Proven Practices and Plug&Play Design Template for

Journal Entry Adjustment Functionality

with

IBM Cognos TM1

Prepared:
April 2015

By:
Andreas Kugelmeier
Senior Managing Consultant,
Business Analytics Architect
IBM Analytics
Mobile Phone: +1-215-384-7302
Email: kugelmeier@us.ibm.com

Notices & Disclaimers

Document Version History

Date	Version	Author	Description
4/21/2015	1.0	Andreas Kugelmeier	First Version

Table of Contents

1. About this Document
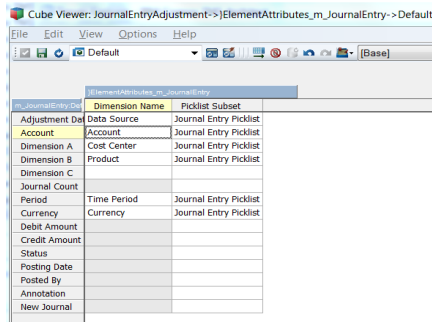
This document outlines Proven Practices for the design of a Journal Entry Adjustment model for TM1 by describing Plug&Play template for Journal Entry Adjustments. The template is available as a zip file in the section <Design Template Objects>.

2. Journal Entry Model

2.1   Cubes & Rules

Journal Entry.cub, with Dimensions:
- **Version**, with Versions Actual, Forecast, Plan, …; keeping the Version dimension in the JE Adjustment model allows for (i) easier JE Adjustment by Version, including What-If, (ii) quick access of archived Journal Entry Details for Historical Versions, (iii) flexibility when setting up security for Journal Entries by including Version security.
- **TM1 Journal Number**[1], containing the Journal No
- **Line Item**, containing the Journal Entries per Journal No
- **m_Journal Entry**, containing the Journal entry cube measures:
    - Adjustment Data Source: target data source for the adjustment. Data Source (or 'Book') is a dimension in the target cube that is used to delineate data by its origins
    - Account: Account No/Name
    - Dimension A, B, C: Dimension A/B/C Element Name. The Dimension Name for Dimension A/B/C is to be entered under m_JournalEntry Element Attribute 'Dimension Name'. In the sample model, dimension 'Cost Center' is configured to be 'Dimension A', dimension 'Product' is configured to be 'Dimension B' & no dimension is configured to be 'Dimension C'. Picklists for the dimension elements are derived by the Subset name entered under Attribute 'Picklist Subset':

    - Journal Count: =1 in Line Item 0 of a valid journal (used to count # of Journals)
    - Period: YYYYMM; period for which JE Adjustment is to be posted
    - Currency: Currency for which the adjustment is to be made

    - Debit Amount: $ amount, Total of Debits needs to equal Total of Credits
    - Credit Amount: $ amount, Total of Debits needs to equal Total of Credits

    - Status: Status of Journal, Status = 'Posted' for Posted Journals
    - Posting Date: posting date & time stamp
    - Posted By: TM1 User Name of user who posted the Journal

    - Annotation: Free form comment/annotation for Journals
    - New Journal: for information purposes only when Copying Journals; For the originating Journal No, the new (copied) Journal No will be displayed here
    - Last Journal: Under 'All Journals', 'All Line Items', will show the Last Journal that was posted

---

[1] The Prefix TM1 is recommended to accommodate Cognos Controller/TM1 environment. Cognos Controller will create and maintain a dimension called 'Journal Entry.dim' as part of the FAP model which stages Controller data in TM1. The 'Journal Entry.dim' cannot be used for TM1 originated Journal Entry Adjustments, hence a 'TM1-owned' dimension is needed.

