# BTSHOL11: Enabling Business Activity Monitoring

Objectives

In this lab, you will learn how to enable a specific application using BAM for true real-time end to end visibility in the context of a business activity. After completing this lab, you will be able to:

* Define a complete BAM observation model (Business Analyst)
* Deploy BAM observation model (IT Professional)
* Map BAM observation model to implementation (Developer)
* View KPI’s and create business alerts on them (Business User)

Scenario

In this lab, the scenario that will be used is a typical order management business process. The goal of end-to-end visibility (that BAM enables) is to empower the business user to make real-time business decisions based on the current KPI status of the running business. But in order to achieve this end goal there are many other roles that play an important part in “BAM enabling” the application.

Each exercise in this lab below corresponds to the following job roles:

* *Business Analyst* – First the business analyst will define the order management business activity (the observation model) and view (the aggregations and pivot table) inside of Excel.

Estimated time to complete this lab: 45 minutes

* *IT Pro* - The observation model can now be sent to the IT Pro who will then deploy the observation model using the BAM management command line utility. The management utility takes the observation model and dynamically generates all of the BAM infrastructure (tables, DTS packages, stored procedures, cubes, etc.) needed to correlate and manage the business events in the context of the activity.
* *Developer* - The developer can now use the tracking profile editor to view the business activity (previously deployed by the IT Pro) and proceed to map each business event and piece of data to the implementation (orchestration schedules or pipelines natively from tool or use BAM API for custom applications). Everything is now in place and once real events start firing as the business activity comes on-line the business end user can go to the BAM portal and see the live information. They can also set custom business alerts on their key performance indicators to be proactive in order to keep a close eye on the over real-time trends of the business.

User Name: **Administrator**

Password: **pass@word1**

Exercise 1  
Familiarize yourself with the solution

Through the rest of this lab you using a simple order approve solution.

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Configure the starting lab environment. | 1. Click Start, click Microsoft Visual Studio 2012.   The Visual Studio 2012 splash screen appears and then Visual Studio starts.   1. In Visual Studio, from the **File** menu, click **Open**, and then click **Project/Solution**. 2. In the Open Project dialog box, browse to **C:\Labs\Lab 11\Start** and select Lab11.sln. Click **Open**. 3. The solution appears in the Solution Explorer and the. Double-click the OrderApprovalProcess.odx to open the order process orchestration.      1. The **Receive Order** shape receives an order message 2. The **Send For Approval** shape initiates the **OrderIdCorrelation** correlation set on the Order/OrderId, and sends the message out for approval. 3. The **Receive Approval** shape is following the correlation set, and receives the approved (or declined) Order. 4. The **Approved** branch checks if the Order is approved: ApprovedOrder.Approved == true 5. The Approved/Denied Order is sent out. |
| Build and Deploy | 1. In the Solution Explorer, right-click the *Lab11* project and select **Deploy***.*In the Output window,make sure the solution was deployed without errors. 2. On the **Start** menu, click **BizTalk Server Administration.** 3. In the Biztalk Server 2016 Administration Console, select **Biztalk Server 2016 Administration > BizTalk Group > Applications > Lab11** 4. Right-click the Lab11 application and select **Import > Bindings** 5. In the **Import Bindings** dialog, browse to **C:\Labs\Lab 11\Start\Lab11\Lab11.BindingInfo.xml**, and click **Open.** 6. Continue to open the **Lab11 > Orchestrations** node and right-click the Lab11.OrderApprovalProcess orchestration, and select **Properties.** 7. In the Orchestration Bindings dialog, select **Bindings** in the left pane, and set the port bindings according to the table below:  |  |  | | --- | --- | | Inbound Logical Ports | Receive Ports | | ReceiveOrderPort | NewOrder | | ReceiveApprovalPort | ApprovedOrder | |  |  | | **Outbound Logical Ports** | **Send Ports/Send Port Groups** | | SendOrderForApprovalPort | SendOrderForApproval\_FILE | | SendApprovedOrderPort | SendApprovedOrder\_FILE | | SendDeniedOrderPort | SendDeniedOrder\_FILE |  1. Set the Host to **BizTalkServerApplication.** 2. Start the **Lab11** Application |
| Test the solution | 1. In Windows Explorer, navigate to **C:\Labs\Work\Lab 11\FileDrop** and **copy** the Order\_500.xml file to the **New Order** folder. 2. As the files are picked up, browse to the **C:\Labs\Work\Lab 11\FileDrop\Waiting For Approval** folder. Move the {Guid}.xml file to the **Approved** folder. 3. Browse back to the **FileDrop** folder where there now should be a **APPROVED\_{Guid}.xml** file. 4. Repeat the steps above using the Order\_1000.xml file, which should result in an **DENIED\_{Guid}.xml** file in the FileDrop folder. |

Exercise 2  
Defining a Business Activity (Business Analyst)

In this exercise you will take on the role of a *Business Analyst* and will complete the creation of the Order Management activity. You will identify key business milestones and the data of interest. You will conclude this exercise by exporting the business activity, including the business milestones, to be used to identify the Key Performance Indicators using Excel.

