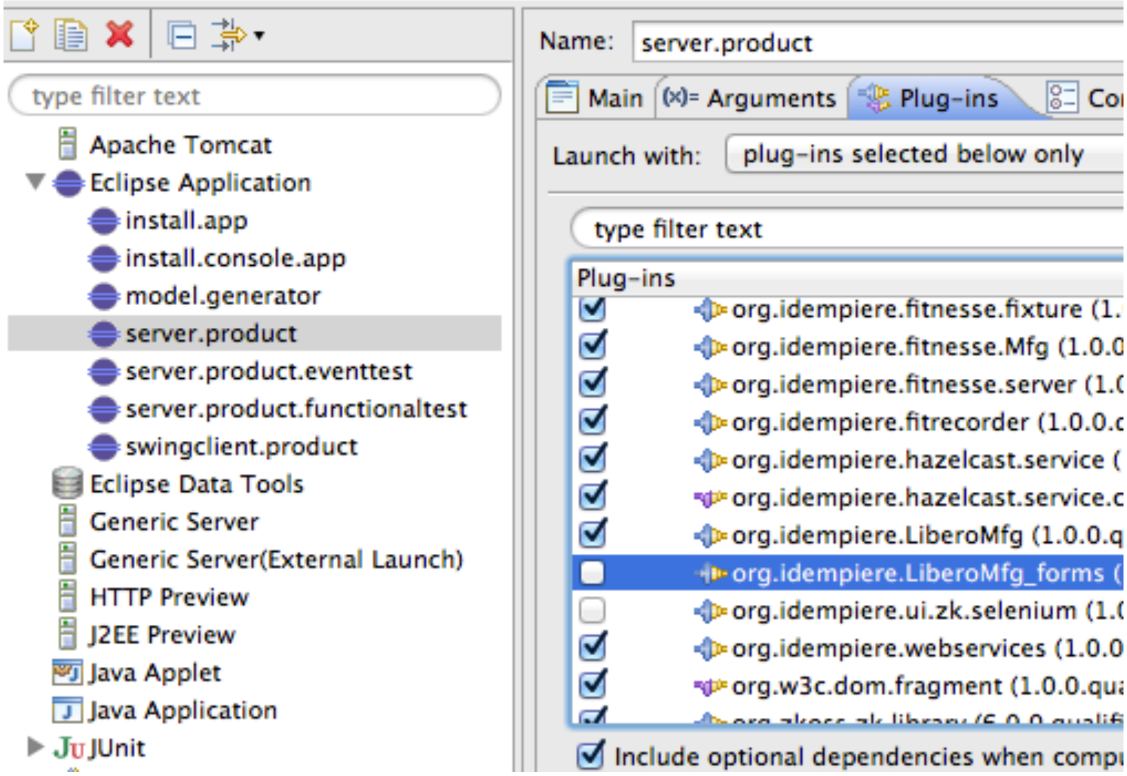


```
/**
 *
 * @author hengsin (new Event ModelValidator regime)
 * @author Victor Perez, Teo Sarca,
 *         Trifon Trifonov <li>[ 2270421 ] Can not complete Shipment (Custome
 * @contributor red1@red1.org (refactoring to new OSGi framework)
 *
 */
public class MFG_Validator extends AbstractEventHandler {
    private static CLogger log = CLogger.getCLogger(MFG_Validator.class);
    private String trxName = "";
    private PO po = null;
    @Override
    protected void initialize() {
        registerEvent(IEventTopics.AFTER_LOGIN);
        registerTableEvent(IEventTopics.PO_BEFORE_NEW, I_M_Movement.Table_Name
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_C_Order.Table_Name);
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_C_OrderLine.Table_Name
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_M_Requisition.Table_N
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_M_RequisitionLine.Tab
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_M_Forecast.Table_Name)
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_M_ForecastLine.Table_
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_DD_Order.Table_Name);
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_DD_OrderLine.Table_Na
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_PP_Order.Table_Name);
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_PP_Order_BOMLine.Tabl
        registerTableEvent(IEventTopics.PO_AFTER_NEW, I_C_Order.Table_Name);
```

Launching in Eclipse

This is quite an easy matter today with the latest iDempiere settings. Just go to the Run As or Debug As function in your Eclipse IDE and choose the Debug Configurations.. Of course you can go straight and choose the 3 options already defined such as server.product. But Configuring As.. is to check that you got the new plugins activated to run along with the iDempiere stack. Otherwise they won't be registered and cannot be made active for reference by your data model.





Name: server.product

Launch with: plug-ins selected below only

Plug-ins
- ☑ org.idempiere.fitnesse.fixture (1.
- ☑ org.idempiere.fitnesse.Mfg (1.0.0
- ☑ org.idempiere.fitnesse.server (1.0
- ☑ org.idempiere.fitrecorder (1.0.0.c
- ☑ org.idempiere.hazelcast.service (
- ☑ org.idempiere.hazelcast.service.c
- ☑ org.idempiere.LiberoMfg (1.0.0.q
- □ org.idempiere.LiberoMfg_forms (
- □ org.idempiere.ui.zk.selenium (1.0
- ☑ org.idempiere.webservices (1.0.0
- ☑ org.w3c.dom.fragment (1.0.0.qua

☑ Include optional dependencies when compi

When the Configuration window opens, you can select the Eclipse Application you intend to use and go to the Plug-ins tab check the plugin projects concerned. Here, we have the testing plugins checked as well as the LiberoMfg plugin. The LiberoMfg_forms is the option for swingclient.product. You do not use it under server.product which launches the ZK Ajax Web UI for use in a web browser.