Journal Entry.RUX

```
feedstrings;
SKIPCHECK;

#region String rules
[{'Actual','Forecast','Plan'},'All Journals','All Line Items','Last Journal'] = S: stet;
[{'Actual','Forecast','Plan'},'Last Journal'] = S: '';
[{'Actual','Forecast','Plan'},'All Journals'] = S: '';
[{'Actual','Forecast','Plan'},'All Line Items'] = S: '';
#if no Dimension Name (attr. of measures dim 'm_JournalEntry') specified or if specified Dimension does not exist, make blank
[{'Actual','Forecast','Plan'},{'Adjustment Data Source','Account','Dimension A','Dimension B','Dimension C'}] = S:
              IF ( DB ( '}ElementAttributes_m_JournalEntry', !m_JournalEntry, 'Dimension Name') @= '' % DIMIX ( '}Dimensions', DB( '}ElementAttributes_m_JournalEntry',
!m_JournalEntry, 'Dimension Name')) =0 ,
                            '',
                            continue);
#'New Journal' entry will be written by TI to Line item 0
[{'Actual','Forecast','Plan'},'Line Item':'0',{'Status','Posting Date','posted by','New Journal'}] = S: stet;
[{'Actual','Forecast','Plan'},{'Status','Posting Date','posted by','New Journal'}] = S: '';
#Enter Journal Period and Currency into Line Item 0 only
[{'Actual','Forecast','Plan'},'Line Item':'0',{'Period','Currency'}] = S: stet;
[{'Actual','Forecast','Plan'},'Line Item':'0'] = S: '';
#Bring in period and currency from Line Item 0 for valid Journal Entry Line items
[{'Actual','Forecast','Plan'},{'Period','Currency'}] = S:
              IF ( DB('Journal Entry',!Version,!TM1 Journal Number, !Line Item,'Debit Amount') = 0
                  & DB('Journal Entry',!Version,!TM1 Journal Number, !Line Item,'Credit Amount') = 0,
                            '',
                            DB('Journal Entry', !Version, !TM1 Journal Number, '0', !m_JournalEntry));
#block 'All Line Items'
[{'Actual','Forecast','Plan'},{'Account','Dimension A','Dimension B','Dimension C'},'All Line Items'] = S: '';
#endregion

#region N-Rules
# set Journal Count to 1 if there is a Total Debit and a Total Credit Amount
[{'Actual','Forecast','Plan'},'Line Item':'0','Journal Count'] = N:
   IF ( DB('Journal Entry',!Version,!TM1 Journal Number,'All Line Items','Debit Amount') <> 0
                  & DB('Journal Entry',!Version,!TM1 Journal Number,'All Line Items','Credit Amount') <> 0,
                            1,
                            0);
[{'Actual','Forecast','Plan'},'Line Item':'0'] = N: 0;
[{'Actual','Forecast','Plan'},'Journal Count'] = N: 0;
#only allow for Credit or Debit Amount entry if the Journal Line Item has been defined entirely,
# for all applicable dimensions and with valid elements
[{'Actual','Forecast','Plan'},{'Debit Amount','Credit Amount'}] = N:
          IF (
          DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Account') @= ''
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension A', 'Dimension Name') @<> ''
                  & DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension A') @= '')
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension B', 'Dimension Name') @<> ''
                  & DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension B') @= '')
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension C', 'Dimension Name') @<> ''
                  & DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension C') @= '')
          % DB('Journal Entry', !Version, !TM1 Journal Number, '0', 'Period') @= ''
          % DB('Journal Entry', !Version, !TM1 Journal Number, '0', 'Currency') @= ''
          % DIMIX ( DB ( '}ElementAttributes_m_JournalEntry', 'Account', 'Dimension Name'), DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Account')) = 0
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Adjustment Data Source', 'Dimension Name') @<> ''
                  & DIMIX ( DB ( '}ElementAttributes_m_JournalEntry', 'Adjustment Data Source', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Adjustment Data Source')) = 0 )
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension A', 'Dimension Name') @<> ''
                  & DIMIX ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension A', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension A')) = 0 )
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension B', 'Dimension Name') @<> ''
                  & DIMIX ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension B', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension B')) = 0)
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension C', 'Dimension Name') @<> ''
                  & DIMIX ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension C', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension C')) = 0)
          % DTYPE ( DB ( '}ElementAttributes_m_JournalEntry', 'Account', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Account')) @= 'C'
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Adjustment Data Source', 'Dimension Name') @<> ''
                  & DTYPE ( DB ( '}ElementAttributes_m_JournalEntry', 'Adjustment Data Source', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Adjustment Data Source')) @= 'C')
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension A', 'Dimension Name') @<> ''
                  & DTYPE ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension A', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension A')) @= 'C')
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension B', 'Dimension Name') @<> ''
                  & DTYPE ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension B', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension B')) @= 'C')
          % ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension C', 'Dimension Name') @<> ''
                  & DTYPE ( DB ( '}ElementAttributes_m_JournalEntry', 'Dimension C', 'Dimension Name'),
                            DB('Journal Entry', !Version, !TM1 Journal Number, !Line Item, 'Dimension C')) @= 'C'),
                            0,
                            continue);
# do not allow a Debit posting for a Line item that already has a Credit Posting (either Credit OR Debit per Line Item)
[{'Actual','Forecast','Plan'},'Debit Amount'] = N: IF ( ['Credit Amount'] <> 0 , 0 , continue );
#endregion

FEEDERS;

[{'Actual','Forecast','Plan'},'All Line Items',{'Debit Amount','Credit Amount'}] => ['Line Item':'0','Journal Count'];
[{'Actual','Forecast','Plan'},{'Credit Amount','Debit Amount'}] => ['Period'],['Currency'];
```