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Create the Order Activity and View. | 1. On the **Start** menu, point to **Microsoft Office Excel 2016.** 2. Enable the BAM AddIn in Excel by going to File tab (top left corner) – Excel Options – AddIns – Excel Addins [Go] – and select Business Activity Monitoring. 3. Click the Add-Ins tab on the top, then click the **BAM** drop-down box and select **BAM Activity**   Untitled   1. In the **Business Activity Monitoring Activity Definition** dialog click the **New Activity** button. Type **OrderActivity** in the Activity name text box. 2. Create the following OrderActivity Activity items by clicking the **New Item** button:  |  |  | | --- | --- | | Item Name | Item Type | | ReceivedOrder | Milestone | | Customer | Text (200) | | OrderTotal | Integer | | SentForApproval | Milestone | | ApprovalResponse | Milestone | | Approved | Milestone | | Declined | Milestone |  1. Click the **Ok** button to create the Activity 2. In the **Business Activity Monitoring View Creation** dialog click **Next** and then **Next** again. 3. In the **BAM View** dialog, type **OrderView** in the View name text box. Check the **Select all activities**  checkbox and click **Next**. 4. Select all Activity Items, and click **Next**. 5. Add a new Group by clicking **New Group**. Set the **Business milestone alias** to **Finished** – include **Approved** and **Declined**.   As the process can end in either the Approved- or the Denied shape we need to group these two events in order for us to measure the total time.   1. Add a new Duration by clicking **New Duration**. Set the Duration name to **ProcessDuration** – between *ReceivedOrder* and *Finished*. Set the Time resolution to **Second**.   In reality we’d probably not be interested to measure the process in seconds.   1. Add a new **Duration**. Set the Duration name to **ApprovalDuration** – between **SendForApproval** and **ApprovalResponse**. Set the Time resolution to **Second**. 2. Click **Next**. 3. Add a new Dimension by clicking the **New Dimention** button. Set the Dimension name to **CustomerDimension** (Data Dimension) using Customer. 4. Add another Dimension, set the Dimension name to **ReceivedDimension** (Time Dimension) using *ReceivedOrder*, down to the minute (Year, quarter, month, day, hour, minute). 5. Add a Measure by clicking the **New Measure** button. Set the Measure name to **OrderTotalMeasure** (Measure) using *OrderTotal*. Leave the Aggregate type to **Sum.** 6. Click **Next > Next > Finish** to compete the process. 7. From the **PivotTable Field List** (on the right pane), drag the **Customer Dimension** to the area on the left. 8. Drag **Received Dimension** to the top area. 9. Drag **OrderTotalMeasure** to the middle area. 10. Mark the Pivot table as **Real Time Aggregation** by clicking the **OrderTotalMeasure** cell (B2) and click the Real Time Aggregation button in the Add-ins tab:   Untitled   1. Save the workbook as **OrderActivity.xlsx** in the **C:\Labs\Work\Lab 11\Activities** folder. |

Exercise 3  
Deploying the BAM Observation Model (IT Professional)

In this exercise, you will take on the roll of an **IT Professional**. You will deploy the BAM tracking configuration created in the previous exercise. To accomplish this, the IT Professional uses the BAM management utility (bm.exe) to generate the BAM infrastructure. The BAM Management utility consumes the BAM definition that was created in the previous two exercises. The BAM definition XML which was exported from the Excel Workbook will be deployed using the BAM Management utility. The BAM Management utility can be told where to deploy the infrastructure, such as the server name, database name, and other database settings.

