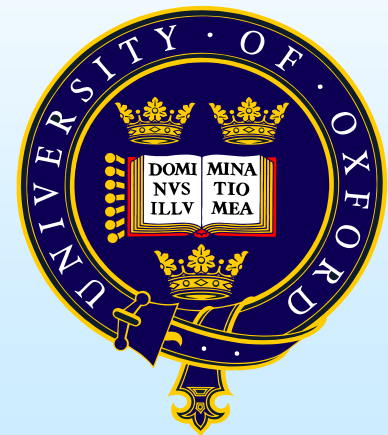


Update on COMA CB: Metadata for Conditions DB Management



Elizabeth Gallas
Oxford

December 1, 2011



COMA_CB Development Documentation

Devel Doc moved to TWiki (from gallas.web.cern.ch/):

<https://twiki.cern.ch/twiki/bin/viewauth/Atlas/ComaCoolMetadataDev>

- **Link to ppt with COMA_CB Schema tables**
 - 3 versions (labelled September, October, November)
 - October version currently on ATLR
 - Account, Folder definitions, Global Tags
 - NO: Folder Tags or their Global associations
 - November version currently in development INT8R
 - Tables created
 - Loading program in development
- Status of November version: next slides
- **Link to Table/Column Documentation**
- **Details of data loading**
 - Loading order, “Business Rules” (what is excluded), what triggers updates to loaded data (in principle)

COMA Loading Program Operations

- “COMA loading”:
 - Is a series of customized loading programs loading one/more tables in the COMA schema
 - Executed once per day (~6:30 CET) OR “on demand”
 - Based on current operations (cron)
 - “COMA_CB loading” is one of many loading programs
 - For table/columns with Conditions DB related sources
- “COMA_CB loading” has 4 stages.
 1. Account related tables loading
 2. Folder related tables loading
 3. Folder Tag table loading
 4. Global Tag related tables loading
- Some COMA_CB tables have “external” sources
 - Those tables need “other” entry programs

November 2011: COOL_CB Conditions DB Management Metadata

CB_Instances
 P – CBI_NAME
 CBI_DESC

CB_Schemas
 P – CBS_NAME
 CBS_SYSTEM
 CBS_DESC

CB_OnOffs
 P – CBO_NAME
 CBO_DESC

CB_Owner_Instances
 P – CBOI_INDEX

 U – OWNER_NAME
 FU – CBI_NAME

 F – CBO_NAME
 F – CBS_NAME

CB_GT_TO_OIS
 P – CBG2O_INDEX

 FU – CBGT_INDEX
 FU – CBOI_INDEX

 SYS_INSTIME

CB_GTAGS
 P – CBGT_INDEX

 U – TAG_NAME

 TAG_LOCK_STATUS
 TAG_DESCRIPTION
 SYS_INSTIME

 CBGT_INSTIME
 CBGT_RANK

CB_GTag_Infos
 P – CBGTI_INDEX

 FU – TAG_NAME
 U – STATUS_SINCE

 STATUS_UNTIL
 STATUS_SUMMARY
 NEXTGTAG_ID
 USED_FOR_TASK
 Deployment_Location

CB_PColumns
 P – CBP_INDEX

 FU – CBF_INDEX
 U – CBP_COLUMN

 CBP_TYPE
 CBP_DESC

CB_NODES
 P – CBF_INDEX

 FU – CBIO_INDEX
 U – NODE_FULLPATH

 NODE_NAME
 NODE_ID
 NODE_PARENTID
 NODE_ISLEAF
 NODE_INSTIME
 LASTMOD_DATE

 CBF_LASTMOD_DATE

CB_GT_TO_FTS
 P – CBG2F_INDEX

 FU – CBGT_INDEX
 FU – CBFT_INDEX

CB_REFERENCES
 PF – CBF_INDEX

 FOLDER_IOVTABLENAME
 FOLDER_TAGTABLENAME
 FOLDER_IOV2TAGTABLENAME
 FOLDER_CHANNELTABLENAME

NODE_DESCRIPTION
 FOLDER_VERSIONING

 CBF_NODE_PATH
 CBF_IOV_BASE
 CBF_ATT_TYPE
 CBF_CHAN_COUNT
 CBF_PAYLOAD_COUNT

CB_FTAGS
 P – CBFT_INDEX

 FU – CBF_INDEX
 U – TAG_NAME

 TAG_LOCK_STATUS
 TAG_DESCRIPTION
 SYS_INSTIME

 CBFT_INSTIME
 CBFT_ROWCOUNT
 CBFT_LAST_OBJTIME
 CBFT_SINCE_TIME
 CBFT_UNTIL_TIME

CBAMI_GTAGS
 P – TAG_NAME

 IS_ACTIVE
 DATASET_COUNT
 DATE_FIRST_DATASET
 DATE_LAST_DATASET

CBAMI_GTAG_USES
 TAG_NAME
 Project_Name
 Run_Start
 Run_End

 AMITag_Array ?

