Earned Value Management Practitioners Forum

Making Life Simpler Using SQL Statements in Cobra

John Fiske
FiskePM LLC

Objectives

- SQL Overview
- Cobra Data Model
- SQL Command Utility
- Copy Project
- Limiting Access to SQL
- Useful Scripts
- Saving Scripts

SQL Overview

What is SQL?

- SQL stands for **S**tructured **Q**uery **L**anguage
- SQL is used to communicate with a database
- ANSI has created standards for SQL so that the language is supported across database types

What is SQL?

- SQL statements are used to perform tasks such as to update data on or retrieve data from a database
- Are many places to learn more about SQL, such as: w3schools

  http://www.w3schools.com/sql/

Free SQL Training at: www.w3schools.com/sql/

SQL Command Utility

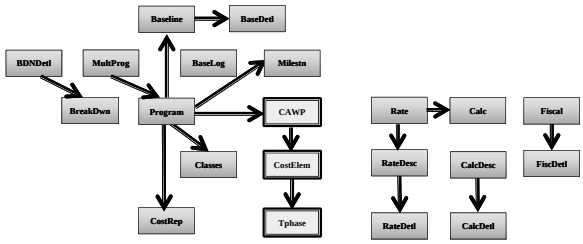
- The Cobra SQL Command Utility is a powerful export and global update utility
- Users can run SQL commands and scripts against the Cobra tables
- Basic SQL statements such as Select, Update, Delete, and Insert are supported

SQL Commands Supported by Cobra

- Cobra supports Select, Update, Delete, and Insert statements
- Select statements copy data from tables to a view format that can be saved to Excel
- Update statements are used to update existing records in a table
- Delete statements are used to delete records from a table
- Insert statements are used to insert new records in a table

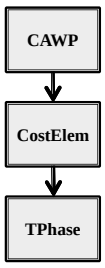
Cobra Data Model

Cobra Data Model

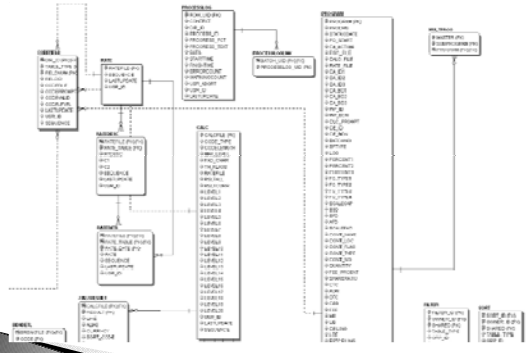


Cobra Data Model, continued

- The CAWP, CostElem, and Tphase tables have a many-to-one relationship with each other
- Other key fields exist in the core tables; the Program and CAPWID are the minimum for returning activity data



Cobra 5.1 ERD.jpg



Cobra 5.1 ERD.jpg, CAWP Table

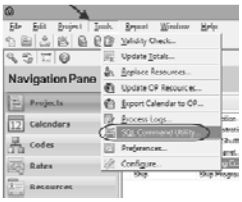
SQL Command Utility

SQL Scripts Provided with Cobra

- There are several sample scripts shipped with Cobra that can be found in the folder: C:\Program Files (x86)\Deltek\Cobra 5 \Samples\Scripts
- The SQL statements shown in this presentation, unless otherwise noted, were prepared for Microsoft SQL Server
- SQL scripts for Microsoft SQL Server and for Oracle differ slightly in syntax and cannot be used interchangeably

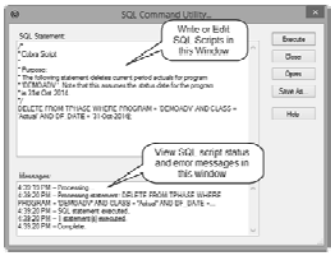
Accessing the SQL Command Utility

- Select Tools > SQL Command Utility from the main menu in Cobra



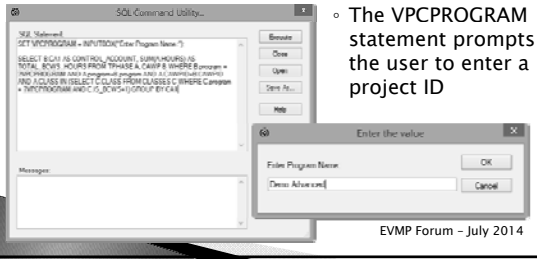
SQL Command Utility

- The SQL Command Utility has two windows; the top section to view/edit SQL scripts and the bottom to display the results of the script once it has been run



