

FUSION MIDDLEWARE BUSINESS ACTIVITY MONITORING

# Event Stream Analytics in Oracle BAM 12.1.3

ORACLE WHITE PAPER | SEPTEMBER 2014



## Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

# **Table of Contents**

Disclaimer	1
Introduction	3
Accessing the Project	3
Prerequisites	3
Importing the Project	4
Running the Project	4
Data Objects	5
Business Queries	6
Business Views	14
Alerts	21
Dashboards	22
Creating a Project	23
Creating a Data Object	23
Business Queries	26
Flat SQL Query	26
Continuous Query	26
Tree Model Query	29
Business Views	30
Table Type View	30
Tree Map Type View	30
Dashboards	31
Accessing the Dashboard	32
Troubleshooting	35

Project Import Issues	35
Business Query Issues	35
Business View Issues	36
Dashboard Issues	37
Data Population Issues	37

## Introduction

This document provides information on using the Oracle BAM Event Stream Analytics sample project in Oracle BAM 12c by elaborating on the BAM Tree Model Query, Active Data Service (ADS) and Continuous Query features. There is some sample data provided to simulate the twitter feeds, which Oracle BAM monitors real time. This data is well-suited for the continuous query to detect patterns from streaming events.

This project depicts a scenario where a user uses Oracle BAM to monitor customer sentiments toward various products by real time monitoring the tweets in real time. The process is as follows:

- 1. Customers tweet their sentiments toward various products.
- 2. Oracle Enterprise R is used to perform text mining to look for certain key words.
- 3. Based on the results from 2, sentiment score (negative or positive) is assigned for a given tweet.
- 4. Oracle BAM monitors these tweets and looks for patterns from various customer responses in two ways
  - a. Monitor customer sentiment toward various products, visualized by 'Tree Map' visualization.
  - b. Check for irate customers by looking for negative tweets from the same customer within a given time interval.

Using the Oracle BAM Tree Model Query and Business Views, you can present feedback for the eight products using color-coded areas. With the dashboard, you can gauge customer sentiments toward various products.

# Accessing the Project

This project uses Oracle BAM Monitor Count Template (this monitors streaming events coming into Oracle BAM and looks for 3 negative/positive tweets for a given customer within a period of time) and Oracle BAM Tree Model Query. Monitor Count Templates and Tree Model Query are new features in Oracle BAM 12c. Monitor Count Template uses CQL and the embedded Oracle OEP Engine to support the query. Tree Model Query uses Oracle ADF technology.

#### Prerequisites

- 1. Copy BAM "samples" directory to ORACLE\_HOME/soa/bam.
- 2. Set environment variable JAVA\_HOME.
- 3. Update../../bin/ BAMCommandConfig.xml and add the following parameters:

(make sure to replace \*\*password\*\* with valid password): <password>\*\*password\*\*</password>

- 4. If there's a previous project BAMEventStreamAnalystics installed, clear the continuous queries or delete the project. If you are importing the BAMEventStreamAnalystics project for the first time, you can ignore the following steps.
  - a. Open Oracle BAM composer: http://<hostname>:<port>/bam/composer

- b. Choose 'Administrator'
- c. Click 'Continuous Queries Monitoring'
- d. Under the 'Project' dropdown list, select BAMEventStreamAnalystics
- e. Check the 'select all' checkbox
- f. Click 'Deactivate Query'
- g. Click 'Drop Query'

(+ ) /bam/compos	ser/faces/admin		⊽ C Soogle	▶ ♣ ☆ ☆ 自 🖋 ▾ ☰
ORACLE' BAM Composer		Home Administrator Designer		Preferences Help 🕶 weblogic 🕶 🔘
÷ / × 0	Continuous Queries Monitoring ×			3 🛛 -
Data Objects	Project BAMStreamAnalystics 💌 Status All	Activate Query     Deactiv	vate Query Drop Query 🕀 Refresh	
Enterprise Message Sources	Project Query	Status Server	Statement	
B Viewset Monitoring	BAMStreamAnalys PostiveTweet	Active AdminServer CREATE Q	QUERY PostiveTweet as SELECT T.monitor_count , T.AGGProduct , T.AGGser	nderName , T.AGGtweetHashTag1 , T.AGGtweetHashT
	BAMStreamAnalys NegativeTweet	Active AdminServer CREATE Q	QUERY NegativeTweet as SELECT T.monitor_count , T.AGGProduct , T.AGGs	enderName , T.AGGtweetHashTag1 , T.AGGtweetHash
	Statement CREATE QUERY PostveTweet as SELECT T.monitor_co FROM (ISTREAMGELECT IstagWinoper(IstagQine) S.senderName HAVING COUNT(*) >= 3) AST destra Arade beam.cgervice.mdbs.report.cacheBatch=true	ount, T.AGGHoduct, T.AGGenderName , T.AGGHeed derName), *, J.S.AGGenderName , Jackguiragopella derName), *, J.S.AGGenderNam / Jackguiragopella ation *combined:yms:queue,forade. beam.copervice.mdo e*	ithebiTag1, T.AGGhveeHeahTag2, T.sanderteane , BAMStreanAnalysted HaggBroakcut), " JA & AGGPookat, count!" JA Sommir , count, lestagbr seventeane FROM coveremo@FAMST simulate ON VATARGET. CORFITE is alertengine:queuecf/oracle.beam.copervice.mdbs.alertengine;ms:queue/	AS PROJECT_JUMME , PostoveTweef AS QUERY_JUMME appen(staspag)(waetHaahTaq)), *, 7 As JAS S WHREE (exonimistracore 20) GROUP BY anade.beam.copervice.mdbs.reportcache:queuecf 

Figure 1 – Continuous Query Monitoring in Oracle BAM Composer

#### Importing the Project

- 1. Enter bam-105-twitter-feed directory.
- 2. Execute importTwitterFeed.sh <wls password>

After the import is complete, you can open the project and view all components. Oracle BAM uses **bamcommand** to import projects into an Oracle BAM 12c runtime environment. If you only have the archived project file, you can use **bamcommand** to import the project to a running Oracle BAM 12c environment. The following is a sample usage for **bamcommand** to import this project.

\$ORACLE\_HOME/soa/bam /bin/bamcommand -username weblogic -host localhost -port 7001 - cmd import -type project -mode update -file TwitterFeed.zip

#### **Running the Project**

This project displays as Oracle BAM Event Stream Analytics, as shown in Figure 2, and includes two data objects, five business queries, five business views, and two dashboards.

<sup>4 |</sup> EVENT STREAM ANALYTICS IN ORACLE BAM 12.1.3



Figure 2 – Oracle BAM Event Stream Analytics Project

## Data Objects Simple Data Object: oowdemo

**oowdemo** is an **archived relation** simple data object. It is used for collecting twitter feedback data in real time from the **Negative Tweet** and **Positive Tweet** continuous queries. The structure of the data object is as follows. In this sample, the simulation data is populated by **bamloadgen**.

Column Name	Column Type	Data Type	Description
tweetText	DIMENSION	VARCHAR	The feedback message
tweetDate	ATTRIBUTE	DATETIME	The date for the feedback
tweetSource	ATTRIBUTE	VARCHAR	The source of the message
tweetHashTag1	ATTRIBUTE	VARCHAR	The tag of the message
tweetHashTag2	ATTRIBUTE	VARCHAR	The tag of the message
senderName	DIMENSION	VARCHAR	The owner to send the message
senderFriendsCount	DIMENSION	INT	The sender's friend count
senderFollowersCount	ATTRIBUTE	INT	The sender's follower count
senderLanguage	ATTRIBUTE	VARCHAR	The language for the sender

senderLocation	ATTRIBUTE	VARCHAR	The location for the sender
sentimentsscore	DIMENSION	INT	The score for the sentiment
Product	ATTRIBUTE	INT	The evolution's product
SentimentDislay	MEASURE	Calculated Field	If(sentimentscore>0)Then(Concat(Concat( " <img <br="" src="images/round_green16px.gif"/> alt='",sentimentscore),"' />"))Else(If(sentimentscore<1)Then(Conc at(Concat(" <img src='images/round_red16px.gif' alt='",sentimentscore),"' /&gt;"))Else(Concat(Concat("<img src='images/round_amber16px.gif' alt='",sentimentscore),"' /&gt;")))</img </img 
SentimentScore	MEASURE	Calculated Field	sum(sentimentscore)

#### Simple Data Object: TwitterMonitorDO

*TwitterMonitorDO* is a simple data object as well. The outputs of the *Negative Tweet* and *Positive Tweet* continuous queries *can be written back to this data object*. A Tree Model Query presents all *TwitterMonitorDO* data.

