Introduction to Bonita BPM

Mario G.C.A. Cimino, DII, University of Pisa

Business Process Management System

- ✓ Bonita BPM 7 is a powerful application platform for building personalized, process-based business applications that adapt to your business changes in real time.
- ✓ Bonita BPM has two parts: the development environment, Bonita BPM Studio, and the runtime environment, Bonita BPM Platform.
- ✓ Bonita BPM adopts the model-driven approach, a software design methodology for the development of software systems, launched by the Object Management Group (OMG) in 2001.
- ✓ With model-driven engineering, specifications are expressed as models. Models can be expressed with standards, such as the executable Unified Modeling Language (UML), and the BPMN.
- ✓ Models are then processed to automatically generate software. Code generation means that an automated tool derives from the models parts or all of the source code for the software system.

Business Process Management System



Business Process Management System

16 of 85



BP Modeling: Web purchase model with Signavio

1. Download the Bonita BPMS from

http://docenti.ing.unipi.it/m.cimino/_sw/BonitaBPMCommunity-7.5.4.zip http://docenti.ing.unipi.it/m.cimino/_sw/jdk8.zip

2. Extract it to c:\pmi

3. If needed, change the JDK: create a batch file (go.bat)
set JAVA_HOME=C:\pmi\jdk8
set PATH=C:\pmi\jdk8\bin;%PATH%
java -version
BonitaBPMCommunity64.exe

BP Management: Web purchase example with Bonita BPM18 of 85

1. Select New from the Cool bar to create a new diagram

Edit	en Save Pint Import Export Copy Past	UI Designer Configure Run Debug Portal Preview	Preferences Help Welcome
	New Diagram Open a Diagram	Recently Modified	Bonita BPM 7.3.3 Community
ICALI	Videos Bonita BPM Camp Getting started tutorial UI Designer deep dive Bonita BPM 7 Contract	Documentation Image: Search Documentation Image: Q • Bonita BPM Overview • Getting Started • Process Modeling	Training • English • Español • Français • Private session: contact us
CSUURCES	Examples • Expense report • Vacation management • Procurement	Community Forums Shared projects Translate in your language Report an issue	Blog • Oct 10, 2016 - Bonita BPM 7.3.3 is here! • Sep 27, 2016 - Creating financial business value with BPM • Sep 13, 2016 - User adoption and efficiency - engaging U • Sep 5, 2016 - Re-engineering how firms remain

Bonita BPM: Diagram and Pool Name

🖰 *MyDiagram (1.0) 🛛 Step1 Employee lane Choose a new name and versi Start1 <u>P</u>0 Diagram Web Purchase Diagram Name 1.0 ersion Poc Version 1.0 Web Purchase Name ОK 📡 🔍 🗨 🍺 🥡 💾 🌶 General 🕺 📕 Data 🗜 Execution 😻 6.x Application 🥂 Appearance 🗾 Simulation 📀 Validation status Overview A MyDiagram Diagram Diagram 🔻 lane MyDiagram Edit... Name rm Mapping INTERNAL Version 1.0 ontract nlication (application

Bonita BPM: Draw the BPMN model and set names 20 of 85

3. Create the diagram using the toolkit, configure the selected element



2. Click outside the pool, click on Edit, Enter Diagram and Pool Name

Bonita BPM: Connect a sub-process to a pool



Bonita BPM: Add Process variables

22 of 85

5. Select the Web Purchase Pool, go to Data Pane, on Process variable



Bonita BPM: Add enumeration process variables

Process variables: can be used in a process and until the process instance is completed.

6. Enter *customerName*, leave Data type *Text*, and press *Finish&Add* enter *customerEmail*, leave Data type *Text*, and press *Finish&Add* enter *creditCardNumber*, Data type *Integer*, press *Finish&Add* enter *expirationDate*, Data type *Date*, press *Finish&Add* enter *confirmation*, Data type Boolean, press *Finish&Add* enter *products*, click on *List of options*, Name: *PromotionalProducts*, Options: *TV*, *Mobile phone e Japtop*.



