

IBM Software Group

ALM UCM Composite Baselines

April 18, 2008

Stuart Poulin stuart.poulin@us.ibm.com 206-686-9824

Rational. software

Disclaimer and Non-Disclosure

- IBM Rational Plans are subject to change
- Nothing in this document is intended to create any representations or warranties
- IBM Rational warranties are contained in the applicable license agreements



Demo to Present

- ▶ Using UCM Composite Baselines to create ALMBaseline records



Audience

- Release Engineers
- Program Managers



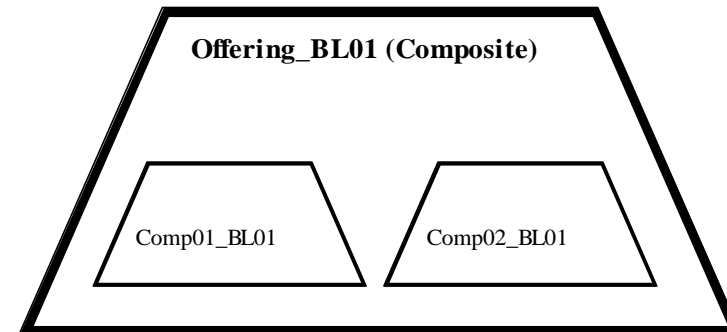
Prerequisites

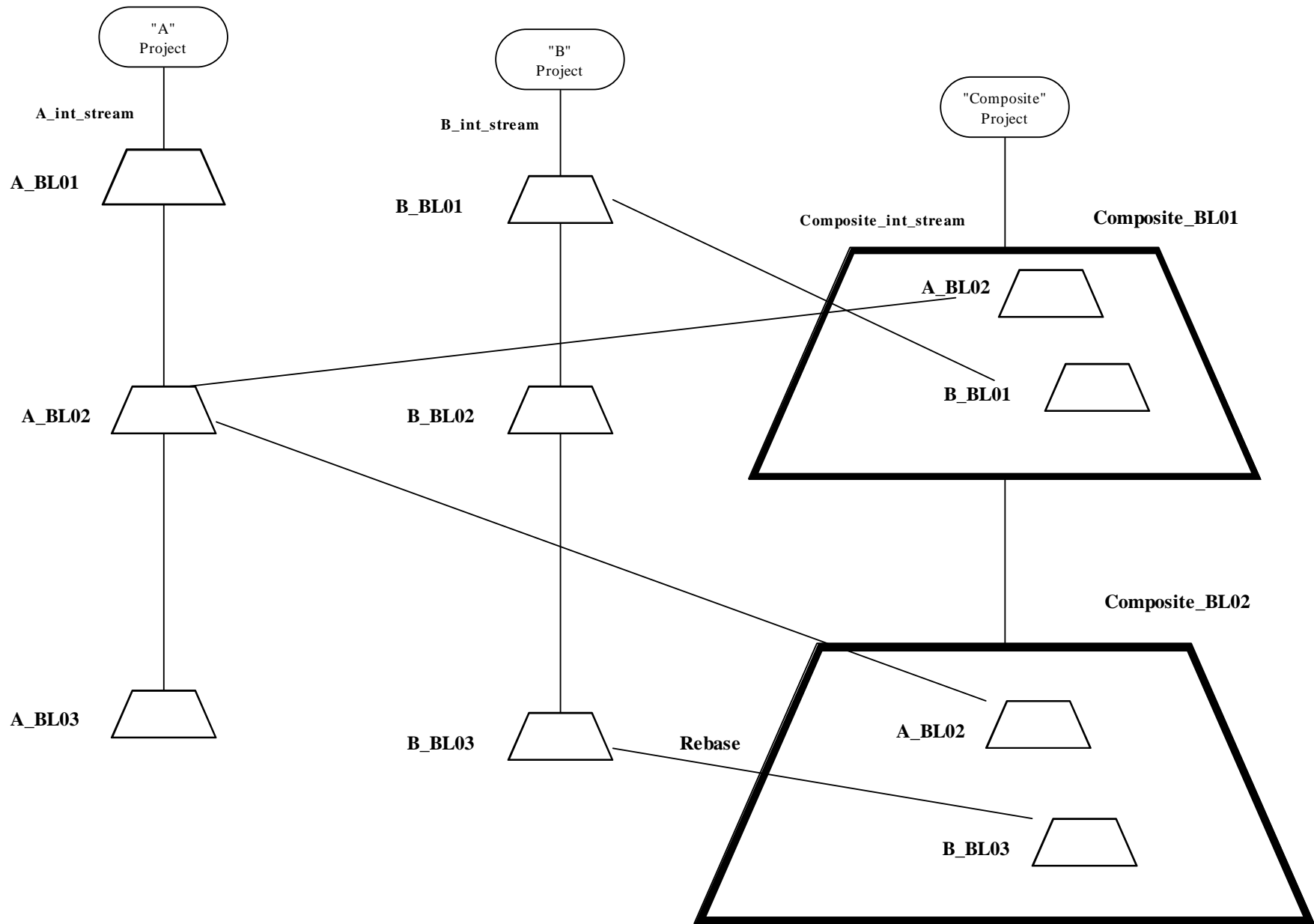
- View a CQ ALM demo explaining the overall high level ALM work flow and features.
- View ALM UCM ToDo demo showing use of UCM and ALM together