Plugin Extensions

Processes and Callouts are loaded via Extensions and they are declared in the MANIFEST's plugin.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.4"?>
<plugin>
   <extension
         id="org.eevolution.process.CalculateLowLevel"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CalculateLowLevel">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CompletePrintOrder"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CompletePrintOrder">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.ComponentChange"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.ComponentChange">
      </process>
   </extension>
   <extension
         id="org.idempiere.liberomfg.ProcessCall4"
         point="org.adempiere.base.Process">
      <process
            class="org.idempiere.liberomfg.ProcessCall4">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CopyFromBOM"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CopyFromBOM">
      </process>
   </extension>
   <extension
         id="id6"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CopyPriceToStandard">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CreateCostElement"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CreateCostElement">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CreateDocType"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CreateDocType">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CreateProductPlanning"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CreateProductPlanning">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CRP"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CRP">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.CRPSummary"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.CRPSummary">
      </process>
   </extension>
   <extension
         id="org.compiere.process.DistributionRun"
         point="org.adempiere.base.Process">
      <process
            class="org.compiere.process.DistributionRun">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.FixPaymentCashLine"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.FixPaymentCashLine">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.ImportProductPlanning"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.ImportProductPlanning">
      </process>
   </extension>
   <extension
```

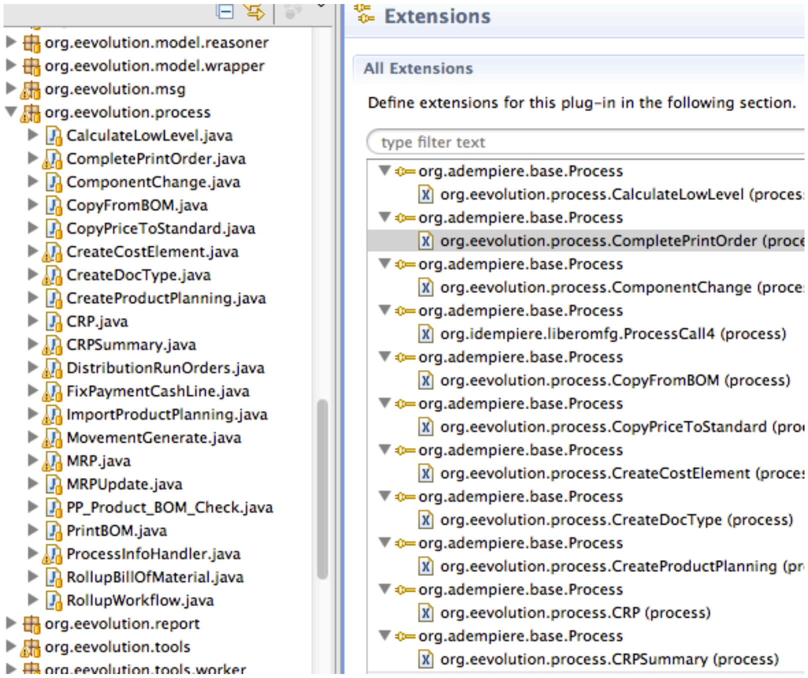
```
         id="org.eevolution.process.MovementGenerate"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.MovementGenerate">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.MRP"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.MRP">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.MRPUpdate"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.MRPUpdate">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.PP_Product_BOM_Check"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.PP_Product_BOM_Check">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.PrintBOM"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.PrintBOM">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.RollupBillOfMaterial"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.RollupBillOfMaterial">
      </process>
   </extension>
   <extension
         id="org.eevolution.process.RollupWorkflow"
         point="org.adempiere.base.Process">
      <process
            class="org.eevolution.process.RollupWorkflow">
      </process>
   </extension>
   <extension
         point="org.adempiere.base.IColumnCallout">
      <callout
            class="org.idempiere.component.Callout_PP_Product_BOMLine"
            columnName="M_Product_ID"
            tableName="PP_Product_BOMLine">
```

```
        </callout>
        <callout
              class="org.idempiere.component.Callout_PP_Product_BOM"
              columnName="M_Product_ID"
              tableName="PP_Product_BOM">
        </callout>
        <callout
              class="org.idempiere.component.Callout_PP_Order"
              columnName="QtyEntered"
              tableName="PP_Order">
        </callout>
        <callout
              class="org.idempiere.component.Callout_PP_Order"
              columnName="M_Product_ID"
              tableName="PP_Order">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="M_Product_ID"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="M_AttributeSetInstance_ID"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="C_UOM_ID"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="ConfirmedQty"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="M_AttributeSetInstanceTo_ID"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="QtyOrdered"
              tableName="DD_OrderLine">
        </callout>
        <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="QtyEntered"
              tableName="DD_OrderLine">
        </callout>
              <callout
              class="org.idempiere.component.Callout_DD_OrderLine"
              columnName="AD_Org_ID"
```

```
            tableName="DD_Order">
      </callout>
      <callout
            class="org.idempiere.component.Callout_PP_Cost_Collector"
            columnName="PP_Order_ID"
            tableName="PP_Cost_Collector">
      </callout>
      <callout
            class="org.idempiere.component.Callout_PP_Cost_Collector"
            columnName="PP_Order_Node_ID"
            tableName="PP_Cost_Collector">
      </callout>
      <callout
            class="org.idempiere.component.Callout_PP_Cost_Collector"
            columnName="MovementQty"
            tableName="PP_Cost_Collector">
      </callout>
      <callout
            class="org.idempiere.component.Callout_PP_Order_BOMLine"
            columnName="QtyEntered"
            tableName="PP_Order_BOMLine">
      </callout>
      <callout
            class="org.idempiere.component.Callout_PP_Order_BOMLine"
            columnName="QtyRequiered"
            tableName="PP_Order_BOMLine">
      </callout>
   </extension>
</plugin>
```

The Callouts refer to within the org.idempiere.component package that extends from the legacy Callouts in org.idempiere.model.