Column Name	Column	Data Type	Description
	Туре		
tweetHashTag1	ATTRIBUTE	VARCHAR	The tag of the message
tweetHashTag2	ATTRIBUTE	VARCHAR	The tag of the message
senderName	ATTRIBUTE	VARCHAR	The name of the sender
monitorcount	ATTRIBUTE	INT	The evolution count for every sender
ProjectName	ATTRIBUTE	VARCHAR	The project name for the CQL query
QueryName	ATTRIBUTE	VARCHAR	The CQL query name
Product	ATTRIBUTE	VARCHAR	The product name for the evolution

Consider a scenario where you are monitoring product sales efficiency for a company. Your primary goal is that you must monitor sales cost profits. To monitor sales costs and associated profits, you create a SalesCostProfit analysis dashboard that satisfies common business needs like analyzing sales data to identify trends, gauge sales performance, and study sales costs.

These graphical reports allow sales teams to access minute-to-minute information to achieve projected targets and resolve operational challenges. The SalesCostProfit analysis dashboard shows you a snapshot of sales activity. You can analyze metrics like overall sales against the incurred cost, key loss or profit-making transactions, geographical sales tracking, and so on. This process mandates that you make the following considerations.

#### **Business Queries**

#### Flat SQL Query: Inbound Tweet

The Inbound Tweet query retrieves the Product, senderName, Sentiment Graphic and tweethashTag2 information from the **oowdemo** data object.

Inhound	Twee	at						() m -
mbound	Twee	et ×						
								Save
* Data	Object	/oow	demo 💌 ঝ					
Select	fields	from t	he list.	🗔 S	elect All 🚕 🧇 Filters		(🍖 🍕	/ 🗙
\$	$\checkmark$	饂	Product		All are true			
0	1	饇	senderName					
Φ.	$\checkmark$	1	Sentiment Graphic					
\$	$\checkmark$	鰄	tweetHashTag2					
		饇	BEAM_ID		E			
		饂	column_18					
		Ē	DATAOBJECT_CREATED					
		20	DATAOBJECT_MODIFIED					
		饇.	senderFollowersCount					
		饂	senderFriendsCount					
		饇	senderLanguage					
		饂	senderLocation					
		饂	senderScreenName		•			
🛄 Тор	N							•
	modifie	er sq	L					
			Product	senderName	Sentiment Graphic	tweetHashTag2		
ADF				Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		<u>^</u>
WebLo	gic Ser	ver		Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter		(=)
BAM				yin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter		
ADF				Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		
Cohere	nce			Kasriniv	<img alt="2" src="images/round_green16px.gif"/>	Mexicocallcenter		
OAM				Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
								-

Figure 3 – Inbound Tweet Query Page

#### Flat SQL Query: Inbound Tweet Tactical

The Inbound Tweet Tactical query retrieves the Product, senderName, Sentiment Graphic and tweethashTag2 information from the **oowdemo** data object. It is ordered by the **DATAOBJECT\_CREATED** date time column.

Inbou	nd Tw	eet Tac	tical ×					2 🛛 🔹
H							(i) (ii)	Save
* Data	a Obje	ct /oow	demo 💌 🔂					
Selec	t field	ls from	the list.		🗐 Select All 🛛 🚕 📎 🛛 Fi	lters	()	• / ×
₽.	V	10	DATAOBJECT_CREAT	ED	^	All are true		
0	1	惶	Product					
\$	V	饇	senderName					
0	1	6	Sentiment Graphic					
0	V	饇	tweetHashTag2		=			
		饂	BEAM_ID					
		惶	column_18		2			
		20	DATAOBJECT_MODIFIE	D				
		惶	senderFollowersCount					
		饂	senderFriendsCount					
		饂	senderLanguage					
		122	senderLocation					
Пто	. N	۳Ľ,	senderScreenName		*			
	p N							•
	modi	fier St	ι					
	DA	ТАОВЈ	CT_CREATED	Product	senderName	Sentiment Graphic	tweetHashTag2	
May 2	6, 201	4 10:03	52 PM	ADF	Madara	<img alt<="" src="images/round_green16px.gif" th=""/> <th>CAcallcenter</th> <th>Ê</th>	CAcallcenter	Ê
May 2	6, 201	4 10:03	52 PM	WebLogic Server	Alexiauna	<img alt<="" src="images/round_green16px.gif" th=""/> <th>UScallcenter</th> <th></th>	UScallcenter	
May 2	6, 201	4 10:03	52 PM	BAM	yin2014	<img alt<="" src="images/round_green16px.gif" th=""/> <th>Indiacallcenter</th> <th></th>	Indiacallcenter	
May 2	6, 201	4 10:03	52 PM	ADF	Madara	<img alt<="" src="images/round_green16px.gif" th=""/> <th>CAcallcenter</th> <th></th>	CAcallcenter	
May 2	6, 201	4 10:03	52 PM	Coherence	Kasriniv	<img alt<="" src="images/round_green16px.gif" th=""/> <th>Mexicocallcenter</th> <th></th>	Mexicocallcenter	
May 2	6, 201	4 10:03	52 PM	OAM	Alexiauna	<img alt<="" src="images/round_green16px.gif" th=""/> <th>UScallcenter</th> <th>-</th>	UScallcenter	-

Figure 4 – Inbound Tweet Tactical Query Page

### Flat SQL Query: Outbound Tweeter

This query is used for retrieving the monitorcount, QueryName, senderName and tweetHashTag2 information from the *TwitterMonitorDO* data object with a filter QueryName field that is equal to the '**NegativeTweet' query**.

Outbour	nd Two	eeter ;	<							(	2 🛛	
1									i	ලු	Sa	ive
* Data	Object	Twitt	erMonitor	DO 💌 🚱								
Select	fields	from 1	he list.			🗐 Select All 🛛 🚕	~	Filters	Q.	4	1	×
\$	<b>V</b>	饇	monito	rcount				All are true				
\$		饇	QueryN	ame				𝐨 QueryName is equal to "NegativeTweet"				
\$		鑢	sender	Name								
0	<b>V</b>	饇	tweetH	ashTag2								
		饇	BEAM_I	D								
		2	DATAOE	JECT_CREATED								
		5	DATAOB	JECT_MODIFIED				•				
		皥	Product									
		餭	ProjectN	ame								
		饇	tweetHa	shTag1								
								< [				۴.
🔲 Тор	N										_	_
	modifie	21 m										•
	monit		.+	QueryName	conderName			tweetHachTag?				
4	monito	orcoui	n	NegativeTweet	Kasriniv	Mexicocallcenter		tweetnasin agz				^
3				NegativeTweet	Kasriniv	Mexicocallcenter						(=)
4				NegativeTweet	Kasriniv	Mexicocallcenter						
3				NegativeTweet	Kasriniv	Mexicocallcenter						
4				NegativeTweet	Kasriniv	Mexicocallcenter						
3				NegativeTweet	yin2014	Indiacallcenter						
				MaaabiusTuusab		Tadianallanakan						-

Figure 5 – Outbound Tweeter Query Page

### **Continuous Query: Negative Tweet**

The Negative Tweet query is created with the *Monitor Count Template* and is triggered when the volume of bad sentiment scores is greater than three for the same user, within the rolling window of one minute. In this query, it is selected with the Using Rolling Window option, and the range length is set to one minute. The query's output is written back to the **TwitterMonitorDO** data object with the Oracle BAM alert.

NegativeTweet $_{\times}$								?	) 🕅 🔻
Continuous Q	ueries				i	Þ	Save	Δ	
Activate	Continuous Queries : Negativ	eTweet							
Template	Monitor Count Template	<b>I</b>							
Description	If bad sentiment score great that	n 3 for the one user with ro	lling window of 1 minute, output user a	nd tweet position and 'Monitor Count'					
1. Measure				2. Output					
* Data Object /oowdemo Measure Field All Ø Product Sentiment column_1 senderFoll senderFoll senderFoll senderInt SenderLan Court Group By senderName * Count Greater than V Use Rolling * Range Len Update Inter	Graphic Score 8 owersCount adsCount guage ation or equal T Window gth 1 Minute yal 0 Day		E	* Output Ø All Ø monitor_count Ø AGGProduct Ø AGGsweetHashTag1 Ø AGGtweetHashTag2 Ø senderName Action &   • Insert event output fields into TwitterMonitorDO <add action=""></add>					
* Based	on DATAOBJECT_CREATED	•	•						

Figure 6 – Negative Tweet Query Page

By clicking the **Filter** button in the UI, you see that the query's filter is configured with a **sentimentscore less than 0**, which means that the query only retrieves and handles negative feedback data. It will form the query's 'where' clause in the CQL.