UCM Composite Baseline Review

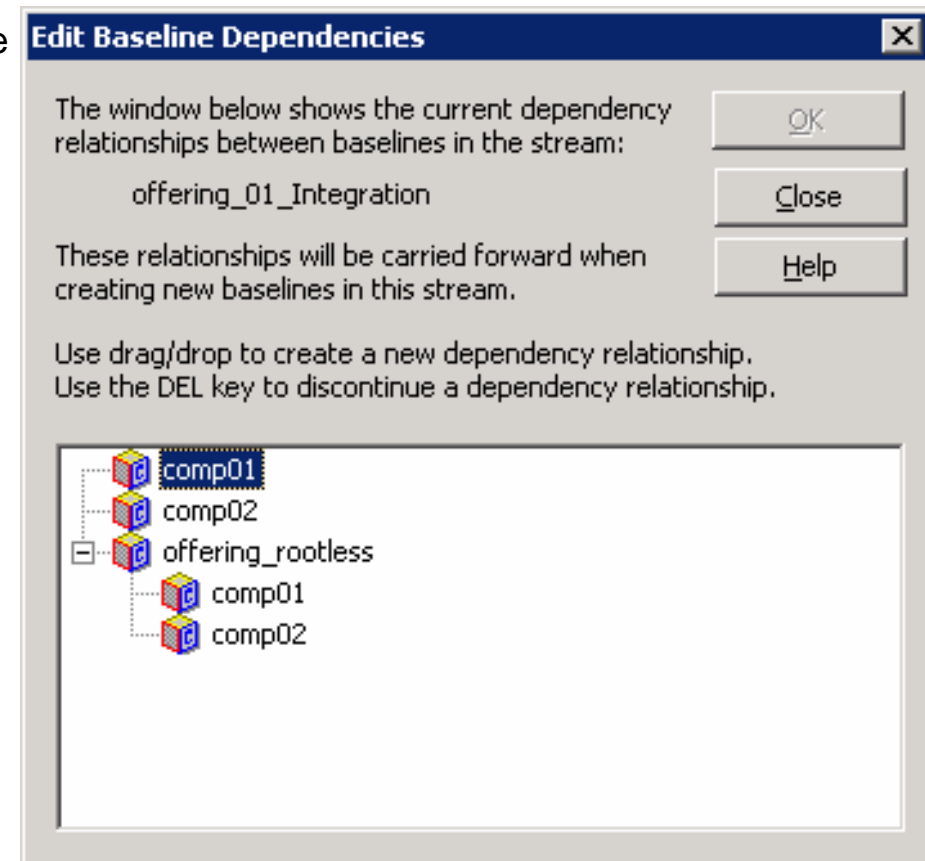
- Composite baselines group other baselines together (Baselines shown as trapezoids)
- Composite Baselines are “Composed of” other baselines
- Next slide shows a couple of projects feeding baselines to a project that includes them in composite baselines
- Baselines can be mixed and matched
- Baselines that fail to test can be reverted while others advance
- Many advantages to Composite baselines explained in ClearCase UCM documentation





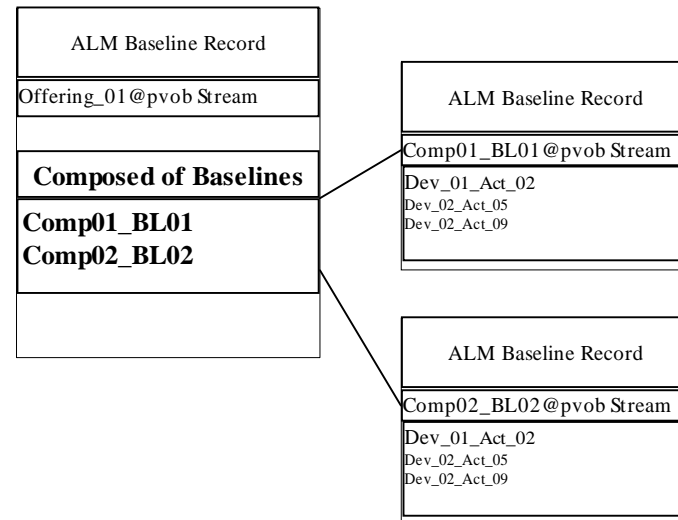
How are Composite Baselines tracked?