The Processes refer directly to the legacy org.eevolution.process package shown on the right. That is copied intact from base plugin.



- org.eevolution.model.reasoner
- org.eevolution.model.wrapper
- org.eevolution.msg
- org.eevolution.process
    - CalculateLowLevel.java
    - CompletePrintOrder.java
    - ComponentChange.java
    - CopyFromBOM.java
    - CopyPriceToStandard.java
    - CreateCostElement.java
    - CreateDocType.java
    - CreateProductPlanning.java
    - CRP.java
    - CRPSummary.java
    - DistributionRunOrders.java
    - FixPaymentCashLine.java
    - ImportProductPlanning.java
    - MovementGenerate.java
    - MRP.java
    - MRPUpdate.java
    - PP_Product_BOM_Check.java
    - PrintBOM.java
    - ProcessInfoHandler.java
    - RollupBillOfMaterial.java
    - RollupWorkflow.java
- org.eevolution.report
- org.eevolution.tools
- org.eevolution.tools.worker

Extensions

All Extensions

Define extensions for this plug-in in the following section.

type filter text

- org.adempiere.base.Process
    - org.eevolution.process.CalculateLowLevel (proces
- org.adempiere.base.Process
    - org.eevolution.process.CompletePrintOrder (proce
- org.adempiere.base.Process
    - org.eevolution.process.ComponentChange (proce
- org.adempiere.base.Process
    - org.idempiere.liberomfg.ProcessCall4 (process)
- org.adempiere.base.Process
    - org.eevolution.process.CopyFromBOM (process)
- org.adempiere.base.Process
    - org.eevolution.process.CopyPriceToStandard (pro
- org.adempiere.base.Process
    - org.eevolution.process.CreateCostElement (proce
- org.adempiere.base.Process
    - org.eevolution.process.CreateDocType (process)
- org.adempiere.base.Process
    - org.eevolution.process.CreateProductPlanning (pr
- org.adempiere.base.Process
    - org.eevolution.process.CRP (process)
- org.adempiere.base.Process
    - org.eevolution.process.CRPSummary (process)

Callouts

The Callout declarations above seems a lot. So I am pasting here the legacy definition for convenient reference in case there is confusion and a need to double-check which Callout is been accounted for.

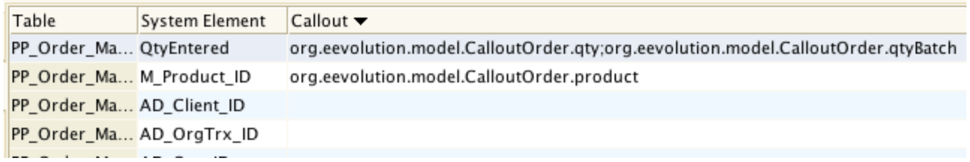


Table	System Element	Callout ▾
PP_Order_Ma…	QtyEntered	org.eevolution.model.CalloutOrder.qty;org.eevolution.model.CalloutOrder.qtyBatch
PP_Order_Ma…	M_Product_ID	org.eevolution.model.CalloutOrder.product