COOL_CB Folder Related Tables

Folder related tables are loaded SECOND

Insert:

- All Folders meeting criteria (add new when found)

Update:

- **Chan_Count** each time program is executed
- Other table/columns when **LastMod_Date** changes

CB_PColumns

P – CBP_INDEX
FU – CBF_INDEX
U – CBP_COLUMN
CBP_TYPE
CBP_DESC

CB_REFERENCES

PF – CBF_INDEX
FOLDER_IOVTABLENAME
FOLDER_TAGTABLENAME
FOLDER_IOV2TAGTABLENAME
FOLDER_CHANNELTABLENAME

CB_NODES

P – CBF_INDEX
FU – CBIO_INDEX
U – NODE_FULLPATH
NODE_NAME
NODE_ID
NODE_PARENTID
NODE_ISLEAF
NODE_INSTIME
LASTMOD_DATE
CBF_LASTMOD_DATE
NODE_DESCRIPTION
FOLDER_VERSIONING
CBF_NODE_PATH
CBF_IOV_BASE
CBF_ATT_TYPE
CBF_CHAN_COUNT
CBF_PAYLOAD_COUNT

CB_GT_TO_FTS

P – CBG2F_INDEX
FU – CBGT_INDEX
FU – CBFT_INDEX

CB_FT_TAGS

P – CBFT_INDEX
FU – CBF_INDEX
U – TAG_NAME
TAG_LOCK_STATUS
TAG_DESCRIPTION
SYS_INSTIME
CBFT_INSTIME
CBFT_ROWCOUNT
CBFT_LAST_OBJTIME
CBFT_SINCE_TIME
CBFT_UNTIL_TIME

Dashed tables:
Not created yet!

CBAMI_GT_TAGS

P – TAG_NAME
IS_ACTIVE
DATASET_COUNT
DATE_FIRST_DATASET
DATE_LAST_DATASET

CBAMI_GT_TAG_USES

TAG_NAME
Project_Name
Run_Start
Run_End
AMITag_Array ?

COOL_CB Folder Tag Table

CB_Instances

P – CBI_NAME
CBI_DESC

CB_Schemas

P – CBS_NAME
CBS_S
CBS_D

CB_

P – CBO
CBO_DE

CB_PC

P – CBP_I
FU – CBF
U – CBP_C

CBP_Type

CBP_TYPE
CBP_DESC

CB_REFERENCES

PF – CBF_INDEX

FOLDER_IOVTABLENAME
FOLDER_TAGTABLENAME
FOLDER_IOV2TAGTABLENAME
FOLDER_CHANNELTABLENAME

CB_Owner_Instances

P – CBOI_INDEX
U – OWNER

CB_GT_TO_OIS

P – CBG2O_INDEX

CB_GT_TAGS

CBGT_Infos

INDEX

ME
INCE

MARY

SK

ation

not made yet.

Dashed tables:
Not created yet!

CBAMI_GT_TAGS

P – TAG_NAME

IS_ACTIVE
DATASET_COUNT
DATE_FIRST_DATASET
DATE_LAST_DATASET

CBAMI_GTAG_USES

TAG_NAME
Project_Name
Run_Start
Run_End

AMITag_Array ?