Edit Filt	er		×
		(a, 🕂	/ 🗙
A 4	Il are true Sentimentscore is less than 0		
	Finish Cancel		

Figure 7 – Edit Filter Page

By clicking the 'TwitterMonitorDO' link in the Action menu, you can review the query's related alert.

#### Action

ø

Insert event output fields into TwitterMonitorDO

#### <add action>

Figure 8 – Action Menu Accessed Through the Edit Filter Page

This alert aims to put the query's output write back to the TwitterMonitorDO data object.

)ata Object			
TwitterMonitor	DO		
Operation Type	insert <b>T</b>		
efine Mapping	S		
Туре	Data Object Column	Event Output Field	Upsert Key
DATETIME	DATAOBJECT_CREATED	▼	
DATETIME	DATAOBJECT_MODIFIED	<b>T</b>	
VARCHAR	tweetHashTag1	T.AGGtweetHashTag1 🔻	
VARCHAR	tweetHashTag2	T.AGGtweetHashTag2 ▼	
VARCHAR	senderName	T.senderName 🔹	
INT	monitorcount	T.monitor_count	
VARCHAR	ProjectName	PROJECT_NAME	
VARCHAR	QueryName	QUERY_NAME	
	Draduct	T AGGProduct	

Figure 9 – Map Fields Edit Page

#### **Continuous Query: Positive Tweet**

The Positive Tweet query is created with the Monitor Count Template and is triggered while the volume of good sentiment scores is greater than three for the same user, with the rolling window of one minute. In this query, it is selected with the Using Rolling Window option, and the range length is set to one minute. The query's output is written back to the data TwitterMonitorDO object.

Continuos Queries:	PostiveTweet ×		(?) 🛚	a -
Vertex       Network         Templat       Iminiar Count Templat         Iminiar Count Templat       Iminiar Count Templat         Product       Iminiar Count Templat         * Data Object       Iminiar Count Templat         * Output       Iminiar Count Templat         Iminiar Count Templat       Iminiar Count Templat         Iminiar Count Templ	Continuous Queries		i) 🔎 Save 🛕	
Attract Continuous Queries : Postive Yveet       Template     Image: Count Template       Description     Image: Count Template       * Data Object     Output       / Measure Field     Image: Count       * Data Object     Image: Count       / Resure Field     Image: Count       Image: Count				
Template Monitor Count Template Description	Activate Continuous Queries : PostiveTweet			
J. Measure         * Data Object         * Double of the one user with rolling window of 1 minute, output user and tweet position and "Monitor Count"         * Data Object         * Double of the one user with rolling window of 1 minute, output user and tweet position and "Monitor Count"         * Output         * Senderfollowers/Count         * enderfollowers/Count         * output         * output      <	Template Monitor Count Template 🚺 🚺			
Courter     C	Description If good sentiment score great than 3 for the one user with rolling window of 1 minu	ute, output user and tweet position and 'Monitor Count'		
1. Messure     * Data Object   * Output   Measure Field   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I   I <td></td> <td></td> <td></td> <td>•</td>				•
* Data Object //oowdemo Measure Field All Product Sentiment Graphic Sentiment Score Column_13 SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount SendeffollowersCount Sendeffo	1. Measure	2. Output		
Jowdemo   Measure Field   All   Product   Sentiment Graphic   Sentiment Score   Column_18   Sentiment Score   Column_18   Sentiment Graphic   SenterName   Action   Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action Action </td <td>* Data Object</td> <td><ul> <li>Output</li> </ul></td> <td></td> <td></td>	* Data Object	<ul> <li>Output</li> </ul>		
Measure Field   All   Ø Product   Sentiment Graphic   Sentiment Scroe   Colum_18   senderfollowerScount   senderfollowerScount   senderfollowerScount   senderfollowerScount   sendertanguage   sendertanguage   v Use Rolling Window   * Range Length   1   Minute v   Update Interval   0   DATAOBECT_CREATED v	/oowdemo			
All   ♥ Product   Sentiment Graphic   Sentiment Score   Column_18   senderfollowersCount   senderfinedsCount   senderfined	Measure Field	monitor_count		
✓ Product ✓ AcGsenderName ✓ AcGsenderHashTag1 ✓ AcGswetHashTag2 ✓ senderFollowersCount SenderfollowersCount SenderLaguage ✓ senderLaguage ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ✓ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Use Rolling Window * Range Length 1 Minute ※ Note ※	I All	✓ AGGProduct		
Sentiment Graphic       ■         Sentiment Graphic       ■         Sentiment Score       ■         Golumn_18       ■         senderFollowersCount       ■         senderfollowersCount       ■         senderfollowersCount       ■         senderfollowersCount       ■         senderLocation       ■         Group By       ■         senderthame       ■         * Count       Greater than or equal         Greater than or equal       3         ✓ Use Rolling Window       ■         * Range Length       1         Mutterval       0         Day       ■         * Based on       DATADOBJECT_CREATED	✓ Product	AGGsenderName		
Sentiment Score   Colum_13   SenderfollowersCount   senderf.anguage   senderf.anguage   senderf.anguage   senderf.ame	Sentiment Graphic	AGgtweetHashTag1		
Colum_13   SenderfollowersCount   * Count   Group By   SenderfollowersCount   * Count   SenderfollowersCount   * Count   SenderfollowersCount   * Count   SenderfollowersCount   * Count   SenderfollowersCount   * Count <td>Sentiment Score</td> <td>✓ senderName</td> <td></td> <td></td>	Sentiment Score	✓ senderName		
Sender FriendsCount   sender/FriendsCount   sender/Language   sender/Lan	column_18			
SenderLanguage   senderLocation     Group By   senderName   * Count   Greater than or equal *   3 *      * Use Rolling Window   * Range Length   1 *   Wute *   Update Interval   0 *   Day *   * Based on DATAD0BECT_CREATED *	senderFriendsCount			
Group By   senderName   * Count   Greater than or equal *   3 *   * Use Rolling Window   * Range Length   1 *   Minute *   Update Interval   0 *   Day *   * Based on   DATAOBJECT_CREATED *	senderLanguage	Action		
Group By senderName Count Greater than or equal  Group By Uugate Interval Dupdate Interval	senderLocation 👻	E		
senderName     * Count   Greater than or equal •     3 • <b>Count *</b> Output the point of the poi	Craup Ry	Insert event output fields into TwitterMonitorDO		
* Count     Greater than or equal *     3 *       * Use Rolling Window     *       * Range Length     1 *       Update Interval     0 *       * Based on     DATAOBJECT_CREATED *				
* Count Greater than or equal  3 V Use Rolling Window * Range Length 1 Minute Update Interval 0 Day * Based on DATAOBJECT_CREATED *	sendername	<add action=""></add>		
Greater than or equal V Use Rolling Window Range Length Update Interval Based on DATAOBJECT_CREATED V	* Count			
Vuse Rolling Window     Range Length     1	Greater than or equal 💌 3 🗘			
*Range Length 1 Minute Update Interval 0 Day *Based on DATAOBJECT_CREATED	✓ Use Rolling Window			
Update Interval 0 Day *Based on DATAOBJECT_CREATED	* Range Length 1 💭 Minute 💌			
* Based on DATAOBJECT_CREATED V	Update Interval 0 🖨 Day			
	* Based on DATAOBJECT CREATED			

Figure 10 – Positive Tweet Query Page

By clicking the Filter button in the UI, you see that the filter is configured with sentimentscore greater than 0, which means that the query only handles negative feedback data. It forms the query's 'where' clause in the CQL.

dit Filter				
	()	÷	I	×
All are true Sentimentscore is greater than 0				
Finish Cancel				

Figure 11 – Edit Filter page showing the 'sentimentscore is greater than 0' filter.

By clicking the 'TwitterMonitorDO' link in the Action menu, you can review the query's related alert.

#### Action



Insert event output fields into TwitterMonitorDO

#### <add action>

Figure 12 – Utilizing the TwitterMonitorDO Link iun the Actions Menu.

The alert aims to put the query's output write back to the TwitterMonitorDO data object.

	•
Column Event Output Field	Upsert Key
T_CREATED	•
T_MODIFIED	•
ag1 T.AGGtweetHashTag1	•
ag2 T.AGGtweetHashTag2	<b>T</b>
T.senderName	<b>T</b>
t T.monitor_count	<b>T</b>
PROJECT_NAME	▼
QUERY_NAME	▼
T.AGGProduct	•
	Column       Event Output Field         T_CREATED

Figure 13 – The TwitterMonitorDo Data Object

For a continuous query, the same TwitterMonitorDO data object is used to write back the Negative Tweet and Positive Tweet query outputs. By adding a filter by QueryName, you can divide the two outputs. For more information on applying the QueryName filter, see the Outbound Tweeter Flat SQL Query section.