PP_Order_Ma…	AD_Client_ID	
PP_Order_Ma…	AD_OrgTrx_ID	
PP_Order_Ma…	AD_Org_ID	

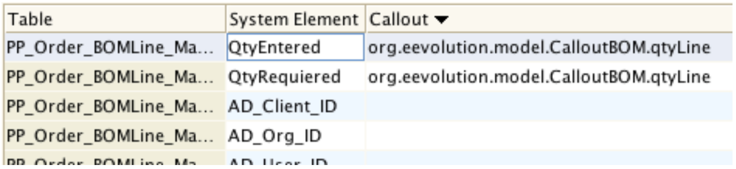


Table	System Element	Callout ▾
PP_Order_BOMLine_Ma…	QtyEntered	org.eevolution.model.CalloutBOM.qtyLine
PP_Order_BOMLine_Ma…	QtyRequiered	org.eevolution.model.CalloutBOM.qtyLine
PP_Order_BOMLine_Ma…	AD_Client_ID	
PP_Order_BOMLine_Ma…	AD_Org_ID	
PP_Order_BOMLine_Ma…	AD_User_ID	

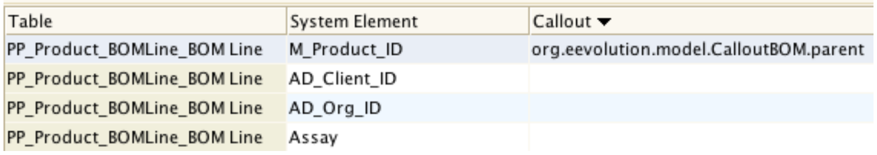


Table	System Element	Callout ▾
PP_Product_BOMLine_BOM Line	M_Product_ID	org.eevolution.model.CalloutBOM.parent
PP_Product_BOMLine_BOM Line	AD_Client_ID	
PP_Product_BOMLine_BOM Line	AD_Org_ID	
PP_Product_BOMLine_BOM Line	Assay	

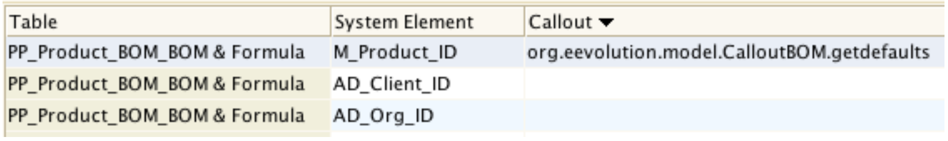


Table	System Element	Callout ▾
PP_Product_BOM_BOM & Formula	M_Product_ID	org.eevolution.model.CalloutBOM.getdefaults
PP_Product_BOM_BOM & Formula	AD_Client_ID	
PP_Product_BOM_BOM & Formula	AD_Org_ID	

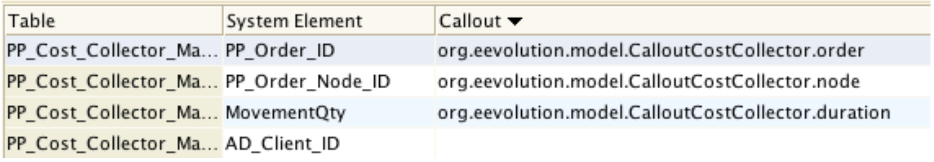


Table	System Element	Callout ▾
PP_Cost_Collector_Ma…	PP_Order_ID	org.eevolution.model.CalloutCostCollector.order
PP_Cost_Collector_Ma…	PP_Order_Node_ID	org.eevolution.model.CalloutCostCollector.node
PP_Cost_Collector_Ma…	MovementQty	org.eevolution.model.CalloutCostCollector.duration
PP_Cost_Collector_Ma…	AD_Client_ID	

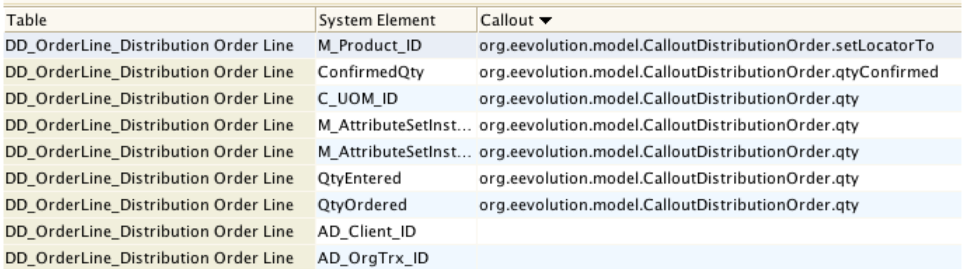


Table	System Element	Callout ▾
DD_OrderLine_Distribution Order Line	M_Product_ID	org.eevolution.model.CalloutDistributionOrder.setLocatorTo
DD_OrderLine_Distribution Order Line	ConfirmedQty	org.eevolution.model.CalloutDistributionOrder.qtyConfirmed
DD_OrderLine_Distribution Order Line	C_UOM_ID	org.eevolution.model.CalloutDistributionOrder.qty
DD_OrderLine_Distribution Order Line	M_AttributeSetInst…	org.eevolution.model.CalloutDistributionOrder.qty
DD_OrderLine_Distribution Order Line	M_AttributeSetInst…	org.eevolution.model.CalloutDistributionOrder.qty
DD_OrderLine_Distribution Order Line	QtyEntered	org.eevolution.model.CalloutDistributionOrder.qty
DD_OrderLine_Distribution Order Line	QtyOrdered	org.eevolution.model.CalloutDistributionOrder.qty
DD_OrderLine_Distribution Order Line	AD_Client_ID	
DD_OrderLine_Distribution Order Line	AD_OrgTrx_ID	

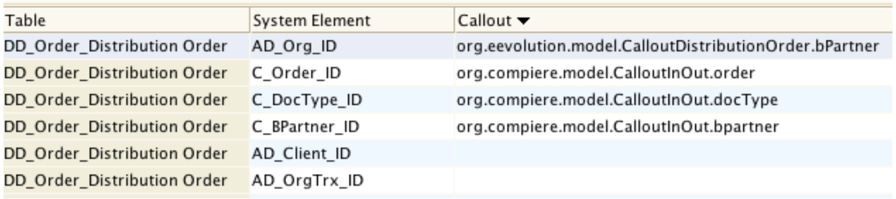
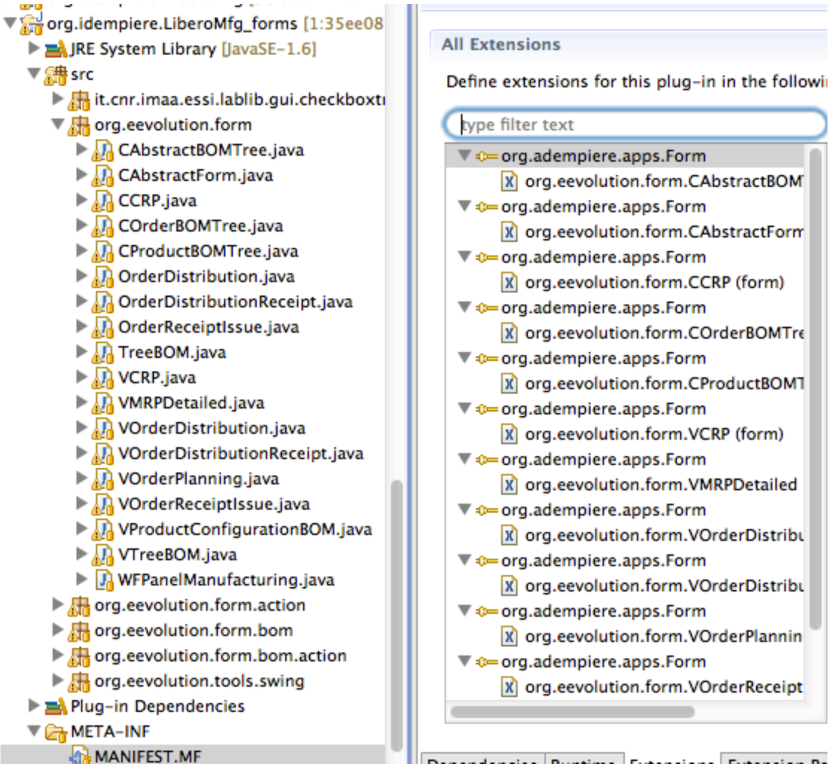


Table	System Element	Callout ▾