Entering SQL Commands

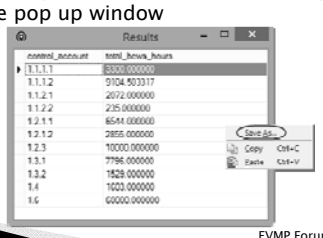
- Multiple statements can be entered in the utility
  - Highlight one statement to run it
  - Highlight none to run them all in the order they appear



- The VPCPROGRAM statement prompts the user to enter a project ID

SQL Statement Results

- The output from the previous SQL statement will appear as shown
  - The output can be saved to a .xls or .csv file by right clicking on the results and selecting "Save As" from the pop up window



Make a Copy of Your Project

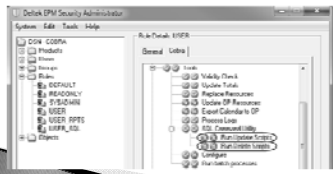
File Save As

- Most of the SQL scripts shown in this presentation will not affect your project
- However, please, any time you attempt to make a change to a project using SQL, first make a copy of the project
  - With the project open, select File, Save As from the main menu

Limiting Access to SQL

Limiting Access to SQL Command Utility

- Access to the SQL Command Utility should be limited to Select statements for most users
- This is accomplished through Role settings in the EPM Security Administrator
  - Specifically, the ability to Update and Delete scripts should be disabled for most users



  - With the settings set as shown, users will still be able to run Select statements

Limited Access to Update and Delete Scripts

- In some instances you may need to allow users limited access to Update and/or Delete scripts
  - For example, one site I support creates a new project each month
  - This site has hundreds of batch files they use each month to prepare CAM notebook reports
  - It was an incredibly time consuming task to replace the project name each month in the batch reports
  - The solution to this issue was to enable some users to run Update scripts to set the program value in the BATCHREP table to the new project name

Limited Update Script Access

- A Custom Menu was set up to execute the following script:
  - UPDATE BATCHREP SET PROGRAM = ?PROGRAM WHERE OWNER_ID = 'LOVETTE' OR OWNER_ID = 'CARRT' OR OWNER_ID = 'WILLIAMSONB'
- A new Role was created that allowed users to perform Update scripts but denied access to the SQL Command Utility

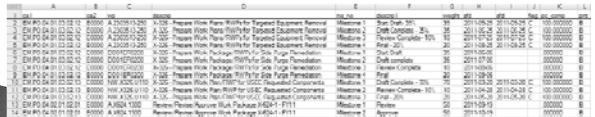
Useful Scripts

Milestone Status List

- The following script could be used to create a milestone status list

```
SELECT a.CA1, a.CA2, a.WP, a.DESCRIP, b.MS_NO, b.DESCRIP, b.WEIGHT,
b.SFD, b.AFD, b.FLAG,b.PC_COMP, a.PMT FROM COBRA!CAWP a,
COBRA!MILESTN b where a.WP<>' ' and a.PROGRAM='PMB0814' and
a.PMT='B' and a.PROGRAM=b.PROGRAM and a.CAWPID=b.CAWPID
ORDER BY a.CA1, a.CA2, a.WP, b.MS_NO
```

- The output would look like:

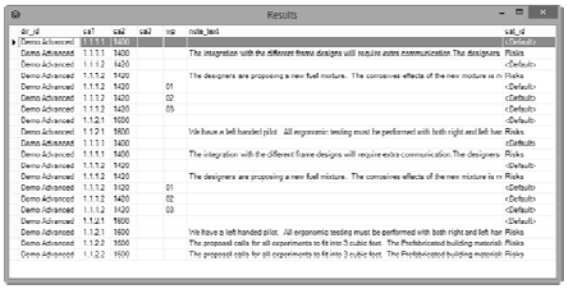


Export / Import Notes

- The following script exports all of the note fields attached to a project
  - The notes can be edited in Excel and then re-imported using the Integration Wizard as Project data

```
SELECT DIR_ID, CA1, CA2, CA3, WP, NOTE_TEXT, CAT_ID FROM
CAWP INNER JOIN COB_NTX ON CAWPID = FK_ID INNER JOIN
COB_CAT ON CAT_UID = ROW_ID WHERE CONTEXT IN ('CA',
'WP') AND DIR_ID = 'Demo Advanced'
```

Export / Import Notes



Change the Level Where Actuals are Loaded

- When a new project is created, the user can specify whether actuals are captured at the CA or WP level
  - As soon as actuals are imported into a project, this setting can no longer be changed within the Cobra menus
- The following script changes the actuals level to CA

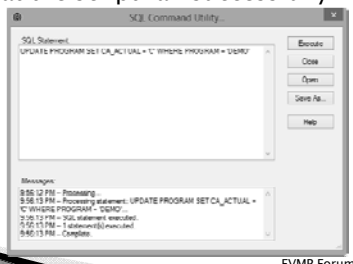
```
UPDATE PROGRAM SET CA_ACTUAL='C' WHERE PROGRAM='xxx'
```

- The following script changes the actuals level to WP

```
UPDATE PROGRAM SET CA_ACTUAL='W' WHERE PROGRAM='xxx'
```


Change the Level Where Actuals are Loaded

- With this type of SQL script you do not expect physical output, but do expect Cobra to tell you that the script ran successfully



Export Resource Assignments

- There is no function within Cobra to export resource assignments for editing, if you want to also change the WP start and/or finish dates at the same time
  - *However*, it is possible to create export files using SQL statements
  - In the following example both a CA/WP file and a resource file will be created
    - These can be combined into one file if you are using a more recent build of Cobra v5
  - These files can then be imported into Cobra using the Integration Wizard after updates have been made to them

Export Resource Assignments, continued

- This statement creates the CA/WP file:

```
SELECT CA1, CA2, CA3, WP,
RTRIM(CA1)+RTRIM(CA2)+RTRIM(CA3)+WP AS ID, DESCRIP, SSD,
SFD FROM COBRA!CAWP WHERE PROGRAM = 'Demo Advanced'
AND WP <> ''
```

- This statement creates the resource file:

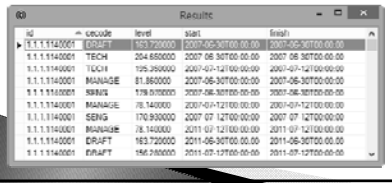
```
SELECT RTRIM(C.CA1) + RTRIM(C.CA2) + RTRIM(C.CA3) +
RTRIM(C.WP) AS ID, T.CECODE, [LEVEL] = CASE WHEN
T.[HOURS] <> 0 THEN T.[HOURS] ELSE T.DIRECT END,
T.DF_DATE AS START, T.DF_DATE AS FINISH FROM CAWP C,
TPHASE T WHERE C.CAWPID=T.CAWPID AND C.PROGRAM =
'Demo Advanced' AND T.CLASS = 'Budget'
```

- *Note:* for large projects, additional conditioning may be added to reduce the number of records

Export Resource Assignments

- Results from first script:

- Results from second script:



Adding or Removing Project Security

- Project security is set up in Cobra on a project by project basis
- For sites which have dozens or hundreds of projects, changing user access to "Read Only" can be an enormous task
  - This is sometimes necessary when the Central Project Controls group is performing processes which effect all projects
  - Each project has to be manually opened and each user's access changed to "Read Only"
- The following SQL scripts automate this process
- These scripts were prepared for Oracle and may need to be rewritten to work with MS SQL

Adding or Removing Project Security, continued

- The following script locks out user write access on all projects that begin with the characters "EM"

```
UPDATE WST_ACL ACL SET ACL.READONLY=1 where
(ACL.READONLY=0) AND (ACL.USR_ID<>' ') AND ACL.DIR_UID in
(select DIR.DIR_UID from WST_DIR DIR WHERE DIR.TABLE_TYPE
='PROGRAM' AND SUBSTR(DIR.DIR_ID,1,2)='EM')
```