#### Tree Model Query: Tweeter Tree Map Query

In the Measure panel, select the Sentiment Score field (It is a calculated field of sum(sentimentscore)), and in the Hierarchy, select TwoTier hierarchy which is created within the oowdemo data object.

Tweeter Tree Map Query $_{ imes}$				⑦ ⊠ ▼
â			()	🚱 Save
* Data Object /oowdemo 💌 🚱	💫 🧇 Hierarchy	Filters	()	• • / ×
Contraction (NOOP)  Contr	Hierarchy • TwoTier Levels bweetHachTag2 Product	• All are t	лие	
Top N				•
tweetHashTan2	Product	Sentiment Score	COUNT(*)	
CAcallcenter	ADF	258	365	*
CAcallcenter	BAM	259	190	=
CAcallcenter	OAM	311	190	
CAcallcenter	OID	172	223	
CAcallcenter		1,000	968	
Indiacallcenter	ADF	-159	212	-

Figure 14 – The Tweeter Tree Map Query Page

The hierarchy is defined in the Hierarchies Tab in the data object before you create the query.



Figure 15 – The Edit Hierarchy Window

#### **Business Views**

#### **Business View: Inbound Tweets Active**

This is a **List** type view from the **Table** category business view. It is bound with the **Inbound** *Tweet* query, and is configured with the *Runtime-Interaction* option.

Inbound Tweets Active $_{\rm X}$			2 🛛 🗸
		* Query Inbound Tweet	Properties Tuntime-Interaction
Inbound Tweets			
Product	senderName	SentimentDislay	tweetHashTag2
ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter
WebLogic Server	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter (E)
BAM	vin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter
ADE	Madara	<imo alt="1" src="images/round_green16px.gif"></imo>	CAcallcenter
Coherence	Kasriniv	<img alt="2" src="images/round_green16px.gif"/>	Mexicocallcenter
OAM	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
BPM	vin2014	<img alt="0" src="images/round_red16px.gif"/>	Indiacallcenter
BPMN	Kasriniv	<img alt="-2" src="images/round_red16px.gif"/>	Mexicocallcenter
OID	Madara	<img alt="-1" src="images/round_red16px.gif"/>	CAcallcenter
WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
BPM	yin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter
ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter
BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter
OAM	Alexiauna	<img 0'="" src="images/round_red16px.gif alt="/>	UScallcenter
BAM	yin2014	<img alt="0" src="images/round_red16px.gif"/>	Indiacallcenter
ADF	Madara	<img alt="-1" src="images/round_red16px.gif"/>	CAcallcenter
WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
BPM	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter
BPMN	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
Coherence	vin2014	<img alt="-1" src="images/round_red16px.gif"/>	Indiacallcenter
ADF	Kasriniv	<img alt="0" src="images/round_red16px.gif"/>	Mexicocallcenter
OAM	Madara	<img alt="2" src="images/round_green16px.gif"/>	CAcallcenter
BAM	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
ADF	yin2014	<img alt="-1" src="images/round_red16px.gif"/>	Indiacallcenter
Coherence	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter
OAM	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter
BPM	vin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter
BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter
OID	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter
WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
BPM	yin2014	<img alt="2" src="images/round_green16px.gif"/>	Indiacallcenter
ADF	Madara	<img alt="2" src="images/round_green16px.gif"/>	CAcallcenter
BPMN	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter
OAM	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter
BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter
ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter
WebLogic Server	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter
BAM	yin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter
ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter
Coherence	Kasriniv	<img alt="2" src="images/round_green16px.gif"/>	Mexicocallcenter
OAM	Alaviauna	kima are-limanaa/round aroon16py oif alt-11 /s	UCcolleontor

Figure 16 – The Inbound Tweets Active Page

Click *Runtime-Interaction*, enter into Active Data tab, select *Turn this query into a continuous query*, and set the *Interval* to 5 seconds. With this configuration, the query is refreshed automatically every 5 seconds.

Inbound Tweets Active $\times$				2 🛛 🔻
		* Query Inbound Tweet 💌 🚺	Properties  Runtime-Interaction	(i) Save
Inbound Tweets				
Product	senderName	SentimentDislay	tweetHashTaq2	
ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter	
WebLogic Server	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter	(E)
BAM	vin7014	<imp alt="1" src="images/round_green16nv_gif"></imp>	Indiacallcenter	
ADF	Runtime-Interaction		× center	
Coherence			ocallcenter	
OAM	Drilling		center	
BPM	Full full clis query into a continuous query		allcenter	
BPMN	Actions		ocallcenter	
OID	Active Data		center	
Wehl onic Server	Active Data Collapsing		center	
BPM	Interval 5 🖨 seconds	•	allcenter	
ADE			center	
BPMN	Use a time window		ncallcenter	
OAM			center	
BAM	Sliding Range Based on DATAOBJECT CREA	TE	allcenter	
ADE			center	
WebLogic Server	Range Length 1 seconds	V	center	
BPM	Range Edityti 1		ncallcenter	
BPMN			center	
Coherence	Update Interval 1 seconds	•	allcenter	
ADE			ocallcenter	
OAM			center	
BAM			center	
ADE			allcenter	
Coherence			ocallcenter	
OAM			center	
RDM			allcaster	
BPMN			ancenter	
OID			center	
WebLogic Server			center	
BDM			allcenter	
ADE			center	
RDMN			scallconter	
DAM			center	
PDMN			scallconter	
ADE			canter	
Wahl agis Capyor	0		center	
RAM	3	Apply	Close	
ADE	Madara	<ing 1'_="" src="images/round_green16ny_giP_alt="></ing>	Chcallcenter	
Coherence	Kacriniy	<pre><img 1'="" src="images/round_green16px.git alt="/></pre>	Mexiconalizanter	
OAM	Alaviaura	<pre>sing src=images/round_greentop.igit_alt='1'/s </pre>	II Callconter	

Figure 17 – Turning a Query into a Continuous Query

#### **Business View: Inbound Tweets Tactical**

This view is bound with the Inbound Tweet Tactical query.

Inbound Tweets Tactical $\times$					?	•
			* Query Inbound Tweet Tactic 💌 🚱	Properties Cuntime-Interaction	()	save
Inbound Tweets Tactical						
DATAOBJECT CREATED	Product	senderName	SentimentDislay	tweetHashTag2		
14/05/26 22:03:52:178	ADF	Madara	<img 1'="" src="images/round green16px.gif alt="/>	CAcallcenter		~
14/05/26 22:03:52:178	WebLogic Server	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter		(E)
14/05/26 22:03:52:178	BAM	yin2014	<img alt="1" src="images/round green16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	Coherence	Kasriniv	<img alt="2" src="images/round_green16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OAM	Alexiauna	<img alt="1" src="images/round green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPM	vin2014	<img alt="0" src="images/round_red16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	BPMN	Kasriniv	<img alt="-2" src="images/round_red16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OID	Madara	<img alt="-1" src="images/round_red16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPM	yin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OAM	Alexiauna	<img alt="0" src="images/round_red16px.gif"/>	UScalicenter		
14/05/26 22:03:52:178	BAM	vin2014	<img alt="0" src="images/round_red16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="-1" src="images/round_red16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPM	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	BPMN	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	Coherence	yin2014	<img alt="-1" src="images/round_red16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Kasriniv	<img alt="0" src="images/round_red16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OAM	Madara	<img alt="2" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	BAM	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	ADF	vin2014	<img alt="-1" src="images/round_red16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	Coherence	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OAM	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPM	vin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OID	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	WebLogic Server	Alexiauna	<img alt="1" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPM	vin2014	<img alt="2" src="images/round_green16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="2" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	BPMN	Kasriniv	<img alt="1" src="images/round_green16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	OAM	Alexiauna	<img alt="1" src="images/round green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BPMN	Kasriniv	<img alt="-1" src="images/round_red16px.gif"/>	Mexicocallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="1" src="images/round_green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	WebLogic Server	Alexiauna	<img alt="2" src="images/round_green16px.gif"/>	UScallcenter		
14/05/26 22:03:52:178	BAM	yin2014	<img alt="1" src="images/round_green16px.gif"/>	Indiacallcenter		
14/05/26 22:03:52:178	ADF	Madara	<img alt="1" src="images/round green16px.gif"/>	CAcallcenter		
14/05/26 22:03:52:178	Coherence	Kasriniv	<img alt="2" src="images/round_green16px.gif"/>	Mexicocallcenter		
14/05/26 22:02:52:170	0.4M	Alaviauna	rime are-limagoolround, geoontEnv aif alt-111/2	UScallcontor		+

Figure 18 – The Inbound Tweets Tactical View Page

#### Business View: Inbound Tweet TreeMap

This view is bound with the *Tweeter Tree Map Query*. According to the *TwoTier* hierarchy (tweetHashtag2, Product), you can divide the panel into different areas. The first group is divided by tweetHashtag2. They are: CAcallcenter, Indiacallcenter, UScallcenter and Mexicocallcenter. For every tweetHashTag2, the new group is Product. They are: ADF, ODI, OAM, BAM, BPM, Weblogic Server, BPMN, and Coherence.