DD_Order_Distribution Order	AD_Org_ID	org.eevolution.model.CalloutDistributionOrder.bPartner
DD_Order_Distribution Order	C_Order_ID	org.compiere.model.CalloutInOut.order
DD_Order_Distribution Order	C_DocType_ID	org.compiere.model.CalloutInOut.docType
DD_Order_Distribution Order	C_BPartner_ID	org.compiere.model.CalloutInOut.bpartner
DD_Order_Distribution Order	AD_Client_ID	
DD_Order_Distribution Order	AD_OrgTrx_ID	

Swing Forms

The LiberoMFG customised dialog forms are mostly created for the Swing client and so a packaging into a separate plugin is required for the adempiere-client instance. I gave it a suffix name as: LiberoMfg_forms. In it the plugin.xml only addresses those forms for the Swing client:



```
▼ org.idempiere.LiberoMfg_forms [1:35ee08
   ▶ JRE System Library [JavaSE-1.6]
   ▼ src
      ▶ it.cnr.imaa.essi.lablib.gui.checkboxt
      ▼ org.eevolution.form
         ▶ CAbstractBOMTree.java
         ▶ CAbstractForm.java
         ▶ CCRP.java
         ▶ COrderBOMTree.java
         ▶ CProductBOMTree.java
         ▶ OrderDistribution.java
         ▶ OrderDistributionReceipt.java
         ▶ OrderReceiptIssue.java
         ▶ TreeBOM.java
         ▶ VCRP.java
         ▶ VMRPDetailed.java
         ▶ VOrderDistribution.java
         ▶ VOrderDistributionReceipt.java
         ▶ VOrderPlanning.java
         ▶ VOrderReceiptIssue.java
         ▶ VProductConfigurationBOM.java
         ▶ VTreeBOM.java
         ▶ WFPanelManufacturing.java
      ▶ org.eevolution.form.action
      ▶ org.eevolution.form.bom
      ▶ org.eevolution.form.bom.action
      ▶ org.eevolution.tools.swing
   ▶ Plug-in Dependencies
   ▼ META-INF
      MANIFEST.MF
```

```
All Extensions
Define extensions for this plug-in in the followi
type filter text
▼ org.adempiere.apps.Form
     org.eevolution.form.CAbstractBOM
▼ org.adempiere.apps.Form
     org.eevolution.form.CAbstractForm
▼ org.adempiere.apps.Form
     org.eevolution.form.CCRP (form)
▼ org.adempiere.apps.Form
     org.eevolution.form.COrderBOMTre
▼ org.adempiere.apps.Form
     org.eevolution.form.CProductBOMT
▼ org.adempiere.apps.Form
     org.eevolution.form.VCRP (form)
▼ org.adempiere.apps.Form
     org.eevolution.form.VMRPDetailed
▼ org.adempiere.apps.Form
     org.eevolution.form.VOrderDistribu
▼ org.adempiere.apps.Form
     org.eevolution.form.VOrderDistribu
▼ org.adempiere.apps.Form