Bonita BPM: set expressions for the exclusive transitions 24 of 85



Bonita BPM: configure a message connector for a task 25 of 85



Add..

Cancel

Finish

Select an actor Employee actor -- This is an example of actor that is mapped to any ACME users

Employee - Add

verview

OVEE

ORDER IS NEEDED

🥖 General 🕮 📕 Data 🗜 Exec

Actors *

CUSTOMER

Pool

Actors

- Auu
- Name: employee
- Finish

Bonita BPM: Map actors to people, forms for human tasks27 of 85

Mapping Actors –people

- Click on Configure on the cool bar.
- Select *customer* in Actor mapping > *Users* > *anthony.nichols (pwd bpm)* Similarliy
- Click on Configure on the cool bar.
- Select *employee* in Actor mapping > Users > april.sanchez (pwd bpm)

Forms and Data Objects

- Select the Web Purchase Pool > Tab Execution
- > Tab Instantiation Form
- ≻ 6.x
- 6.x Application
- Add
- ≻ ...



Bonita BPM: forms for human tasks

Forms and Data Objects

- Select the Process Variables customerName, customerEmail, products.

- A default form is created

• 1			
Customer	Name	Ι	
Customer	Email	Ι	
Products			
	Submit		
	Submit		
	Submit	🖉 General 23 🕅 Appearance 📀 Validation status 🗈 6.x Previe	4
· · · · · · · · · · · · · · · · · · ·	Submit	General SI M Appearance ♥ Validation status □ 6.x Prevent WEBPURCHASE	<i>и</i>
rator ASSIGNMENT	Submit	General 32 M Appearance Validation status (), 6,x Preview WEBPURCHASE General General Actions	a
rator ASSIGNMENT reasion customerEmai	Submit	General General General General Actoms Valdators Name WEBPURCHASE	<i>a</i>
rator ASSIGNMENT ression field_customerEmail reason ASSIGNMENT	Submit	General Si M Appearance Si Validation status G. x. Preview WEEPURCHASE General Actions Yaldators Name WEEPURCHASE Show page table I WEEPURCHASE	« •) () [
rator ASSIGNMENT ression customerEmail ression field_customerEmailt rator ASSIGNMENT PromotionalProducts values Empty list		✓ General IM Appearance ✓ Validation status G.x. Prever ✓ WEBPURCHASE General General General Actions Name WEBPURCHASE Show page tide I WEBPURCHASE Description	» • • • • • •

Enter the name and description for the form.

🖌 Add form

Enter the name and description of the form. Select data to show and update in the form. Click on t ϵ to switch between Process data and Business data. Choose the type of widget, depending on the ϵ				
Name WEB PURCHASE				
Bosiness variables Process va	iriables		Mandahara	- Paradaraha (
Vame CustomerName CustomerEmail creditCardNumber expirationDate confirmation V products	Text field Text field Text field Date Checkbox Radio buttons	+ + + + + +		

Activity Sales review

- Tab Execution > Tab Form > 6.x Application > Add Select customerName, customerEmail, and products as read only; finally add confirmation Activity Pay
- Tab General > task type: Human
- Tab Execution > Tab Form >
 6.x Application > Add > Select
 creditCardNumber, and expiration Date.

