

## **How we track purchasing commitment with our reporting budget? (Without using encumbrance accounting)**

### **Abstract:**

In our 11.5.10.2 apps installation we are using reporting budget only (without budgetary control and not using encumbrance accounting). It was not possible for our 'Budget & Planning' team to track real-time purchasing commitment. But they needed to know the current commitment for a particular Natural account and Cost center, before approving another new purchase order. This document outlines how we provide our user a workaround to track real-time commitment, without using budgetary control.

### **Solution:**

We used five custom views to summarize necessary data. Finally we used one single SQL to generate output in Oracle Discoverer. The end user simply enters one GL account and the commitment data along with actual and budget is shown. Below is the brief purpose of each view. View definitions are provided at the end of this document.

- a) **bl\_code\_combinations** a custom view to simplify the queries. This returns natural account & Cost center for active CCID's.
- b) **bl\_gl\_actual\_balance** returns YTD actual booking for GL accounts, cost center in functional currency.
- c) **bl\_gl\_budget\_balance** same as above, only return the YTD budget.
- d) **bl\_po\_commitment\_detail** returns detail of approved purchase order or release amount detail till date.
- e) **bl\_po\_commitment** summarize data from the above view. It also converts foreign currency to functional currency. Approval date was used as conversion rate date. Built-in api, `gl_currency_api.convert_amount` was used to convert currency. Already approved purchase order amount is assumed as commitment, for now.
- f) Lastly we used one single SQL to generate commitment tracking output in Discoverer.

### **Output:**

We have finally created following two simple discover report based on a SQL (provided at the end of this document).

1. Cost center wise commitment (for a specific GL account)
2. Commitment for a specific cost center (across whole COA)

Now simply by running above two reports with parameter, end user can view the commitment. It gives them a complete picture of current commitment along with actual and budget amount closely the way FSG suppose to show commitment in case of encumbrance account was enabled. Sample report output is shown below.

Output 1: Cost center wise commitment (for a specific GL account)

---

**Single Account Commitment Tracking**

Period: **Feb-08**

Account Code: 6030851, Ac Name: Other Marketing Expenses

Budget: 2008\_0+12

<b>Cost Center</b>	<b>CC Name</b>	<b>YTD Budget</b>	<b>YTD Actual</b>	<b>YTD Commitment</b>	<b>Variance</b>
A101	CEO Office	300,000.00	0.00	150,000.00	150,000.00
A102	PM Office	0.00	0.00	0.00	0.00
A303	Logistics	0.00	0.00	0.00	0.00
A601	CCD General	1,250,000.00	25,000.00	150,000.00	1,100,000.00
A602	Care Line	0.00	0.00	0.00	0.00
A603	Support Center	0.00	25,000.00	0.00	0.00
A604	Corporate	0.00	0.00	150,000.00	-150,000.00
A711	CCO	0.00	0.00	0.00	0.00
A731	Marketing	1,500,000.00	50,000.00	0.00	1,500,000.00
A751	Sales Office	0.00	0.00	0.00	0.00
A752	Corporate sales	500,000.00	0.00	10,000.00	490,000.00
A753	Regional Sales	0.00	0.00	0.00	0.00
A754	Project Sales	0.00	50,000.00	0.00	0.00
A755	Sales Support	0.00	0.00	0.00	0.00
A756	Trade Marketing	0.00	0.00	0.00	0.00
	<b>Sum:</b>	3,550,000.00	150,000.00	460,000.00	

---

Output 2: Commitment for a specific cost center (across whole COA)