     org.eevolution.form.VOrderPlannin
▼ org.adempiere.apps.Form
     org.eevolution.form.VOrderReceipt
```

Core Code Entanglement

I mentioned earlier that dropping the Libero packages from core has some impact. The class I am talking about is in DocumentEngine.java where the PP_Order and PP_Cost_Collector document actions are controlled:

```
/********************
*  Manufacturing Order
 */
else if (AD_Table_ID == I_PP_Order.Table_ID)
 {
        if (docStatus.equals(DocumentEngine.STATUS_Drafted)
                || docStatus.equals(DocumentEngine.STATUS_InProgress)
                || docStatus.equals(DocumentEngine.STATUS_Invalid))
        {
                options[index++] = DocumentEngine.ACTION_Prepare;
                options[index++] = DocumentEngine.ACTION_Close;
        }
        //          Complete                 ..  CO
else if (docStatus.equals(DocumentEngine.STATUS_Completed))
        {
                options[index++] = DocumentEngine.ACTION_Void;
                options[index++] = DocumentEngine.ACTION_ReActivate;
        }
}
/********************
*  Manufacturing Cost Collector
 */
else if (AD_Table_ID == I_PP_Cost_Collector.Table_ID)
```

2Pack Preparation

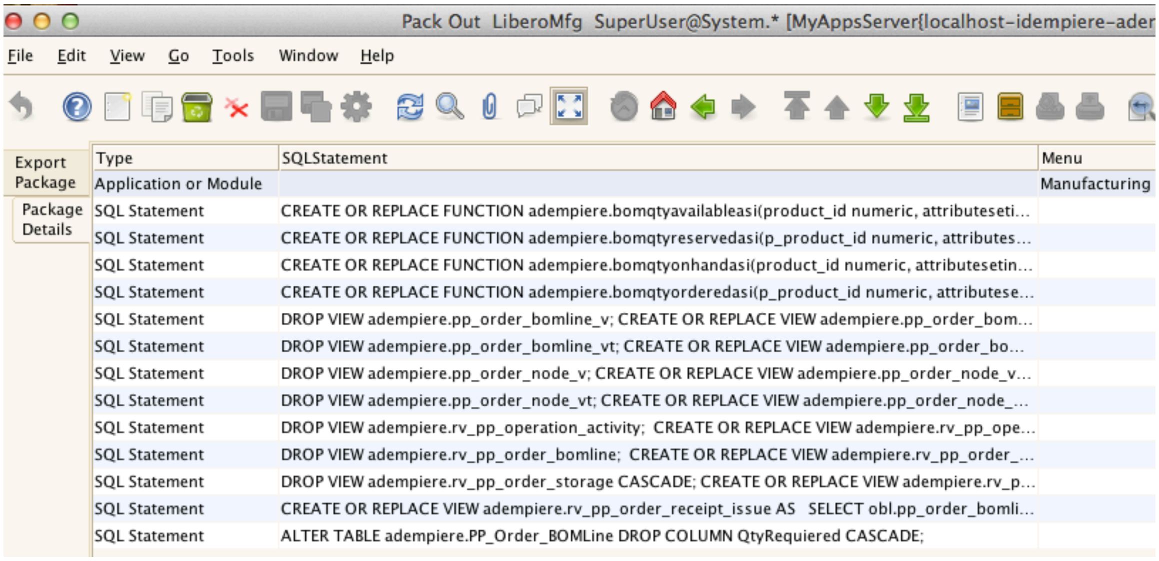
I usually save the database instance which I used to produce the LiberoMfg.zip 2Pack that is embedded into the LiberoMfg plugin. You can retrieve the database form my SourceForge repo:

http://sourceforge.net/projects/red1/files/p2/LiberoMFG/ExpDat20131009_123527.jar/download

Restoring this database allows you to go to the Pack Out window and examine the settings and modify it to produce another version of the 2Pack. This is needed when a future bug occurs with the Pack In.

At the moment the PackOut is a simple main record of the Manufacturing Menu. I have to add the recently dropped functions from iDempiere: http://idempiere.atlassian.net/browse/IDEMPIERE-1370

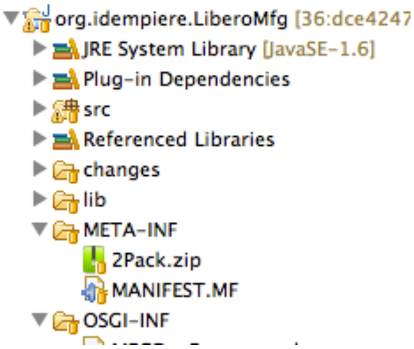
Then I have to add the views that are affected by the typo 'requiered' change to 'required'.


Pack Out LiberoMfg SuperUser@System.* [MyAppsServer{localhost-idempiere-ader
File Edit View Go Tools Window Help
Export Package
Package Details

Type	SQLStatement	Menu
Application or Module		Manufacturing
SQL Statement	CREATE OR REPLACE FUNCTION adempiere.bomqtyavailableasi(product_id numeric, attributeseti...	