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Deploying the BAM Observation Model (IT Professional) | 1. In Windows Explorer, navigate to **C:\Labs\Work\Lab 11\Activities** and double-click the **BM** shortcut. 2. In the **Command** window, type:   **BM deploy-all -DefinitionFile:"C:\Labs\Work\Lab 11\Activities\OrderActivity.xlsx"** and then press **ENTER**.  If you get an exception saying: “ERROR: Failed to open BAM Excel workbook file…” you need to set your regional settings to English (United States), and try again.   1. Examine the **C:\Labs\Work\Lab 11\Activities** folder. There should now have be a **OrderActivity\_LiveData.xlsx** created. 2. Close the **Command** window. |
| View created tables and views | 1. Click Start, click **SQL Server Management Studio.** 2. Connect to the server. Make sure you use the settings as below:      1. In the object explorer, expand **Databases > BAMPrimaryImport > Tables.** Expand the **dbo.** **bam\_OrderActivity\_Completed > Columns**.Except from the *ActivityId*  and the *LastModified* columns the table columns are the same as your Activity Items**.** Leave the SQL Server Management Studio open, as you’ll use it later. |

Exercise 4  
Mapping a BAM Activity to the Implementation (Developer)

In this exercise, you will map the BAM activity (which you completed by adding milestones and data to the process visualization in exercise #2) to the actual running process itself. To accomplish this, you will use BizTalk’s Tracking Profile Editor (TPE).

The TPE is a developer tool used to create or edit tracking profiles. Tracking profiles control the behavior of run-time components which intercept and store data. Changes in visibility requirements can be applied at any time by simply updating the tracking profile. Because they can be modified at any time without impact upon the running solution, tracking profiles are an important point of flexibility for process management.

In the following tasks you will launch TPE tool, load the activity, and then interactively load the various event sources and mappings between them and the target items in the BAM activity definition.

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Open the Tracking Profile Editor (TPE) | 1. On the Start menu, click Tracking Profile Editor. |
| Select the target BAM activity | 1. In the left pane, click the “Click here to import a BAM Activity Definition” link. 2. In the **Import BAM Activity Definition** dialog, select the **OrderActivity**. 3. In the right pane, click the “Click here to select an event source” link. 4. In the **Select Event Source Parent Assembly** dialog, Select the **Lab11** assembly, and click **Next**. 5. Select the **Lab11.OrderApprovalProcess** and click **Next.** |
| Map Activity Items to Event sources | 1. Proceed to map the Event source to the Activity items by dragging the **Receive Order** shape from the orchestration to the **ReceiveOrder** Activity Item on the left. 2. Continue mapping the following Event Sources and Activity Items.  |  |  | | --- | --- | | Event Source | Activity Item | | Send For Approval | SentForApproval | | Receive Approval | ApprovalResponse | | Send Order | Approved | | Declined | Declined |  1. Right-click the **Receive Order** shape in the Event source pane, select **Message Payload Schema**. Expand the schema and drag the **CustomerName** node to the **Customer** Activity Item in the Activiity pane. 2. Drag the **TotalSum** to the **OrderTotal** |
| **Deploy the mapping you just built in the TPE**. | 1. In the Tracking Profile Editor, from the **Tools** menu, click **Apply Tracking Profile**, and then click **Yes**. 2. Click **OK** on the confirmation dialog box. 3. Save the Tracking Profile to the Desktop, and then close the Tracking Profile Editor. |
| **Test solution to make sure data is tracked** | 1. In Windows Explorer, navigate to **C:\Labs\Work\Lab 11\FileDrop** and **copy** the Order\_500.xml file to the **New Order** folder. 2. Open the **SQL Server Management Studio** click the **New Query** button in the upper left corner.Type the following query:   SELECT \* FROM [BAMPrimaryImport].[dbo].[bam\_OrderActivity\_**Completed**]   1. Press F5 to execute the query. There should still be no rows in the table.. 2. As the files are picked up, browse to the **C:\Labs\Work\Lab 11\FileDrop\Waiting For Approval** folder. Move the {Guid}.xml file to the **Approved** folder. 3. Browse back to the **FileDrop** folder where there now should be a **APPROVED\_{Guid}.xml** file. 4. Re-run the query (F5) from step **b**. Examine the result. 5. In Windows Explorer, navigate to **C:\Labs\Work\Lab 11\FileDrop** and **copy** the Order\_1000.xml file to the **New Order** folder. 6. In the **SQL Server Management Studio** in the already opened query window, add another query:   SELECT \* FROM [BAMPrimaryImport].[dbo].[bam\_OrderActivity\_**Active**]   1. Press F5 to execute the query**.** Thereshould now be one row. 2. By completing the process and moving the {Guid}.xml file to the **Approved** folder, the bam\_OrderActivity\_**Active** table should now be empty, while there should now be two rows in thebam\_OrderActivity\_**Completed** table. |

Exercise 5  
Consuming BAM Data & the Portal Experience (End User)

In this exercise, you will consume BAM data, interact with BAM alerting features, and then run some purchase orders through the underlying process to experience a typical business end-user interaction with visibility data. To accomplish this, you will use the Business Activity Monitoring (BAM) Portal. The BAM Portal is an out-of-box functionality that ships with Biztalk Server 2016. The portal is targeted at business end-users primarily, though it will no doubt be used by business analysts, IT professionals, and anyone else with a need to view business performance data.