---

**Cost Center Commitment Tracking**

Period: **Feb-08**

CC Name: Marketing, Cost Center: A731

Budget: 2008\_0+12

Account Code	Ac Name	YTD Budget	YTD Actual	YTD Commitment	Variance
6021501	Medical costs	0.00	0.00	0.00	0.00
6021502	Group insurance	150,000.00	150,000.00	0.00	150,000.00
6021503	Medical Insurance	0.00	150,000.00	750,000.00	-750,000.00
6021601	Refreshments costs	0.00	0.00	0.00	0.00
6021602	Staff Recreation	100,000.00	0.00	0.00	100,000.00
6023253	Protocol etc	0.00	0.00	0.00	0.00
6023301	Staff Entertainment	1,000,000.00	0.00	0.00	1,000,000.00
..					
..					
..					
..					
..					
..					
6029106	Logistics charges	0.00	0.00	0.00	0.00
6029109	Office Security expenses	0.00	0.00	0.00	0.00
6029110	Petrol/Mileage costs	1,000,000.00	150,000.00	750,000.00	250,000.00
6029112	Vehicles rental expense	1,000,000.00	150,000.00	750,000.00	250,000.00
6029113	Electricity costs	0.00	-5,000.00	0.00	0.00
6029114	Water expense	0.00	0.00	0.00	0.00
6029115	Telephone expense	1,000,000.00	15000.00	750,000.00	250,000.00
6029116	Postage/Courier expense	1,000,000.00	150,000.00	750,000.00	250,000.00
6029117	Other General expense	0.00	0.00	0.00	0.00
6029119	News paper & Periodicals	0.00	0.00	0.00	0.00
	<b>Sum:</b>	<b>7,000,000.00</b>	<b>4,215,000.00</b>	<b>3750,000.00</b>	

---

**Solution Specific Consideration:**

- In our installation SEGMENT4 is the natural account and SEGMENT3 is the cost center. It can vary.
- The SQL are for 11.5.10.2 specific data dictionaries. For other version it might change.
- Our functional currency is BDT. We had to change the parameter accordingly while calling `gl_currency_api.convert_amount` for currency conversion.
- We used some fixed value like `flex_value_set_id`, `set_of_books_name`, `set_of_books_id`, `currency` etc in query as per our installation. This will vary.
- Current GL period have to be open to appear actual YTD amount.
- We can filter GL account and cost center range anytime as per our need in the `bl_code_combinations` view definition.
- We often run 'Payables Transfer to General Ledger' in Payable and post to reflect most updated actual figured in GL account.
- We are planning to add 'Annual Budget' column in both reports in similar way.
- The output can also be generated in TOAD using the SELECT statements.
- The known limitation is, it has, budget name hard coded to keep it simple. We have to change the budget name in case of new budget is uploaded.

**Conclusion:**

It is not a perfect way for commitment tracking. This is rather an alternative approach to track commitment with fair accuracy; in case of encumbrance accounting is not used. As long it meets our end user requirement, it fine with us.

---

**DISCLAIMER:**

The information in this article is the opinion of the author, not of Oracle Corporation. Any content, materials, information or software downloaded or otherwise obtained through the use of the site is done at your own discretion and risk. Oracle shall have no responsibility for any damage to your computer system or loss of data that results from the download of any content, materials, information or software.

---

## Appendixes: View Definitions & SQL:

### 1. bl\_code\_combinations:

```
CREATE OR REPLACE VIEW bl_code_combinations AS
SELECT DISTINCT cc.segment4, fvv.description "AC_NAME",
                cc.segment3, fvv.description "CC_NAME",
                TO_CHAR(SYSDATE, 'Mon-YY') Period
FROM   gl_code_combinations cc,
       fnd_flex_values_vl fvv,
       fnd_flex_values_vl fvvv
WHERE  cc.chart_of_accounts_id = 50268           -- Change here
AND    fvv.flex_value_set_id = '1009350'       -- Value set for Natural Account
AND    fvv.flex_value = cc.segment4
AND    fvvv.flex_value_set_id = '1009349'      -- Value set for Cost Center
AND    fvvv.flex_value = cc.segment3
-- Range of natural account and cost center
AND    ((NVL(cc.segment3, '') >= 'A000' AND NVL(cc.segment3, '') <= 'A999'))
AND    ((NVL(cc.segment4, '') >= '0000000' AND NVL(cc.segment4, '') <= '9999999'))
AND    cc.enabled_flag = 'Y'                  -- Enable CCID Only
AND    cc.summary_flag = 'N'                  -- Child CCID only
ORDER BY cc.segment4, cc.segment3
```