In	hound Tweet TreeMan						() <b></b>
-							
		* Query Tweeter Tree	: Map Qu 💌 🙀 Value	COUNT(*)	Default Color NOOP(SentimentScore	Properties 📰 Runtime-Inte	eraction i Save
In	bound Tweet TreeMap						
	CAcallcenter		Indiacallcenter			Mexicocallcenter	
ADF		ROM RAM		ADF BPMN		4N	
					Coherence		
			UScallcenter				
	OID	ΟΑΜ	ОАМ	WebLogic Server	ВАМ	ВРМ	ADF
	ВАМ				BPMN	Coher	ence

Figure 19 – The Inbound Tweet Tree Map

In order to display different areas with different colors, you must define Thresholds in the Properties tab.

Properties					2
Properties General Titles Thresholds Node Depth Layout Labels	Color [ Hierarchy level ]	NOOP(SentimentScore Product Clinear Threshold Thresholds Threshold 0-1 2-1000	Custom Threshol	lds Overflow Color Underflow Color No Value	2
		All properties in t	this this tab re	quire a Save to be displayed.	
?					Apply Close

Figure 20 – Defining Thresholds in the Properties Tab

#### **Business View: Outbound Tweets Active**

This view is bound with the Outbound Tweeter query and is configured with the Runtime-Interaction option. Click Runtime-Interaction, enter into Active Data tab, select Turn this query into a continuous query, and set the Interval as 5 seconds. With this configuration, the query will be refreshed automatically, every 5 seconds.

Outbound Tweets A	ctive $\times$					? 🛛 🕶
			* Query O	utbound Tweeter 🔳 🚺	Properties The Runtime-Interaction (1)	Save
Outbound Tweets A	ctive					
monitorcount	t	OuervName	senderName	1	tweetHashTag2	
15		NegativeTweet	Kasriniv		Mexicocallcenter, Mexicocallcenter, Mexicocallcenter,	Mexi 🔺
4		NegativeTweet	Kasriniv		Mexicocallcenter	
3		NegativeTweet	vin2014		Indiacallcenter	
3		N 10 7 1			Mexicocallcenter	
4	Runtime-Interacti	ion			× Mexicocalicenter	
3					Mexicocalicenter	
4	Drilling	Turn this query into a continuous query			Mexicocalicenter	
3	Drining	rum chis query into a continuous query			Mexicocalicenter	
4	Actions				Mexicocalicenter	
2	Active Data				Maxicocalicenter	
4		Active Data Collapsing			Mexicocalicenter	
7		Interval 5 🗘 seconds 💌			Tediacelleseter	
2					Indiacalicenter	
2		Use a time window			Inulacation ter	
2					Indiacalicenter	
3		Sliding Range Based on DATAOBJECT_CREATE			Mexicocalicenter	
4					Mexicocallcenter	
3		Pange Length 1 carondo	×		Mexicocalicenter	
4		Kange Lengun 1 Seconds	*		Mexicocalicenter	
4					Mexicocallcenter	
3		Update Interval 1 seconds	*		Mexicocallcenter	
3					Mexicocallcenter	
4					Mexicocallcenter	
3					Indiacallcenter	
3					Indiacallcenter	
3					Indiacallcenter	
3					Mexicocallcenter	
7					Mexicocallcenter, Mexicocallcenter, Mexicocallcenter,	, Mexi
7					Indiacallcenter, Indiacallcenter, Indiacallcenter, India	acallc
11					Indiacallcenter, Indiacallcenter, Indiacallcenter, India	acallc
14					Mexicocallcenter, Mexicocallcenter, Mexicocallcenter,	, Mexi
16					Mexicocallcenter, Mexicocallcenter, Mexicocallcenter,	, Mexi
17					Mexicocallcenter, Mexicocallcenter, Mexicocallcenter,	, Mexi
3					Indiacallcenter	
3					Indiacallcenter	
3					Indiacallcenter	
3					Mexicocallcenter	
4		1			Mexicocallcenter	
3	(?)			Apply Close	Mexicocallcenter	
4	•				Mexicocallcenter	
3		NegativeTweet	Kasriniv		Mexicocallcenter	
4		MaastivaTweet	Vacriniu		Mavicocallcontor	+

Figure 21 – Turning the Outbound Tweets Active Query into a Continuous Query

#### **Business View: Outbound Tweets Tactical**

This view is bound with the Outbound Tweeter query.

Outbound Tweets Tactical $_{\rm X}$			3 🕅 -
		* Query Outbound Tweeter	💌 🍓 Properties 🛄 Runtime-Interaction 🧃 Save
Outbound Tweets Tactical			
monitorcount	QueryName	senderName	tweetHashTag2
15	NegativeTweet	Kasriniv	Mexicocallcenter, Mexicocallcenter, Mexicocallcenter, Mexi 🔺
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocalicenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
4	NegativeTweet	Kasriniv	Mexicocallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	yin2014	Indiacallcenter
3	NegativeTweet	Kasriniv	Mexicocallcenter
7	NegativeTweet	Kasriniv	Mexicocallcenter, Mexicocallcenter, Mexicocallcenter, Mexi
7	NegativeTweet	yin2014	Indiacallcenter, Indiacallcenter, Indiacallcenter, Indiacallc
11	NegativeTweet	yin2014	Indiacalicenter, Indiacalicenter, Indiacalicenter, Indiacalic
14	NegativeTweet	Kasriniv	Mexicocalicenter, Mexicocalicenter, Mexicocalicenter, Mexi
16	NegativeTweet	Kasriniv	Mexicocalicenter, Mexicocalicenter, Mexicocalicenter, Mexi
17	NegativeTweet	Kasriniv	Mexicocallcenter, Mexicocallcenter, Mexicocallcenter, Mexi
3	NegativeTweet	yin2014	Indiacalicenter
3	NegativeTweet	yin2014	Indiacallcenter
3	Negative I weet	yin2014	Indiacalicenter
3	Negative I weet	Kasriniv	mexicocalicenter
4	NegativeTweet	Kasriniv	Mexicocalicenter
3	NegativeTweet	Kasriniv	Mexicocalicenter
7	Negative I weet	Kasriniv	mexicocalicenter
3	Negative i weet	Kasriniv	mexicocalicenter
~	measter i weat	P Series	HAVEACHEADTAL

Figure 22 – The Outbound Tweets Tactical View Page

#### Alerts

# NegativeTweet\_Output\_Alert

This alert is created during the <add action> in NegativeTweet query.

legativeTweet_Output_Aler	tx						? 🛛 🔹
Alert Display Na	me : NegativeTweet_Output_	Alert / Alert : NegativeTweet_C	Output_Alert			()	? Save
Select NegativeTweet	t <del>*</del> fields into TwitterMonitorDO						
<add action=""></add>							
Map Fields Data Object TwitterMonitorDO Operation Type insert	•	×			×		
Define Mappings							
Туре	Data Object Column	Event Output Field		Jpsert Key			
DATETIME	DATAOBJECT_CREATED		•				
DATETIME	DATAOBJECT_MODIFIED		•				
VARCHAR	tweetHashTag1	T.AGGtweetHashTag1	•				
VARCHAR	tweetHashTag2	T.AGGtweetHashTag2	•				
VARCHAR	senderName	T.senderName					
1111	monitorcount	1.monitor_count					
VARCHAR	ProjectName	PROJECT_NAME	•		_		
VARCHAR	QueryName	QUERY_NAME	•				
VARCHAR	Product	T.AGGProduct	•		_		
					ок		

Figure 23 - NegativeTweet\_Output\_Alert

## Postive Tweet\_Output\_Alert

This alert is created during the <add action> in Positive Tweet query UI. This alert uses the alert engine to write back the CQL result to one data object. You must configure the mapping between CQL output fields with the data object column.