The portal provides functionality such as search, alert creation, and management, and basic data analysis tools through familiar Office UI controls.

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Generate initial inbound flow of purchase orders. | 1. Double-click the file **C:\Labs\Work\Lab 11\FileDrop\More Samples\Run.bat** to create and submit 10 purchase orders to the process. Wait a few seconds before you press **Enter** to approve the orders. 2. After the command window closes, there should be 10 more orders in the FileDrop folder. You can redo this step if you want more data. |
| View the aggregations using Microsoft Excel 2016. | 1. If you have Excel opened, close it. 2. Brows to the C:\Labs\Work\Lab 11\Activities folder, and open OrderActivity\_LiveData.xlsx 3. Generate some more messages using the previous step. 4. Right-click the pivot table, and click “Refresh”. 5. View the result. |
| Launch the BAM Portal and access the role-specific view of this business process  The BAM portal pages are broken into three main sections: 1) banner/branding information across the top; 2) navigation frame called “My Views” on the left; 3) welcome/content in the frame on the right. | 1. Open Internet Explorer and browse to <http://localhost/bam> 2. In the **My Views** navigation pane, expand the **OrderView->Activity Search->OrderActivity** view. 3. In the “Column Chooser”, select all and click the “>>” button. 4. Press the **Execute Query** button, and review the result. |

(*continued*)

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Navigate to the data aggregation called PivotTable1, and drill into the data pivot to show the various stages of processing. | 1. In the **My Views** navigation pane, expand the **OrderView->Aggregations->Pivottable1**,and thenclick **PivotTabell1**. 2. If you get a dialog box forn the **Microsoft Office Web Components** click OK. 3. Drill down into the Pivottabell1to view received orders by minutes, with a corresponding total figure. |
| Repeat the steps using the the Excel spread. | 1. In Windows Explorer, navigate to **C:\Labs\Work\Lab 11\Activities** open OrderActivity\_**LiveData**.xlsx. 2. Examine the Pivot table, as it is very similar to the one you’ve just seen in the BAM portal. |

(*continued*)

|  |  |
| --- | --- |
| Tasks | Detailed steps |
| Investigate the individual items underlying the aggregation | 1. Right-click the same pivot table cell representing the **OrderTotal**, and select **Show Results** from the context menu.   At this point you are sent to the **Activity Search** page, which will actually display the underlying individual items represented by the aggregate count in the pivot cell you just clicked. The idea here is that you recognize you have a bottleneck in progress (possibly) but need to see the items in that accumulation to know if you **really** have a problem. |
| Refine the data presentation and review the detailed status information for an order on the refined query result list | 1. Expand the **Column Chooser** section of this page, then select all remaining items on the left side and use the **>>** button to move them to the right side (selected columns) box. 2. Collapse the **Column Chooser** section, then hit the **Execute Query** button at the top right of the page to run the query again, this time using an exhaustive set of columns to present to the user. 3. Double click any one of the Purchase Orders displayed.   It is also possible that the user may wish to modify the query itself. Specifically, if the number of items presented by this **Show Results** action is too large, it may be necessary to use some refinement criteria to get the list down to something that you can act upon. For example, of the orders piling up in this processing stage, you may only care about those that have been in that stage for more than X hours. You can refine the query by expanding the section of this page called Query and modifying the set of query clauses you see there. |
| Review the detailed status information for an order on the refined query result list | In looking at the detailed information for this order you suspect that there is a problem with the order, whether it’s some missing data, or the order was processed incorrectly. You begin looking for a reason why. Your investigation options are exhausted, and you need to invoke help from an IT Professional.   1. Click the **Assistance…** button to invoke the **Request Technical Assistance** dialog box. 2. In the **Subject** text box, type **Message Threshold Alert**. 3. In the **Problem Description** box, type **The message threshold has been reached**. 4. Click the **Send Report** button.   This sends an entry to the relevant BTS application event log under the assumption that the IT/Administrator has set an operational alert using Microsoft Operations Manager’s (MOM) event log monitoring capabilities. |