### 2. bl\_gl\_actual\_balance:

```
CREATE OR REPLACE VIEW bl_gl_actual_balance as
SELECT   bal.period_name, segment3, segment4,
SUM(NVL (bal.year_to_date_dr, 0) - NVL (bal.year_to_date_cr, 0)) YTD_Actual
FROM   glbv_gl_standard_balances bal, gl_code_combinations cc
WHERE  bal.account_id = cc.code_combination_id
AND    bal.set_of_books_id = 1001              -- Change Here
AND    bal.SET_OF_BOOKS_NAME='Banglalink Set of Books' -- Change Here
AND    bal.BALANCE_TYPE='Actual'
AND    bal.currency_type='Functional'
AND    bal.currency='BDT'                      -- Change Here
AND    ((NVL(cc.segment3, '') >= 'A000' AND NVL(cc.segment3, '') <= 'A999'))
AND    ((NVL(cc.segment4, '') >= '0000000' AND NVL(cc.segment4, '') <= '9999999'))
AND    cc.chart_of_accounts_id = 50268
AND    NVL(bal.year_to_date_dr, 0) - NVL(bal.year_to_date_cr, 0) <> 0
AND    bal.period_name = TO_CHAR(SYSDATE, 'Mon-YY')
GROUP BY bal.period_name, cc.segment4, cc.segment3
ORDER BY cc.segment4, cc.segment3
```

### 3. bl\_gl\_budget\_balance\_YTD:

```
CREATE OR REPLACE VIEW bl_gl_budget_balance_YTD AS
SELECT   bal.period_name, segment3, segment4,
SUM (NVL (bal.year_to_date_dr, 0) - NVL (bal.year_to_date_cr, 0)) YTD_Budget
FROM   glbv_gl_standard_balances bal, gl_code_combinations cc
WHERE  bal.account_id = cc.code_combination_id
AND    bal.set_of_books_id = 1001              -- Change Here
AND    bal.set_of_books_name = 'Banglalink Set of Books' -- Change Here
AND    bal.balance_type = 'Budget'
AND    bal.currency_type = 'Functional'
AND    bal.currency = 'BDT'                      -- Change Here
AND    bal.budget_name = 'BL_2008_0+12'         -- Hardcoded, change Here
AND    ((NVL(cc.segment3, '') >= 'A000' AND NVL(cc.segment3, '') <= 'A999'))
AND    ((NVL(cc.segment4, '') >= '0000000' AND NVL(cc.segment4, '') <=
'9999999'))
AND    cc.chart_of_accounts_id = 50268
AND    bal.period_name = TO_CHAR(SYSDATE, 'Mon-YY')
AND    NVL(bal.year_to_date_dr, 0) - NVL(bal.year_to_date_cr, 0) <> 0
GROUP BY bal.period_name, cc.segment4, cc.segment3
```

```
ORDER BY cc.segment4, cc.segment3
```