- Click the Run button in the Cool bar
- Open two different browsers and point to http://localhost:8080/bonita/login.jsp
- First browser > customer login > username: *anthony.nichols* password: *bpm*
- Second browser > employee login > username: *april.sanchez* password: *bpm*



Bonita BPM: use case

30 of 85

SUBMIT

- Select the t	ask and press <i>take</i> - first case: press on SUBMIT (without confirmation) > you will receive an email
& Bonitasoft	Welcome: April Sanchez + 🕑 User + Settings
	I Tasks ∞ Cases 0 Processes
<	Filters > Form Comments Overview
To do	Process All- Search Q SALES REVIEW
My tasks	In task name column Customer Name
Done tasks	Task list C Customer Email
	L TAKE & RELEASE 1-1/1 O mario.cimino@unipi.it
	Confirmation
	SALES REVIEW WEB PURCHASE Dec 04 12:48 PM Products
	1 - 1/1 TV mobile phone
	laptop
	SUDM

- As a customer, start a new process in the first browser
- Fill again the customer form and submit
- As an employee, check the confirmation flag and submit

S Bonitasoft		Welcome: Anthony Nichols - Q User
	I 🗹 Tasks ∞ Cases 🛛 🖓 Processes	
<	Filters	Form Comments Overview
īo do 🛛 🖪	Process All - Search C	2 PAY
/ly tasks 🛛 🕚	in task harrie corum	Credit Card Number *
)one tasks	Tack list C	1234567
		Expiration Date *
	TAKE 🚣 RELEASE 1-1/1	2016 December 9 10
	🗖 💄 Task name 🔺 Process name 🛛 Due date	B SUBMIT
	PAY WEB PURCHASE Dec 04 1:09 PI	M
	1-	1/1

- As an employee, you can now see in *done tasks* the task history

Bonita BPM: Database and Web Service connectors 32 of 85



- 1. Create the diagram above (for detailed steps see the first tutorial):
- 2. New Diagram > complete the flow with the toolkit leaving the default task types.
- 3. Select *Step1 > General* Tab *> Task type:* Service.
- 4. Select *Step2* > *General* Tab > *Task type:* Human.
- 5. Click on *Save* in the cool bar.
- 6. Create the process variables:
- 7. Select *Pool > Data* Tab *> Process Variables: Add >* Name: *customer > Finish & Add >* Name: *deposit > Finish*

8. Create the pool form

- 9. Select *Pool* > Tab *Execution* > Instantiation form > 6.x
- 10. Tab 6.x Application > Add > Select Tab Process variables > Select deposit, and mandatory > Finish

11. Create the Step2 form

- 12. Select *Step2* > Tab *Execution* > form > 6.x
- 13. Tab 6.x Application > Add > Select Tab Process variables > Select customer, and read only > Finish

Bonita BPM: Database connector

14. Create the MySQL Database:

- 15.1st method: import the file *bank-dump.sql* into a MySQL server.
- 16.2nd method: download the file www.iet.unipi.it/m.cimino/wdis/res/dbms.zip and extract it on C:\wdis. Finally, click on C:\wdis\mysqlStart

17. Access the Database with MySQL client:

- 18. Click on C:\wdis\mysqlClient6.1 > Click on the "+" icon close to MySQL connections > enter a name and click OK.
- 19. Select the *bank* schema > *Tables* > *account* > right click > Select rows.

20. Create the DB Connector:

21.On Bonita, select *Step1* > Tab *Execution* > Connectors out (*) > *Add* > Categories: *Database Others* > *Connector definition* > *MySQL* 5.5 JDBC 4... > Next

22.Name: *dbconn1 > Next.* Enter URL: *jdbc:mysql://localhost:3306/bank* Username: *root* Password: *Next*

(*) *Connectors out* are carried out at the end of the step, whereas *Connectors in* at the begin of the step.