PostiveTweet_Output_Alert >	c						2 🛛 🔹
■Alert Display Na	me : PostiveTweet_Output_	Alert / Alert : PostiveTweet_Ou	tput_Alert				
Select PostiveTweet -	ields into TwitterMonitorDO						
<add action=""></add>							
Map Fields					×		
Data Object							
TwitterMonitorDO							
Operation Type insert							
Define Mappings							
Type	Data Object Column	Event Output Field	Upse	ert Key			
DATETIME	DATAOBJECT_CREATED						
DATETIME	DATAOBJECT_MODIFIED	T ACCHURANIA-Tant			_		
VARCHAR	tweetHashTag1	T.ACChueatHashTag1					
VARCHAR	conderName	T. conderName					
INT	monitorcount	T monitor count					
VARCHAR	ProjectName	PROJECT NAME			_		
VARCHAR	QueryName	OUERY NAME					
VARCHAR	Product	T AGGProduct					
	110000	- market realified		hanned			
					-		
					OK		

Figure 24 - Postive Tweet\_Output\_Alert

#### Dashboards

#### **Oracle BAM Event Stream Analytics Active**

This dashboard has three views: Inbound Tweet TreeMap on the top, Inbound Tweets Active at the bottom left, and Outbound Tweets Active at the bottom right. The tree map does not support active data services but the list views are active views.



Figure 25 - The Oracle BAM Event Stream Analytics Active Dashboard

### **Oracle BAM Event Stream Analytics Tactical**

This dashboard has three views: Inbound Tweet TreeMap at the top, Inbound Tweets Tactical at the bottom left, and Outbound Tweets Tactical at the bottom right. These two tactical views cannot be refreshed automatically. You must refresh the browser to see the latest chart.



Figure 26 - The Oracle BAM Event Stream Analytics Tactical Dashboard

# Creating a Project

This section shows you how to create a similar Oracle BAM Event Stream Analytics project.

#### **Creating a Data Object**

Open Oracle BAM composer, click the 'Administrator' link, choose the 'Data Object' node and click '+', and select 'Simple Data Object'. In this case project, we have selected the Archived and Continuous Query Type with RELATION, and have filled the data object name with oowdemo or TwitterMonitorDO. However, because TwitterMonitorDO is a write back data object, you must add the correct columns to adapt with the continuous query's output.

* Name	oowdemo
* Display Name	oowdemo
* Type	Simple Data Object 🔹
Archived	~
* Continuous Query Type	STREAM I RELATION
Category	
Number of String Columns	25
Number of Long String Columns	25
Number of Integer Columns	25
Number of Float Columns	25
Number of Decimal Columns	25
Number of Date/Time Columns	10
Description	
2	Create Cancel

Figure 27 – Creating a Data Object

In order to define the write back data object, you must know the output fields of your continuous queries. You can create the continuous query before defining this data object. After saving the query, the output is clear. You can then create this data object according to the continuous query's output. For example, consider that you want to create a data object to write back the output of two continuous queries: Negative Tweet and Positive Tweet. First save the queries. This ensures that the following fields are listed in the output panel: monitor\_count, AGGProduct, AGGsenderName, AGGtweetHashTag1, AGGtweetHashTag2 and senderName. You can now add appropriate columns in the data object and use one data object to write back many of queries' outputs. In order to distinguish all the data in the data object, you can create two columns in the data object: ProjectName and QueryName. For all continuous queries created by templates, these two fields will be output impliedly.

To create a new project, open Oracle BAM composer, click the 'Designer' link, and choose 'create' under the dropdown list to create a project. Fill out the Name and Display Name fields and click the 'Create' button, as shown in Figure 28.

Create		×
Create BAM P	roject	
*Name	BAMEventStreamAnalytics	
* Display Name	BAM Event Stream Analytics	
Description		
-		
?	Create BPM Example Create Cancel	

Figure 28 - Creating a new Oracle BAM Project

If you want to modify the project display name, you can choose 'Rename' under the dropdown list, edit the Display Name and click the 'Save' button, as shown in Figure 29.

Rename		×
Name	BAMStreamAnalystics	
* Display Name	BAM Event Stream Analytics	
	Save Cancel	

Figure 29 – Renaming a project's display name.

After creating the data objects, you must add them to the project. Click 'Design' to open the project, then click the 'Data Objects' node. Click the '+' button to open the 'Data Objects' window. Select the data objects you want to add, and click the 'Add' button to add them to the project.

		View All	View All		
Display Name	Path	Туре	Category	Description	
MonitorCntOutputDDO	/oracle/writeback	Derived Data Object			
MovAggrOutputDDO	/oracle/writeback	Derived Data Object			
oowdemo		Simple Data Object			
PATTERNMATCH	/oracle/callcenter	Logical Data Object			
PATTERNMATCH_DIMENSION	/oracle/callcenter	Simple Data Object			
PATTERNMATCH_FACT	/oracle/callcenter	Simple Data Object			
PATTERNMATCH_FACT_STREAM	/oracle/callcenter	Simple Data Object			
PreseedingFileHistory	/oracle/bam/internal	Simple Data Object			
Process	/oracle/processanalytics	Logical Data Object	PROCESS		
Process (physical)	/oracle/processanalytics/internal/physical	Simple Data Object	PROCESS		
Process Definition (physical)	/oracle/processanalytics/internal/physical	Simple Data Object	PROCESS		
Role Definition (physical)	/oracle/processanalytics/internal/physical	Simple Data Object	PROCESS		
StreamDO1		Simple Data Object			
TopNOutputDDO	/oracle/writeback	Derived Data Object			
TrendingDetectionOuputDDO	/oracle/writeback	Derived Data Object			
TwitterMonitorDO		Simple Data Object			

Figure 30 – Adding Data Objects to a Project

While creating a project, the project name and display name are validated by the following rules. To ensure that the project is created correctly, here are some suggested best practices:

- 1. The project name and display must be unique in the Oracle BAM system.
- 2. The project name must begin with capital or lower case letters only, i.e., a-z, or A-Z.
- 3. Only letters, numbers (0-9) and the underscore are allowed in the project name.

4. The project name length must be less than 128 characters.

#### **Business Queries**

This section provides information on the business queries supported in Oracle BAM. In addition to outlining their setups, you can also look at some of the suggested best practices while creating each of these business queries.

## Flat SQL Query

Open a project, then choose 'Business Queries'. Click '+', and select 'FLAT SQL Query' to create a Flat SQL Query.



Figure 31 – Creating a Flat DQL Business Query

Open the query edit page, and fill the data object field, select the output fields from the list. You can add a filter to the query by clicking the '+' button in Filter panel, which opens the filter editor.

#### Suggested best practice:

The Flat SQL Query is a plain query, and it cannot retrieve data which has the 'grouped by' clause. To retrieve data with groups or different dimensions, select **Type** with '**Group SQL Query'** in the Create Business Queries UI.

#### **Continuous Query**

To create a Continuous Query, open a project, choose 'Business Queries', click '+', and select 'Continuous Query'.

Bu	usiness Queries									×
	* Name	Negativ	eTweet							
	* Display Name	Negativ	eTweet							
	*Туре	۲	2	0		0		0	īoj	
			Continuous Query		Group SQL Query		Tree Model Query		Flat SQL Query	
(	?							Create	Cance	2

Figure 32 – Creating a Continuous Query

Open the continuous query edit page and select one of templates from the dropdown list. Fill all the required fields and save the query.

Click the Filter button in the UI to add the query's filter. If the filter has been configured with a sentimentscore less than 0, the query only retrieves and handles negative feedback data. It forms the query's 'where' clause in the CQL.

NegativeTweet ×								2 🛛 🔹
Continuous	Queries					i	Save	Δ
Activate	Continuous Queries : Negative	weet						
Template	Monitor Count Template	<b>•</b> (i)	Edit Filter		×			
Description	If bad sentiment score great than	3 for the one use	All are true		()- + / 🗶			
1. Measure			Y sentimentscore is less than 0					
* Data Ohiert		P Filter						
/oowdemo Measure Field		• 🔞						
All								
Sentimer	it Graphic							
Column	it Score 18	E						
senderFo	llowersCount			Finish Cancel				
senderFr	iendsCount							
senderLa	nguage cation	Ŧ						
				🔅 👻 Insert event out	put fields into TwitterMonitor	rDO		
Group By				<add action=""></add>				
senderName								
* Count								
Greater than	or equal 💌 3 🗘							
🗸 Use Rollin	g Window							
* Range Lei	ngth 1 Minute 💌	]						
Update Inte	erval 0 🖉 Day 💌	]						

Figure 33 – Adding a Filter to the Query

To write back the query's output to a data object, click the <add action> link to create an action. Then select the 'Insert values into data object' option in the 'Select an Action' UI, as shown in Figure 34.



Figure 34 - Using the Action Editor to Add Values to a Data Object

Click the <select data object> link, and open the 'Map Fields' UI. Select TwitterMonitorDO as the data object, then select the correct Event Output Field Map with the data object's columns, as shown in Figure 35.