#### 4. bl\_po\_commitment\_detail:

```
CREATE OR REPLACE VIEW BL_PO_COMMITMENT_DETAIL AS
SELECT poh.SEGMENT1 "PO_NUMBER",
CASE
WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' then por.CREATION_DATE
ELSE poh.CREATION_DATE
END "PO_DATE",
poh.CURRENCY_CODE "CURRENCY",
CASE
WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' then por.AUTHORIZATION_STATUS
ELSE NVL(poh.AUTHORIZATION_STATUS, 'INCOMPLETE')
END "APPROVAL_STATUS", -- Added for release
CASE
WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' then por.APPROVED_DATE
ELSE poh.APPROVED_DATE
END "APPROVAL_DATE", -- Added for release
(CASE
WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' then polc.QUANTITY
ELSE pol.QUANTITY
END ) * polc.PRICE_OVERRIDE "PO_LINE_AMOUNT", -- Added for Release
NVL (
CASE WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' THEN por.APPROVED_DATE
ELSE poh.APPROVED_DATE
END
,
CASE
WHEN pot.DOCUMENT_TYPE_CODE = 'RELEASE' THEN por.CREATION_DATE
ELSE poh.CREATION_DATE
END
) RATE_DATE,
polc.QUANTITY_RECEIVED * polc.PRICE_OVERRIDE "GRS_AMOUNT",
glc.SEGMENT3,
glc.SEGMENT4
FROM po_headers_all poh,
po_lines_all pol,
po_line_locations_all polc,
po_distributions_all pod,
po_document_types_all_tl pot,
gl_code_combinations glc,
po_releases_all por
WHERE poh.PO_HEADER_ID=pol.PO_HEADER_ID
AND pol.PO_LINE_ID=polc.PO_LINE_ID
AND pol.PO_HEADER_ID=polc.PO_HEADER_ID
AND polc.PO_HEADER_ID=pod.PO_HEADER_ID
AND polc.PO_LINE_ID=pod.PO_LINE_ID
AND polc.LINE_LOCATION_ID=pod.LINE_LOCATION_ID
AND por.PO_RELEASE_ID(+) = polc.PO_RELEASE_ID
AND pod.SET_OF_BOOKS_ID=1001
AND poh.TYPE_LOOKUP_CODE=pot.DOCUMENT_SUBTYPE
AND pot.ORG_ID=poh.ORG_ID
AND pot.DOCUMENT_TYPE_CODE IN('PO', 'RELEASE') -- Modified for Release. exclude
agreement
AND NVL(poh.CANCEL_FLAG, 'N') <> 'Y'
AND NVL(pol.CANCEL_FLAG, 'N') <> 'Y'
AND NVL(polc.CANCEL_FLAG, 'N') <> 'Y'
AND NVL(por.CANCEL_FLAG, 'N') <> 'Y'
AND pod.CODE_COMBINATION_ID=glc.CODE_COMBINATION_ID
ORDER BY poh.SEGMENT1
```

#### 5. bl\_po\_commitment:

```
CREATE OR REPLACE VIEW bl_po_commitment AS
```

```

SELECT  segment4, segment3,
SUM(NVL(apps.gl_currency_api.convert_amount (
currency, 'BDT',rate_date,'Corporate',po_line_amount),0)) YTD_PO_Issued
FROM apps.BL_PO_COMMITMENT_DETAIL
WHERE APPROVAL_STATUS='APPROVED'
AND TO_CHAR (APPROVAL_DATE,'YYYY') = TO_CHAR (SYSDATE,'YYYY')
GROUP BY segment4, segment3

```

## 6. Final output SQL:

```

SELECT cc.period,
       cc.segment4 ACCOUNT_CODE, cc.AC_NAME,
       cc.segment3 COST_CENTER,cc.CC_NAME,
       NVL(bal.YTD_Actual,0) YTD_Actual,
       NVL(bdgytd.YTD_Budget,0) YTD_Budget,
       NVL(po.YTD_PO_Issued,0) YTD_Commitment
FROM   apps.bl_code_combinations cc,
       apps.bl_gl_actual_balance bal,
       apps.bl_gl_budget_balance_ytd bdgytd,
       apps.bl_po_commitment po
WHERE  cc.segment4 = bal.segment4(+)
AND    cc.segment3 = bal.segment3(+)
AND    cc.segment4 = bdgytd.segment4(+)
AND    cc.segment3 = bdgytd.segment3(+)
AND    cc.segment4 = po.segment4(+)
AND    cc.segment3 = po.segment3(+)
--AND cc.segment4='4011101'
--AND cc.segment3='A401'
ORDER BY cc.segment4,cc.segment3

```