Bonita BPM: Database connector

34 of 85

Connectors	🔓 MySQL 5.5 JDBC 4 database query (1.0.0)
Select a connector definition Select a connector definition from the list below	Database access information Enter the database access information
Categories Connector definitions Image: CMS Source Image: CMS Source Image: CMS Microsoft SQLServer Image: Concele Microsoft SQLServer Image: CAR Microsoft SQLServer Image: Concele Image: Concele Image: CAR Microsoft SQLServer Image: CAR Image: CAL Image: CAL Teradata 14 JDBC 4 database query Image: CAL Image: CAL Image: CAL <t< td=""><td>Driver * i com.mysql.jdbc.Driver URL * i jdbc:mysql://localhost:3306/bank Username i root Password i Image: state of the sta</td></t<>	Driver * i com.mysql.jdbc.Driver URL * i jdbc:mysql://localhost:3306/bank Username i root Password i Image: state of the sta
Definition version 1.0.0	6 Connectors Image: Connector state Output operations definition Image: Connector outputs and store them in process or business variables Retrieve connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables Image: Connector outputs and store them in process or business variables
23. Enter the query	Ad <back next=""> Finish Cancel E denosit > \$ (denosit):</back>
(for autocompletion of variables pr 23. Select Next > Scripting Mode > Ne	ress CTRL + SPACE) ext > Select target: customer

24. Click on the pencil icon to open the Groovy editor.

Bonita BPM: Database connector

Edit expression 28. Expression type: Script Expression type 29. In the text area enter Name * resultset Interpreter GROOVY Connector output π Constant Select a variable... Select a provided variable.. • if (resultset.next()) Parameters Script return resultset.getString("customer"); if (resultset.next()) -📋 Variable return resultset.getString("customer"); else else return "none"; return "none"; T T 4 28. Click on OK > Finish. Evaluate 29. Click on Start button in the coolbar, ☑ Automatic dependencies resolution 30. The Bonita launches the bowser Return type java.lang.String Browse... 31. Enter a deposit and SUBMIT ОК Cancel 32. At Step 2, a customer with more than the deposit will be shown Q **,** 65 2 -4 Import Export CODY UI Designer Configure Run Debua Portal Print Preferen n M iagram (1.0) 🛛 🗾 Pool 🛛 💆 Step2 **G** Bonitasoft & Bonitasoft 8 Step2 Step1 \cdot Employee lane Start1 Pool End1 8 Step2 I <p Deposit * Customer 250 ethan@instanbul.tr 😑 🗖 📝 General 🛢 Data, 🗜 Execution 🕱 😻 6.x Application 🕂 Appearance 🗾 Simulation 🤮 Overview Step1 Connectors out -SUBMIT1 -SUBMIT1 Connectors in lane Operations Connectors out Add... 💫 dbconn1 -- database-mysql (1.0.0) -- ON_FINIS

Bonita BPM: Database and Web Service connectors

1 of 24



- 1. Create the diagram above (for detailed steps see the first tutorial):
- 2. New Diagram > complete the flow with the toolkit leaving the default task types.
- 3. Select Step1 > General Tab > Task type: Service.
- 4. Select *Step2 > General* Tab *> Task type:* Human.
- 5. Click on Save in the cool bar.

6. Create the process variables:

- 7. Select *Pool > Data* Tab *> Process Variables: Add >* Name: *customer > Finish & Add >* Name: *deposit > Finish*
- 8. Create the pool form
- 9. Select *Pool* > Tab *Execution* > Instantiation form > 6.x
- 10. Tab 6.x Application > Add > Select Tab Process variables > Select deposit, and mandatory > Finish

11. Create the Step2 form

- 12. Select *Step2* > Tab *Execution* > form > 6.x
- 13. Tab 6.x Application > Add > Select Tab Process variables > Select customer, and read only > Finish

Bonita BPM: Database connector

14. Create the MySQL Database:

- 15.1st method: import the file *bank-dump.sql* into a MySQL server.
- 16.2nd method: download the file www.iet.unipi.it/m.cimino/wdis/res/dbms.zip and extract it on C:\wdis. Finally, click on C:\wdis\mysqlStart

17. Access the Database with MySQL client:

- 18. Click on C:\wdis\mysqlClient6.1 > Click on the "+" icon close to MySQL connections > enter a name and click OK.
- 19. Select the *bank* schema > *Tables* > *account* > right click > Select rows.