SQL Statement	CREATE OR REPLACE FUNCTION adempiere.bomqtyreservedasi(p_product_id numeric, attributes...	
SQL Statement	CREATE OR REPLACE FUNCTION adempiere.bomqtyonhandasi(product_id numeric, attributesetin...	
SQL Statement	CREATE OR REPLACE FUNCTION adempiere.bomqtyorderedasi(p_product_id numeric, attributese...	
SQL Statement	DROP VIEW adempiere.pp_order_bomline_v; CREATE OR REPLACE VIEW adempiere.pp_order_bom...	
SQL Statement	DROP VIEW adempiere.pp_order_bomline_vt; CREATE OR REPLACE VIEW adempiere.pp_order_bo...	
SQL Statement	DROP VIEW adempiere.pp_order_node_v; CREATE OR REPLACE VIEW adempiere.pp_order_node_v...	
SQL Statement	DROP VIEW adempiere.pp_order_node_vt; CREATE OR REPLACE VIEW adempiere.pp_order_node_...	
SQL Statement	DROP VIEW adempiere.rv_pp_operation_activity; CREATE OR REPLACE VIEW adempiere.rv_pp_ope...	
SQL Statement	DROP VIEW adempiere.rv_pp_order_bomline; CREATE OR REPLACE VIEW adempiere.rv_pp_order_...	
SQL Statement	DROP VIEW adempiere.rv_pp_order_storage CASCADE; CREATE OR REPLACE VIEW adempiere.rv_p...	
SQL Statement	CREATE OR REPLACE VIEW adempiere.rv_pp_order_receipt_issue AS SELECT obl.pp_order_bomli...	
SQL Statement	ALTER TABLE adempiere.PP_Order_BOMLine DROP COLUMN QtyRequiered CASCADE;	

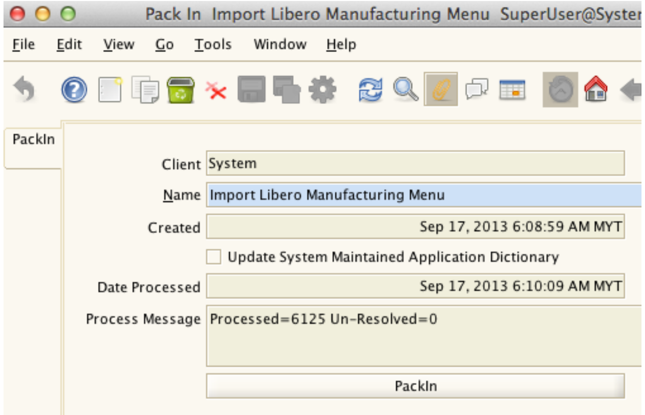
The above is the resulting complete PackOut package needed and I have tested rigorously for both adempiere-client, idempiere-server and over Eclipse source, so far it worked very well with no errors.

Once you PackOut again, put the LiberoMfg.zip in the plugin's META-INF and rename it to 2Pack.zip. Remember its leading caps, and case-sensitive.


org.idempiere.LiberoMfg [36:dce4247
JRE System Library [JavaSE-1.6]
Plug-in Dependencies
src
Referenced Libraries
changes
lib
META-INF
2Pack.zip
MANIFEST.MF
OSGI-INF

During Pack-In, the final successful result is as shown above.

Processed=6125 Un-Resolved=0



Client: System

Name: Import Libero Manufacturing Menu

Created: Sep 17, 2013 6:08:59 AM MYT

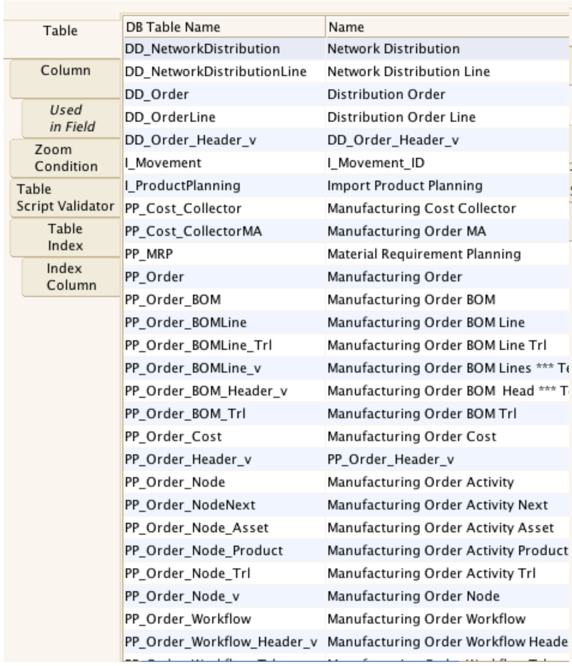
□ Update System Maintained Application Dictionary

Date Processed: Sep 17, 2013 6:10:09 AM MYT

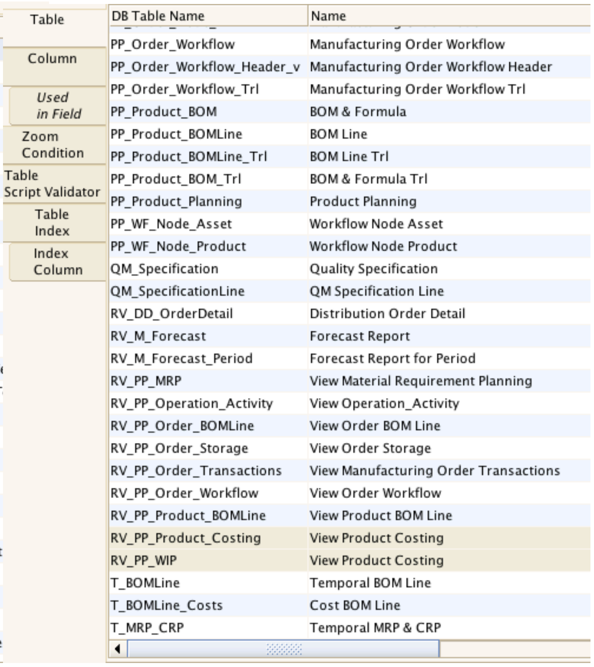
Process Message: Processed=6125 Un-Resolved=0

Legacy Tables

Similar to the Callout reference, I also liked to dump the legacy AD table definition which is associated with the EntityType for LiberoMFG or 'EE01'. Below are the screen-shots of Table records when searched for that EntityType prior to PackOut.