Folder Tag table is loaded THIRD

Insert:

- All Folders Tags for existing folders (?)
- Add new folder tags when program is re-executed

Update:

- All columns for unlocked tags each time program is executed (?)
- ? I am unsure how well this will perform

NODE_ID
NODE_PARENTID
NODE_ISLEAF
NODE_INSTIME
LASTMOD_DATE

CBF_LASTMOD_DATE

NODE_DESCRIPTION
FOLDER_VERSIONING

CBF_NODE_PATH
CBF_IOV_BASE
CBF_ATT_TYPE
CBF_CHAN_COUNT
CBF_PAYLOAD_COUNT

CBG2F_INDEX

FU – CBGT_INDEX
FU – CBFT_INDEX

CB_FT_TAGS

P – CBFT_INDEX


FU – CBF_INDEX
U – TAG_NAME

TAG_LOCK_STATUS
TAG_DESCRIPTION
SYS_INSTIME

CBFT_INSTIME
CBFT_ROWCOUNT
CBFT_LAST_OBJTIME
CBFT_SINCE_TIME
CBFT_UNTIL_TIME


COOL_CB Tables from AMI

- AMI team: involved in many aspects of COMA
- Create COMA_CBAMI_* tables
 - AMI team populates (grant write access)
 - Join to other COMA_CB tables with Global Tag Name
- Solveig did some test queries
 - Demonstrates proof of principle !
 - Can populate
 - Which Global Tags have been used/not
 - Gives a measure of “popularity”
 - Date range of usage
 - For Data AND Monte Carlo
 - Additional tables also possible
- Suggest I create this table
 - AMI team can start test loading



CBAMI_GTAGS
P – TAG_NAME

IS_ACTIVE
DATASET_COUNT
DATE_FIRST_DATASET
DATE_LAST_DATASET



CBAMI_GTAG_USES
TAG_NAME
Project_Name
Run_Start
Run_End

AMITag_Array ?

COOL_CB_Infos Table for Conditions Tag Management

- This is one of the tag related tables
 - suggested by [Andrea Formica](#)
- I can create this table, grant write access to Cool Tag Management
- Column names/types are up to you
 - I copied example columns from [Andrea's draft tables](#)
- Join to other COMA_CB tables with Global Tag Name
- Populate with information from your system
 - Can display Current and Next in interfaces...
 - Can store the history of Current and Next
- Scope of usage is up to COOL Tag Management:
 - Initially: Use a secondary “copy” for use by interfaces
 - Up to you how this aspect evolves with time
- AMI team may be able to help with entry interface
 - Needs specifications, constraints

CB_GTag_Infos

P – CBGTI_INDEX

FU – TAG_NAME

U – STATUS_SINCE

STATUS_UNTIL

STATUS_SUMMARY

NEXTGTAG_ID

USED_FOR_TASK

Deployment_Location

Adding (more) Structural Metadata ?

Structural Metadata → Data about the containers of data

- Examples already in the schema:
 - `CB_Nodes.Chan_Count`, `CB_Ftags.RowCount`
 - Recent email thread with DBAs
 - Discuss: adding quantities like Folder size
- We should discuss what to add !

- Useful for:
 - Conditions DB management
 - Conditions DB Browsing
- Current Plan
 - Get loading program working
 - Develop queries to add
 - Refine most useful quantities
 - Adding columns not difficult

CB_FTAGS

P – CBFT_INDEX

FU – CBF_INDEX

U – TAG_NAME

TAG_LOCK_STATUS

TAG_DESCRIPTION

SYS_INSTIME

CBFT_INSTIME

CBFT_ROWCOUNT

CBFT_LAST_OBJTIME

CBFT_SINCE_TIME

CBFT_UNTIL_TIME

CB_NODES

P – CBF_INDEX

FU – CBIO_INDEX

U – NODE_FULLPATH

NODE_NAME

NODE_ID

NODE_PARENTID

NODE_ISLEAF

NODE_INSTIME

LASTMOD_DATE

CBF_LASTMOD_DATE

NODE_DESCRIPTION

FOLDER_VERSIONING

CBF_NODE_PATH

CBF_IOV_BASE

CBF_ATT_TYPE

CBF_CHAN_COUNT

CBF_PAYLOAD_COUNT

- We all see many used cases for COMA_CB
 - Provide links from external systems (AMI) and documentation, Improve browsing (broad!), Manage Conditions DB generally and COOL tagging, ease/motivate clean up/uniformity
- This is work in progress, many aspects converging ...
 - Some remaining aspects of loading not resolved
- Lesson: nothing is “final” until:
 - Initial data is loaded
 - Update “rules” and implementation reaches closure

→ and even then, nothing is “final”

 - Interface development may drive longer term changes
 - As interfaces evolve: functionality, usability, performance and additional metadata may be added
- A lot of progress made so far
 - Thanks for patience and feedback ...