- The following script gives users write access on all projects that begin with the characters "EM"

```
UPDATE WST_ACL ACL SET ACL.READONLY=0 where
(ACL.READONLY=1) AND (ACL.USR_ID <>' ') AND ACL.DIR_UID in
(select DIR.DIR_UID from WST_DIR DIR WHERE DIR.TABLE_TYPE
='PROGRAM' AND SUBSTR(DIR.DIR_ID,1,2)='EM')
```

Opening and Closing CAs and WPs

- At times changes may need to be made to closed CAs or WPs
  - The following statement can be used to temporarily open closed CAs and WPs
  - The flags should be reversed once the changes are complete

UPDATE COBRA!CAWP SET FLAG = 'O' WHERE FLAG = 'C' AND PROGRAM = 'IPMB1011'

- When FLAG is blank, the CA/WP hasn't started
- When FLAG = 'O', the CA/WP is open
- When FLAG = 'C', the CA/WP is closed
- Any other character will be ignored by Cobra

Renaming Code Field Values

- The following script can be used to rename all of the code field assignments for a particular value

UPDATE COBRA!CAWP SET C3 = 'PRIOR_COST' WHERE C3 = 'REPLAN' AND PROGRAM = 'PMB1011'

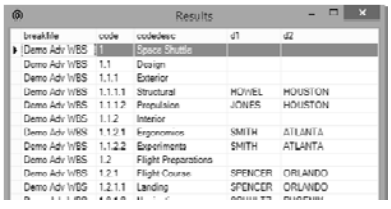
  - *Note:*
    - The code filed values (C1, C2, C3, … C9) are addressed in the same way whether it is a CA code or a WP code
    - C1 can address the first code at the CA level and the first code at the WP level
    - It is important, therefore, to never attach the same code file to the same code value at different levels of the project

Code File Select Script

- The format of the following script can be used to examine any code file and the codes assigned to it
  - In this example, we're looking at the Demo Advanced project's WBS file

SELECT BREAKFILE, CODE, CODEDESC, D1, D2 FROM BDNDETL WHERE BREAKFILE ='Demo Adv WBS'

Code File Select Script



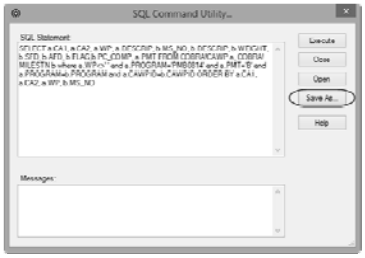
Saving Scripts

How To Save Your Script

- Once you've spent time creating a unique SQL script, it's always a good idea to save a copy of the script for later use
- Do not save these in the C:\Program Files (x86)\Deltek\Cobra 5 folder as these files can be overwritten when a new release of Cobra is loaded
- If the scripts are to be shared with others, save them to a network folder everyone can access

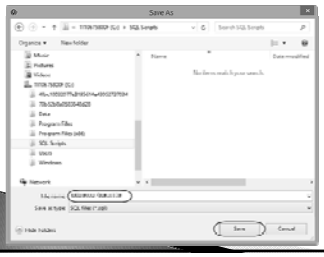
Saving Your SQL Scripts

- Select the Save As button on the SQL Command Utility dialog box



Saving Your SQL Scripts

- Navigate to the folder where you will save your scripts
- Enter a descriptive name in the "File name" window and then click on the Save button



  - The file can be saved with either a .sql or .txt file extension
  - An advantage of the .sql extension is that these scripts can be run from the Custom Menu

Retrieving Saved SQL Scripts

- To retrieve a saved SQL script, click on the Open button in the SQL Command Utility

Retrieving Saved SQL Scripts

- Navigate to the location of the saved SQL script, select the script, and click on Open

Retrieving Saved SQL Scripts

- The saved script will now appear in the SQL Statement window from which you can modify and/or execute the script

Questions?

John Fiske
FiskePM LLC
john@fiskepm.com
Cell (864) 640-6198