- UCM Composite Baselines are Baselines that are “composed” of other baselines.
- The UCM Composite Baseline groups member baselines under a single baseline.
- To create a UCM Composite Baseline, UCM baseline dependencies are first set, the top level UCM Component that will house the Composite baseline is set to depend on member baselines.
- In this example, the offering_rootless UCM component is set to depend on baselines from the comp01 and comp02 UCM components



How Composites are tracked

- Detection of composite baselines is automatically done by the ALM create_baseline_record.pl script.
- If the UCM Baseline includes “member” baselines it is determined to be a UCM Composite Baseline.
- The member baselines of the UCM composite baseline will be placed into the “Composed of Baselines” field on the ALM Baseline record.



ALMBaseline record “Composed of Baselines”

View ALMBaseline stu_pvob_080130 03_ALM_Project_ToDoOffering_75 (ProjectAdmin_ToDo,CQMS.ALMM3.GALWAY@ALM)

Baseline | Activities | History

Baseline Information Obsolete

Project: Add Remove

id	Category	Release	Is Obsolete?
ALM00000553	Category_ToDoOff...	ReleaseLab...	No

Project	RoleLabel	Primary
ALM00000553	Release...	ReleaseE...

Name:

ucm_stream:

PVOB or Loc: Owner:

Promotion Level:

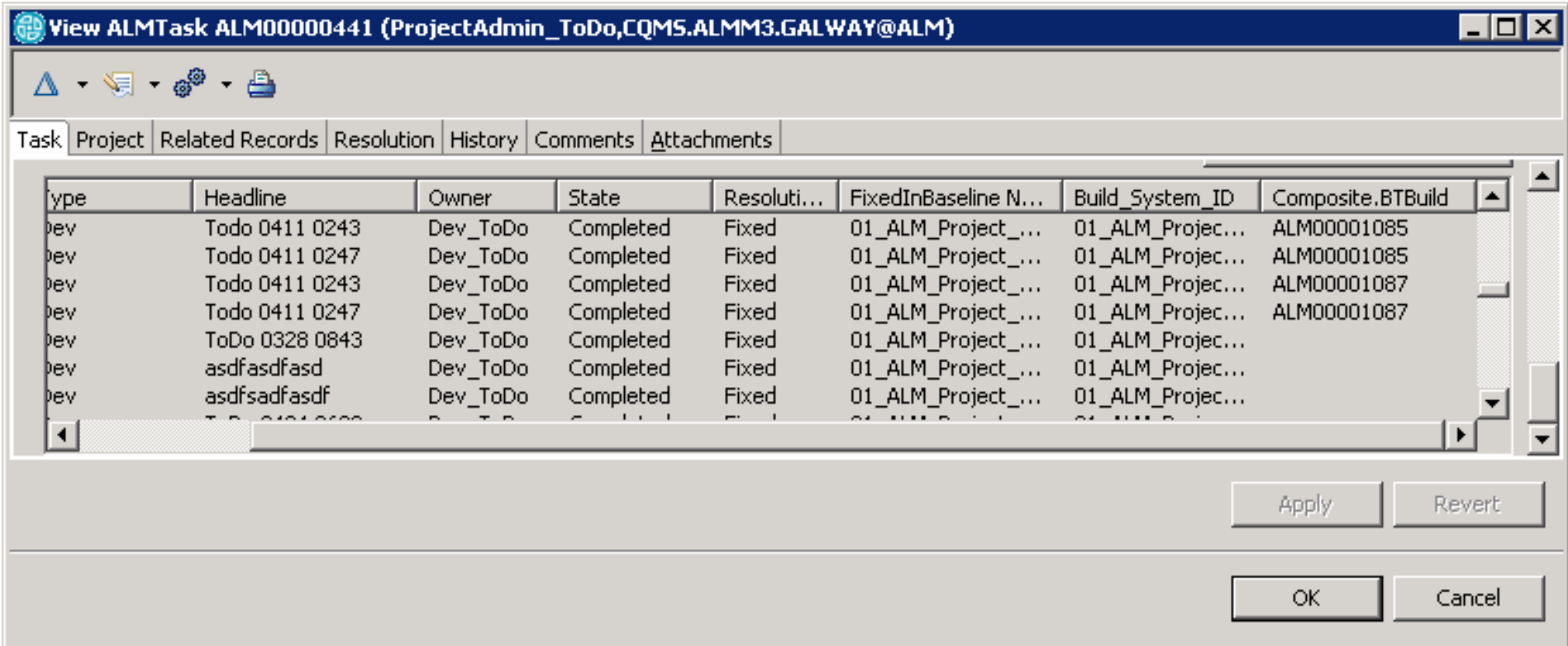
Composed Of Baselines: Add New... Remove