Map Fields					>
Data Object					
TwitterMonitorDO					
Operation Type insert					
Define Mappings					
Туре	Data Object Column	Event Output Field		Upsert Key	
DATETIME	DATAOBJECT_CREATED		•		
DATETIME	DATAOBJECT_MODIFIED		•		
VARCHAR	tweetHashTag1	T.AGGtweetHashTag1	•		
VARCHAR	tweetHashTag2	T.AGGtweetHashTag2	-		
VARCHAR	senderName	T.AGGsenderName	•		
INT	monitorcount	T.monitor_count	•		
VARCHAR	ProjectName	PROJECT_NAME	-		
VARCHAR	QueryName	QUERY_NAME	•		
VARCHAR	Product	T.AGGProduct	•		
					ОК

Figure 35 – Selecting the Appropriate Field Mappings for a Data Object

Click the <u>A</u> button to save and start the alert action to 'listening' for the query's outputs. If the query has been triggered with the output, you can see output data from the data object.

#### Here is a suggested best practice:

The continuous query is different from the group and flat query in Oracle BAM 12c. In order to create a correct continuous query, you must know and understand your requirement. You must also understand which templates can be used for your cases. For the 7 template usages, refer to the Oracle BAM user guide documents. Select the correct template and fill all the required fields in the UI to get your query.

<sup>28 |</sup> EVENT STREAM ANALYTICS IN ORACLE BAM 12.1.3

To validate whether the query meets your requirements, you can preview the CQL by clicking the icon at the top right corner of the UI. You can also populate some test data using Oracle BAM **loadgen** or insert data from the data object's data tab in the UI, and write back the query's output to one data object. Finally, you can check the output to see whether the query is correct.

Figure 36 shows the relation between a CQL and the UI. It helps you understand the template and your requirements.



Figure 36 - A CQL Comparison with its Output in the UI

#### **Tree Model Query**

To create a tree model query, open a project, choose 'Business Queries', click '+', and select 'Tree Model Query'.

Bu	isiness Queries								\$	ĸ
	* Name	Tweeter	TreeMap							
	* Display Name	Tweete	rTree Map Query							
	*Туре	0	2	0		۲		0		
			Continuous Query		Group SQL Query		Tree Model Query		at SQL Query	
(	?							Create	Cancel	

Figure 37 - Creating a Tree Model Query

Suggested best practice:

A Tree Model Query needs 'hierarchy' as a demission to divide data to different tree nodes or levels. If you want to create this query, you must define the Hierarchy property in the data object. You must choose columns that have some fixed value sets as Hierarchy selected columns.

#### **Business Views**

This section provides information on the different business views supported in Oracle BAM.

**Table Type View** 

To create a Table Type View, open a project, choose 'Business Views', click '+', select the List type and Table Category, and select List from View Types.



Figure 38 – Creating a Table Type View

#### Tree Map Type View

To create a Tree Map Type View, open a project, choose 'Business Views' and click '+', select 'Treemap' in the Categories list and select 'Treemap' from View Types.

Business Views	
* Name	InboundTweetsTreeMap
* Display Name	Inbound Tweets TreeMap
Categories	View Types :
Area	Plonde North Care Virginia
	Coopie Hardent
Combo	
Table	
KPI Watchlist	
Gauge	
Scatter/Polar	
Bubble	
Treemap	
?	

Figure 39 – Creating a Tree Map Type View

## Dashboards

Open a project, then choose 'Dashboards' and click '+', select 'Type2' from 'Select the style template' to create a dashboard. Open the dashboard edit page, and drag the specific business views from the appropriate list to the correct area.

Dashboards		1
* Dashboard Name	BAMEventStreamAnalyticsActive	
* Dashboard Display Name	BAM Event Stream Analytics Active	
Dashboard Description		
Select Dashboard Type	<ul> <li>Dashboard</li> <li>Tabbed Dashboard</li> </ul>	
	Type 1 Type 2 Type 3	
Select the style template	Type 4         Type 5         Type 6	
	Type 7         Type 8         Type 9	
3	Create Cancel	

Figure 40 – Creating a Dashboard

Accessing the Dashboard

1. Open Oracle BAM dashboard:

a. 'BAM Event Stream Analytics Active' - Twitter dashboard for active data:

http://<hostname>:<port>/bam/composer/faces/proxypage?project=BAMStreamAnalystics&das

hboard=BAMEventStreamAnalystics

<b>(-) (3</b>	/bam/composer/fac	es/proxypage?dashboard=B	AMEventStreamAnalystic	s&project=BAMS	itreamAnalystics	습 ▼ C 🛛	≠ Google	۹ 🖡
AM Event Stream Ar	nalytics							Q   🖨   🛛
nbound Tweet TreeMap								÷ (
CAcallcenter		UScallcenter		India	callcenter		Mexicocallcenter	
ADF	ВАМ	WebLogic Server	BPMN		Coherence	ВАМ	BPMN	ADF
OID	OAM	OAM	BAM		врм	ADF	BPM	Coherence
nbound Tweets			•	🔋 🔲 👘	Tweets Active			
Product	Tweet From	Sentiment	Call Center	No.	Of Negatives	QueryName	Tweet From	Call Center
BPM	Kasriniv	٥	Mexicocallcenter	•	3	NegativeTweet	Kasriniv	Mexicocallcenter
BPM	Kasriniv	0	Mexicocallcenter	E	3	NegativeTweet	Kasriniv	Mexicocalicenter
ADE	Madara		CAcallcenter		4	NegativeTweet	Kasriniv	Mexicocalcenter
Coherence	vin2014	ă	Indiacallcenter		4	NegativeTweet	Kasriniv	Mexicocalicenter
RDMN	Alexiaupa		LiScallcenter		3	NegativeTweet	vin2014	Indiacalicenter
DPIMIN	Alexidurid	ă	Oscalicenter		3	NegativeTweet	vin2014	Indiacalicenter
OAM	Mauara		CACalcenter		3	NegativeTweet	Kasriniv	Mexicocallcenter
BAM	Alexiauna		UScallcenter					
BAM	Madara		CAcallcenter					
Coherence	yin2014		Indiacallcenter					
OAM	Madara		CAcallcenter					
ADF	Madara	•	CAcallcenter					
ADF	Madara	0	CAcallcenter					
	1010100 (1010)	0	Indiacallegator					
ADF	yin2014	<b>u</b>	Indidudicenter					

Figure 41 – Oracle BAM Event Stream Analytics Dashboard

b. 'BAM Event Stream Analytics Tactical' - Twitter dashboard for tactical data:

http://<hostname>:<port>/bam/composer/faces/proxypage?project=BAMStreamAnalystics&das

hboard=BAMEventStreamAnalyticsTactical

€ 0	/bam/	composer/faces/proxyp	age?dashboard=BAMI	ventStreamAnalyticsT	actical&	project=BAMStreamAnalysti	ics 🏠 🛡 🕑 🔡	▼ Google	₽ ٩	Â
3AM Event Stream	Analytics Tact	ical							🔁   🖨	8
nbound Tweet TreeMa	p								•	۲
CAcallcenter		UScallcenter				Indiacallcenter		Mexicocallcenter		
ADF		OAM WebLogic Server		ВАМ		ВРМ	ADF	BPMN	ВРМ	
		BAM	OAM	BPMN		Вам	Coherence	Coherence	ADF	
nhound Tweate Tactics	J					thought Tractical				•
					@ 00	u of u u				4
Tweet Time	Product	I weet From	Sentiment	Call Center	12	No. Of Negatives	QueryName	Iweet From	Call Center	
3/12/02 11:53:28:296	BAM	Alexiauna		UScallcenter	â	4	NegativeTweet	Kastiniy	Mexicocalicenter	
3/12/02 11:53:27:278	OID	Madara		CAcallcenter		3	NegativeTweet	vin2014	Indiacalicenter	
3/12/02 11:53:26:263	ADF	yin2014		Indiacallcenter		3	NegativeTweet	Kasriniv	Mexicocallcenter	
3/12/02 11:53:25:247	BAM	Alexiauna	•	UScallcenter		3	NegativeTweet	yin 2014	Indiacallcenter	
3/12/02 11:53:24:233	OID	Madara		CAcallcenter		3	NegativeTweet	Kasriniv	Mexicocallcenter	
3/12/02 11:53:23:218	ADF	yin2014	0	Indiacalicenter		3	NegativeTweet	Kasriniv	Mexicocallcenter	
3/12/02 11:53:22:199	BAM	Alexiauna	0	UScallcenter		4	NegativeTweet	Kasriniv	Mexicocalicenter	
3/12/02 11:53:21:189	ADF	vin2014		Indiacalicenter		3	NegativeTweet	yin2014 Kacriniy	Indiacalicenter	
3/12/02 11:53:20:175	BAM	Madara	ő	CAcallcenter		3	NegativeTweet	Kasriniv	Mexicocalicenter	
3/12/02 11:53:19:158	BAM	vin2014	ă	Indiacalicenter		4	NegativeTweet	Kasriniv	Mexicocallcenter	
3/12/02 11:53:18:144	Coherence	Kacriniv	ă.	Maxicocalicenter						
2/12/02 11:55:10:144	ADE	NOSHIV		Indiacalisanter						
3/12/02 11:33:17:132	ADE	ym12014								
3/12/02 11:53:16:112	DAM	Alexiauna		Uscalcenter	-					