20. Create the DB Connector:

21.On Bonita, select *Step1* > Tab *Execution* > Connectors out (*) > *Add* > Categories: *Database Others* > *Connector definition* > *MySQL* 5.5 JDBC 4... > Next

22.Name: *dbconn1 > Next.* Enter URL: *jdbc:mysql://localhost:3306/bank* Username: *root* Password: *Next*

(*) *Connectors out* are carried out at the end of the step, whereas *Connectors in* at the begin of the process.



Bonita BPM: Database connector



- 23. Enter the query
- 24. SELECT * FROM account WHERE deposit > \${oeposit}; (for autocompletion of variables press CTRL + SPACE)
- 23. Select Next > Scripting Mode > Next > Select target: customer
- 24. Click on the pencil icon to open the Groovy editor.

Bonita BPM: Database connector



Example of Web service:

http://www.thomas-bayer.com/axis2/services/BLZService?wsdl

- 1. Install the SOAP UI tool:
- WIN64: http://www.iet.unipi.it/m.cimino/sse/res/SoapUI-x64-5.2.1.exe WIN32: http://www.iet.unipi.it/m.cimino/sse/res/SoapUI-x32-5.2.1.exe MACOS: http://www.iet.unipi.it/m.cimino/sse/res/SoapUI-5.2.1.dmg LINUX: http://www.iet.unipi.it/m.cimino/sse/res/SoapUI-x64-5.2.1.sh
- 3. Right click on Projects

 New SOAP Project Initial WSDL: (enter the URL) OK 4 Expand > dbl click 	File Project Suite Case Step Tools Desktop Help Image: Solution of the state of th	
S SoapUI 5.2.1	S New SOAP Project	×
File Project Suite Case Step Tools	New SOAP Project Creates a WSDL/SOAP based Project in this workspace 2	\$
Empty SOAP REST In port Sa	Project Name: www.thomas-bayer	
Projects	Initial WSDL: ttp://www.thomas-bayer.com/axis2/services/BLZService?wsdl Browse	
R BLZServiceSOAP 1Binding	Create Requests: 🗹 Create sample requests for all operations?	
	Create TestSuite: Creates a TestSuite for the imported WSDL	
B	Relative Paths: Stores all file paths in project relatively to project file (requires save)	
Request 1	OK Car	ncel

Bonita BPM: Web Service connector

6 of 24

- The service takes the BLZ bank code (used in Germany/Austria, ABI+CAB in Italy, incorporated into the IBAN as part of SEPA standardization) as an input
- Example:

54030011 the BLZ of the Bank *Service Credit Union Overseas Headquarters* https://bank-code.net/blz-sort-codes/54030011-service-credit-union-overseas-headquarters-051749

bank-code.net/blz-sort-codes/54030011-service-credit-union-overseas-headquarters-051749		
BLZ Sort Code Details		
BLZ Code / Sort Code	54030011 The banking institution's BLZ sort code	
Bank	Service Credit Union Overseas Headquarters	
Money Transfer	Save on international money transfer fees by using TransferWise, which is up	
Branch	Service Credit Union Branch / business name of service payment provider. This name and the town should be specified in the beneficiary data on invoices and forms.	
BIC / Swift Code	SCRUDE51XXX The banking institution's swift code also known as Business Identifier Code (BIC) .	
City	Sembach	
Zip / Postal Code	67681	

Enter the code and click the play icon ())

<pre><soap:envelope xmlns.blz="http://thomss.blyop.com/blz(" xmlns:soap="http://www.w3.org/2003/05/soap-envelope"></soap:envelope></pre>
thins:biz="http://thomas-bayer.com/biz/">
<soap:header></soap:header>
<soap:body></soap:body>
<blz:getbank></blz:getbank>
<blz:blz>54030011</blz:blz>

The service provides the following details: bank name (ns1:bezeichnung), BIC code (ns1:bic), place (ns1:ort), and postal code (ns1:plz)