DB Table Name	Name
DD_NetworkDistribution	Network Distribution
DD_NetworkDistributionLine	Network Distribution Line
DD_Order	Distribution Order
DD_OrderLine	Distribution Order Line
DD_Order_Header_v	DD_Order_Header_v
I_Movement	I_Movement_ID
I_ProductPlanning	Import Product Planning
PP_Cost_Collector	Manufacturing Cost Collector
PP_Cost_CollectorMA	Manufacturing Order MA
PP_MRP	Material Requirement Planning
PP_Order	Manufacturing Order
PP_Order_BOM	Manufacturing Order BOM
PP_Order_BOMLine	Manufacturing Order BOM Line
PP_Order_BOMLine_Trl	Manufacturing Order BOM Line Trl
PP_Order_BOMLine_v	Manufacturing Order BOM Lines *** T
PP_Order_BOM_Header_v	Manufacturing Order BOM Head *** T
PP_Order_BOM_Trl	Manufacturing Order BOM Trl
PP_Order_Cost	Manufacturing Order Cost
PP_Order_Header_v	PP_Order_Header_v
PP_Order_Node	Manufacturing Order Activity
PP_Order_NodeNext	Manufacturing Order Activity Next
PP_Order_Node_Asset	Manufacturing Order Activity Asset
PP_Order_Node_Product	Manufacturing Order Activity Product
PP_Order_Node_Trl	Manufacturing Order Activity Trl
PP_Order_Node_v	Manufacturing Order Node
PP_Order_Workflow	Manufacturing Order Workflow
PP_Order_Workflow_Header_v	Manufacturing Order Workflow Heade



DB Table Name	Name
PP_Order_Workflow	Manufacturing Order Workflow
PP_Order_Workflow_Header_v	Manufacturing Order Workflow Header
PP_Order_Workflow_Trl	Manufacturing Order Workflow Trl
PP_Product_BOM	BOM & Formula
PP_Product_BOMLine	BOM Line
PP_Product_BOMLine_Trl	BOM Line Trl
PP_Product_BOM_Trl	BOM & Formula Trl
PP_Product_Planning	Product Planning
PP_WF_Node_Asset	Workflow Node Asset
PP_WF_Node_Product	Workflow Node Product
QM_Specification	Quality Specification
QM_SpecificationLine	QM Specification Line
RV_DD_OrderDetail	Distribution Order Detail
RV_M_Forecast	Forecast Report
RV_M_Forecast_Period	Forecast Report for Period
RV_PP_MRP	View Material Requirement Planning
RV_PP_Operation_Activity	View Operation_Activity
RV_PP_Order_BOMLine	View Order BOM Line
RV_PP_Order_Storage	View Order Storage
RV_PP_Order_Transactions	View Manufacturing Order Transactions
RV_PP_Order_Workflow	View Order Workflow
RV_PP_Product_BOMLine	View Product BOM Line
RV_PP_Product_Costing	View Product Costing
RV_PP_WIP	View Product Costing
T_BOMLine	Temporal BOM Line
T_BOMLine_Costs	Cost BOM Line
T_MRP_CRP	Temporal MRP & CRP

Testing from Eclipse

The same FitNesse testing can also be done from Eclipse IDE itself. This is useful when errors are detected and you want to trace the cause of it. By running the plugins and starting them from the console of the Eclipse has no difference for the Fitnesse project to access it as the OSGi services are mapped via the browser URI. You can read up more on this cool iDempiere capability from its creator Low Heng Sin which wrote about in the wiki for the FIT version done by Carlos Ruiz: http://wiki.idempiere.org/en/NF001_Fitnesse_Integration. From there I was able to use its SLiM settings which Heng Sin has included into the plugin.

Contributing Back

The 'free as in freedom' principle of Open Source gives you an opportunity as proven here where all the code expose allows you to learn as well as review further and extend with new or better fixing of the understanding behind this big module. Whenever you have positive feedback do contribute via the respective forums or write to me personally at red1@red1.org