Figure 42 - Oracle BAM Event Stream Analytics Tactical

- 2. Execute startTwitterFeed.sh <wls password>
- 3. To stop the data population, execute stopTwitterFeed.sh

Note: Please refer to the following default values:

- · WLS password: weblogic1
- Hostname: localhost
- Port: 7004

For populating data to the BAM data object, use the bamloadgen command. The following is a sample usage for bamloadgen to populate data to the data object oowdemo with 1 row per second frequency. In this command, we set the duration to 0, so it populates data forever. You can also specify a value for the duration parameter and allow data population for a limited time period.

\$ORACLE\_HOME/soa/bam /bin/bamloadgen -XMLFile data/oowdemo.xml -duration 0 -frequency 1 -username weblogic -password welcome1 -host localhost -port 7001

In order to populate the data using bamloadgen, you need to prepare an xml data file. The fields in the xml file need to be the same as the data object column names. This is a sample xml for populating data to the oowdemo data object. You can use the \$currentTime token to populate a system at the current date. You can also set a fixed date with the date time field in the xml.



Figure 43 – Populating an XML Data File

# Troubleshooting

This section provides information on some scenarios which might require you to run troubleshooting. In this project, Tree Model Query and Continuous Query are used to demonstrate how the event sinks from one flow (the write back DO from the continuous queries) to be the event source of another flow (the data source for dashboard data population).

#### **Project Import Issues**

1. You cannot import the project using bamcommand

The bamcommand script is put into the directory \$ORACLE\_HOME/soa/bam/bin. Firstly, ensure that you have set the correct \$JAVA\_HOME. Then, ensure that the Oracle BAM server is running and that the Oracle BAM Composer can be accessed.

2. You cannot see the project in the Oracle BAM Composer after completing the import.

Ensure that you have set the correct host address and have used the correct and zip file. Then, ensure that you have used the correct '-mode' parameter. If you specify 'append', the project cannot be imported if it has existed on the Oracle BAM server. If you specify 'update', the project will override the previous instance on the Oracle BAM server.

#### **Business Query Issues**

1. Continuous queries are not active.

The continuous query is not active automatically. If you want to activate (register and activate) the query, you need to select the 'Active' check box, then click the 'Save' button. If there isn't an exception while saving the query, it should be active successfully. You can close and reopen the query to check whether the check box is still active. Another way to check the continuous query status is by going to 'Continuous Queries Monitoring' in the 'Administrator' mode, and selecting your project from the Project dropdown list. All the continuous queries in the project are listed on the page. You can activate or deactivate the query by using the toggle button.

4 / X 🕅	Continuous	Queries Mon	itoring ×			(3) 🕅				
Data Objects	Project BAMStreamAnalystics • Status All • Activate Query Deactivate Query Drop Query Deactivate Query									
Enterprise Message Sources  Continuous Queries Monitoring  Viewset Monitoring		Project	Query	Status	Server	Statement				
	BAM:	StreamAnalys	PostiveTweet	Mactive	AdminServer	CREATE QUERY PostiveTweet as SELECT T.monitor_count , T.AGGProduct , T.AGGsenderName , T.AGGtweetHashTag1 , T.AGGtweetHashTag2 , T.send				
	BAM:	StreamAnalys	NegativeTweet	Mactive	AdminServer	CREATE QUERY NegativeTweet as SELECT T.monitor_count , T.AGGProduct , T.AGGsenderName , T.AGGtweetHashTag1 , T.AGGtweetHashTag2 , T.se				
	Statement	CREATE QU ISTREAM(S) AS T destri	ERY PostweTweet as SELECT T monitor_ ELECT Liskagily rapped (Liskagily rapped) abon * Combined syns: queue/practe. beam.o	count, T.AGGProduct e., 1, 1, 3, AGZende e., 1, 1, 3, AGZende e., 1, 1, 1, 3, AGZende e., 1, 1, 1, 2, 3, 4, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	, T.AGGsenderName , rlsane , listaggiVirappe ngme:queue:fjorade.t	T. AGGtweetHashTag1, T. AGGtweetHashTag2, T. sendenName, "BAMStreamAnalystical AS PROJECT_NAME, "PostiveTweet" AS QUERY_JAME FROM ( (pathog@nob.cct), "J. AS AGGHnob.cct, count(") AS monitor, count, letalogi/inspectificitag0/weetHashTag1, (") AS AGStweetHashTag1, (") AS AGGINeetHashTag2, T. sendenName, "BAMStreamAnalystical AS PROJECT_NAME, "PostiveTweet" AS QUERY_JAME FROM ( (pathog@nob.cct), "J. AS AGGHnob.ct, count(") AS monitor, count, letalogi/inspectificitag0/weetHashTag1, ", ") AS AGStweetHashTag1, (") >= 0) ean opervice.mdbs.alertempregmicipus.ec/oracle.beam.opervice.mdbs.reports.cher.gueuest/foracle.beam.opervice.mdbs.reports.acherbashTab1="); // / // // // // // // // // // // // /				

Figure 44 – Making a Continuous Query Active

2. Tree model query cannot be created.

The Tree model query is a hierarchy group query. Ensure that you have defined some hierarchical relation in the data object and that the relation is valid. If the relation cannot group the data within each part, the query is invalid.

3. Cannot retrieve correct data from a Flat SQL Query

For the all the queries in Oracle BAM 12c, the data can be configured with row level security filter. This means that users cannot see all the data in a specific data object even if they can access it. Ensure that your query is correct and retrieves data from the data object. Then, review your query's SQL from the UI. You can check whether there is an additional 'where' clause appended to the query's SQL, in which case, you must obtain the runtime security filter permission for your current user role from the administrator. Alternatively, you can modify the security filter for the role, or try to login with another user role's credentials.

#### **Business View Issues**

1. Cannot see the data in the business view being refreshed automatically.

Ensure that the business view is configured with 'Active Data'. Only the query has to be configured with the active data service; the data into the view can be refreshed automatically. Then, go to 'Continuous Queries Monitoring' in 'Administrator'. Select your project from the Project dropdown list. All the business views with active data service in the project are listed on the page. You can check whether the view's active data is running.

2. Cannot create tree map views

If you have created a tree map view, you must set the correct 'Value' and 'Default Color'. Ensure that the 'Hierarchy Level' and the Thresholds have been set to correct values.

After saving the query, if there isn't data that can be retrieved with the query, the view must be blank. The Tree map must now be working and retrieving data.

#### **Dashboard Issues**

1. Cannot open the dashboard

If you want to view the dashboard, click the 'Open' menu in the dashboard node. If you cannot see a new popup window, ensure that the browser has not blocked it. You can set browser permissions to allow the window to open pop ups for a particular website.

2. Cannot see the data being refreshed automatically

If you have configured some views with active data, the view should be refreshed automatically. If not, ensure that the query can retrieve the data correctly, and that the active data is running on the view. Also, check if the session hasn't expired.

#### **Data Population Issues**

1. Cannot populate data into the specific data object

The bamloadgen shell script must be put into the directory:

\$ORACLE\_HOME/soa/bam/bin. Ensure that you have set the correct \$JAVA\_HOME. Then, ensure that the Oracle BAM server is running and that the Oracle BAM Composer can be accessed. Check if the data object name in the xml and the data formatter are correct.

2. Cannot set the correct date for the data time field

The data time in the xml file must be the correct formatter if you don't want to use the \$currentTime token. bamloadgen uses "yyyy-MM-dd HH:mm:ss.SSS z". Ensure that the appropriate date formatter has been used in your date value.



CONNECT WITH US

B blogs.oracle.com/oracle

facebook.com/oracle

twitter.com/oracle

oracle.com

Oracle Corporation, World Headquarters 500 Oracle Parkway Redwood Shores, CA 94065, USA Worldwide Inquiries Phone: +1.650.506.7000 Fax: +1.650.506.7200

#### Hardware and Software, Engineered to Work Together

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0914

igsimed igsimed