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <nsl:getBankResponse xmlns:nsl="http://thomas-bayer.com/blz/">
        <nsl:getBankResponse xmlns:nsl="http://thomas-bayer.com/blz/">
        <nsl:details>
        <nsl:bezeichnung>Service Credit Union Overseas Headquarters</nsl:bezeichnung>
        <nsl:bic>SCRUDE51XXX</nsl:bic>
        <nsl:bic>SCRUDE51XXX</nsl:bic>
        <nsl:ort>Kaiserslautern</nsl:ort>
        <nsl:plz>67661</nsl:plz>
        </nsl:details>
        </nsl:details>
        </nsl:details>
        </nsl:getBankResponse>
        </soapenv:Body>
    </soapenv:Envelope>
```

Bonita BPM: Web Service connector

8 of 24

- 1. Remove the DB connector
- 2. Select Step1 > Tab Execution > Connectors out > Remove
- 3. Remove the Process Variables
- Select *Pool* > Tab *Data* > Process variables > select *customer* > Remove > OK, select *deposit* > Remove > OK.
- 5. Add the process variables bankCode (blz), bankName (bezeichnung)
- 6. Add > Name: bankCode > Finish&Add > Name: bankName > Finish
- 7. Update the Pool form
- 8. Select *Pool* > 6.x Application > Pageflow > Select *Pool* > *Remove. Add* > *Process* variables > Select bankCode. Press *Finish.*



9. Update the Step2 form

- 10. Select Step2 > 6.x Application > Pageflow > Select Step2 > Remove. Add > Process variables > Select bankName > Finish
- 11. Add the WS connector
- 12.Select Step1 > Execution > Connectors out > Add.
- 13. Categories: SOAP WebService > Web Service Soap1.2 > Choose the NAME > conn2 > Next

```
45. Name: wsconn2 > Next > Enter parameters *
   Service NS: http://thomas-bayer.com/blz/
   Name: BLZService
   Press Next
   Port Name: BLZServiceSOAP12Binding
   EndPoint: http://www.thomas-bayer.com/axis2/services/BLZService
   Binding: http://www.w3.org/2003/05/soap/bindings/HTTP/
   Envelope:
   <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"</pre>
                   xmlns:blz="http://thomas-bayer.com/blz/">
      <soap:Header/>
      <soap:Body>
         <blz:getBank>
            <blz:blz>${bankCode}</blz:blz>
         </blz:getBank>
      </soap:Body>
   </soap:Envelope>
```

46.Next > Next > Returns body > Next > Output operations: (Ctrl + space to find parameters values)

(*) Parameters are extracted by WSDL document http://www.thomas-bayer.com/axis2/services/BLZService?wsdl and by using a SOAP client software such as *SoapUI*.

Bonita BPM: Web Service connector

10 of 24

47.Next > Next > Returns body > Next > Output operations:48.Select *bankName* on the left. Click on the pencil icon on the right. Edit Expression:



49.In the text area (Ctrl + space to select parameters values if needed):

```
import org.w3c.dom.*;
responseDocumentBody.normalizeDocument();
NodeList nl = responseDocumentBody.getElementsByTagName("ns1:bezeichnung");
Element el = (Element) nl.item(0);
return el.getTextContent();
```

👉 Edit expression	
Expression type Connector output Constant Parameters	Name * responseDocumentBody Interpreter GROOVY Select a variable Select a provided variable
Script Uariable	<pre>import org.w3c.dom.*; responseDocumentBody.normalizeDocument(); NodeList nl = responseDocumentBody.getElementsByTagName("ns1:bezeichnung"); Element el = (Element) nl.item(0); return el.getTextContent();</pre>

- 49. Click on Start button in the coolbar
- 50. The Bonita launches the browser
- 51. Enter Bank Code and SUBMIT
- 52. At Step 2, the Bank Name is shown
- 53. Note: The WS may reply with "-1" when the WS is not available (this my occur for free WS)