Finance.cub & Finance.RUX:

A simple example model to receive Journal Entry Adjustments. Contains a simplified Balance Sheet and Income Statement in Dimension 'Account.dim'.
The cube rules are similar to the rules in 'TM1 Rolling Forecast and Planning Artifact.docx' (yet include a Balance Sheet hierarchy as well as Rules for Retained Earnings)

2.2 Workbooks

The workbook 'Journal Entry.xlsx' may be used for entry, copying and posting of Journal Entries:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3							Current Journal: 2		Target Journal: 3		you can only copy into non-posted Targe Journals				
4		Version	Actual			Post Journal		Copy Journal	Last Journal: 2	Target Period: 201402	enter a target period for your new Journal				
5		TM1 Journal Number	3						Reverse?	N	set to 'Y' to reverse the entries in the target journal				
6		STATUS	2015-APR-21 5:46:16 p.m. on Admin by												
7		Measure:	Adjustment Data Source	Account	Dimension A	Dimension B	Dimension C		Period	Currency	Debit Amount	Credit Amount	Annotation	Status	Posting Date	Posted By	New Journal
8		Dimension Name:	Data Source	Account	Cost Center	Product										
9	0							201401	USD				Posted	2015-APR-21 5:46:16 p.m.	Admin	
10	All Line Items									1000	-1000					
796	786	Journal Entry Adjustments	Inventories	Corp4713	not assigned		201401	USD	1000	0					
797	787	Journal Entry Adjustments	Trade receivables	Corp4714	not assigned		201401	USD	0	-1000					
798	788								0	0					
799	789								0	0					
800	790								0	0					
801	791								0	0					
802	792								0	0					
803	793								0	0					
804	794								0	0					
805	795								0	0					
806	796								0	0					
807	797								0	0					
808	798								0	0					
809	799								0	0					
810	800								0	0					

Journals can only be entered for months > the 'Actuals Through Date' attribute value of the Version you are posting against. I.e. you are only allowed to make JE posting for 'Open' periods.

The Workbook leverages the two main Journal Entry processes 'Journal Entry – post.pro' and 'Journal Entry – Copy.pro':

2.3 Journal Entry Processes

2.3.1 Journal Entry – Post.pro

Will post a JE in cube 'Journal Entry'
Parameters:
- pVersion: Version
- pJournalNumber: Journal No to Post

Journals need to be balanced. Currently, the model is configured for GL-based entry, i.e. credits are entered as negative values and debits as positive values. The sum for all Debits in a Journal Entry Adjustment therefore needs to be = the sum of all Credits. For adjustment models where the entry is in Financial statement signs (all positive), change the + operator in the red row below to negative:

```
nTotalDebit = roundp ( CELLGETN ( sSourceCube, pVersion, pJournalNumber, 'All Line Items', 'Debit Amount'), 2 );
nTotalCredit = roundp ( CELLGETN ( sSourceCube, pVersion, pJournalNumber, 'All Line Items', 'Credit Amount'), 2 );
nDiff = NumberToString ( nTotalDebit + nTotalCredit );
IF (nDiff @<> '0');
   sError = 'Journal Entries (Debits & Credits) do not balance!';
   Processbreak;
ENDIF;
```

2.3.1.1 Adjustments to process 'Journal Entry – Post.pro' for posting to target cube

TI-process Prolog Tab: adjust/change the target cube name where highlighted in red below:

```
ExecuteProcess ( 'SYS_IBM_SetConstant_ClientID' );
StringGlobalVariable ( 'TM1ClientID' );

ProcessName = GetProcessName();
sDateTime = TIMST(NOW, '\Y\m\d\h\i\s');
#ExecuteProcess ( 'SYS_IBM_Process_Log_Create_Entry', 'pProcess', ProcessName );

#gLogFileDirectoryPath  = CellGetS ( 'SYS_IBM_Control', 'Process Log Directory Path' , 'S Type' ) ;
#gPrefix = CellGetS ( 'SYS_IBM_Control', 'SystemObjectsPrefix' , 'S Type') ;
#gExportPath = CellGetS ( 'SYS_IBM_Control', 'Outbound Data Directory Path' , 'S Type') ;
#gImportPath = CellGetS ( 'SYS_IBM_Control', 'Inbound Data Directory Path' , 'S Type') ;

#LogFile = gLogFileDirectoryPath | TM1ClientID  | '_Process_' | ProcessName | '_' | pVersion | '_Journal_' | pJournalNumber;
#LogFileProlog = LogFile | '_Prolog.txt' ;
#LogFileData = LogFile | '_Data.txt' ;
#LogFileMetaData = LogFile | '_MetaData.txt' ;
#LogFileEpilog = LogFile |  '_Epilog.txt' ;

sError = '';
sJournalNumberDimension = 'TM1 Journal Number';

sSourceCube = 'Journal Entry';

############## ENTER TARGET CUBE ######################
sTargetCube = 'Financials';
#################################################
```

TI-process Data Tab: adjust/change the CellPutN formula to the dimensionality of the target cube where highlighted in red below:

```
IF( sStatus @<> 'Posted');
  vAccount =   CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Account');
  vDimensionA =  CELLGETS ( sSourceCube, pVersion,  pJournalNumber, vLineItem, 'Dimension A');
  vDimensionB =  CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Dimension B');
  vDimensionC =   CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Dimension C');
  vCurrency = CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Currency');
  vPeriod =   CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Period');
  vAdjustmentdataSource = CELLGETS ( sSourceCube, pVersion, pJournalNumber, vLineItem, 'Adjustment Data Source' );

  ##################### ADJUST TO TARGET CUBE DIMENSIONALITY ###############################
  CellIncrementN ( vValue, sTargetCube, vAccount, vDimensionA, vDimensionB, vAdjustmentdataSource, pVersion, vPeriod, vCurrency );
  #####################################################################################

  CellPutS ( pJournalNumber, sSourceCube, pVersion, 'All Journals','All Line Items','Last Journal');
ENDIF;
```

2.3.2   Journal Entry – Copy.pro:

Will copy a Journal (not the entry) to a new Journal.
Parameters:
- pVersion: Version
- pJournalNumber: Source Journal to be copied
- pTargetPeriod: target period in YYYYMM
- pReverse: If set to Y, entries form source journal will be reversed in target journal
- pTargetJournal: the number of the Target Journal. Only non-posted Journals can be a target Journal

2.4   Sub-processes

The following Sub-Processes are leveraged by 'Journal entry – Post.pro' & 'Journal Entry – Copy.pro':[2]

SYS_IBM_SetConstant_ClientID.pro:
>   Sub-process that creates a unique Client ID string based on the result of the TM1USER() function. The Client ID string is used to create unique, client-specific temporary subsets and views.

SYS_IBM_Subset_Create.pro:
>   Sub-process that creates a subset based on parameter input. Main sub-process to 'SYS_IBM_View_Create.pro':

'SYS_IBM_View_Create.pro':

>   Sub-process that creates a view based on parameter input.

SYS_IBM_View_Destroy.pro:
>   Sub-process that Destroys a view and its temporary system subsets.

2.5   Other Objects

The template leverages the Time Dimension template as described in 'Proven TM1 Practices for Continuous Time Period Dimension Design and Time-related Analysis incl Design Template.docx'.

It is not necessary to use this nor any similar type of time dimension. Any type of time dimension(s) can be used. If a non-continuous time dimension is used (separate year and month dimensions), the code in 'Journal Entry – post.pro' will have to be adjusted accordingly.

---

[2] See asset "Misc. TM1 Utilities for Cube & Dimension Management" for more information on the following sub-processes

2.6 Security

2.6.1 Element Security

We recommend to implement ElementSecurity for the dimensions
Version.dim (or equivalent) & m_JournalEntry (or equivalent). Write-back to the Journal Entry
model/cube shall hence be governed by
- i)   Journal Entry cube security (read, write, none)
- ii)   Version Element Security (to allow granular security by Version/Scenario)
- iii)   m_JournalEntry Element Security (to explicitly lock/prevent write-back to Journal status and posting measures

2.6.2 Cell Security

For the Journal Entry cube, Cell Security is used to block write-back to a Journal once a Journal was
posted. The following rule was put in place for Cell Security Cube '}CellSecurity_Journal Entry.cub':

```
skipcheck;

[{'Adjustment Data Source','Dimension A','Dimension B','Dimension A','Account','Period','Currency','Debit Amount','Credit Amount','Annotation'}] = S:
        IF ( DB('}ElementSecurity_m_JournalEntry', !m_JournalEntry, !}Groups) @= 'WRITE'
                & DB('}ElementSecurity_Version', !Version, !}Groups) @= 'WRITE',
                        IF ( DB('Journal Entry', !Version, !TM1 Journal Number, '0', 'Status') @= 'Posted',
                                'READ',
                                IF ( DB('}ElementAttributes_Version', !Version, 'Version Status') @= 'Locked',
                                        'READ',
                                        continue)),
                        continue);
```

3. Design Template Objects

JournalEntryAdjustmentTemplateTM1Data.zip