Baseline Name	PVOB_OrLocat...	Is Obsolete	Project Name	Project Category	Project Release
01_ALM_Project_ToDo_78	stu_pvob_080...	NO	Project_ToDo	Category_ToDo	ReleaseLabel0...
02_ALM_Project_ToDoNAS_13	stu_pvob_080...	NO	Project_ToD...	Category_ToD...	ReleaseLabel0...



ALMTasks show “Composite BTBuild”

Finds activities in Builds off Composite Baselines



View ALMTask ALM0000441 (ProjectAdmin_ToDo,CQMS.ALMM3.GALWAY@ALM)

Task Project Related Records Resolution History Comments Attachments

type	Headline	Owner	State	Resoluti...	FixedInBaseline N...	Build_System_ID	Composite.BTBuild
dev	Todo 0411 0243	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	ALM00001085
dev	Todo 0411 0247	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	ALM00001085
dev	Todo 0411 0243	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	ALM00001087
dev	Todo 0411 0247	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	ALM00001087
dev	ToDo 0328 0843	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	
dev	asdfasdfasd	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	
dev	asdfasdfasd	Dev_ToDo	Completed	Fixed	01_ALM_Project_...	01_ALM_Projec...	

Apply Revert

OK Cancel



Examples Projects supplied with ALM

■ Project_ToDoOffering

- ▶ “Offering” or “Product” level project
- ▶ Uses Composite Baselines to group components into an offering



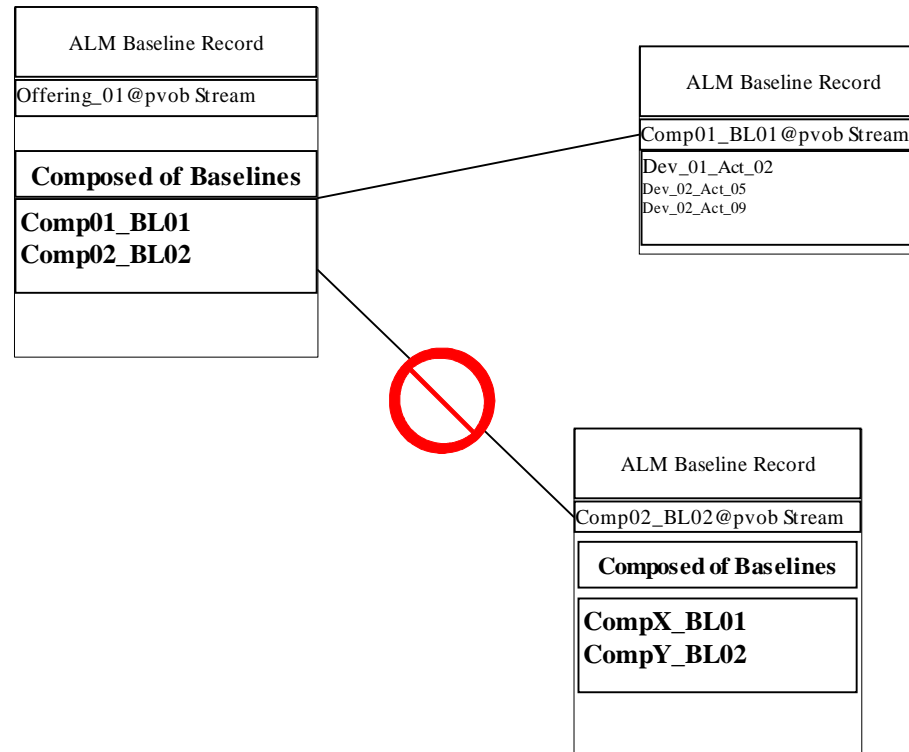
Build Forge Project

■ O1_ALM_Project_ToDo

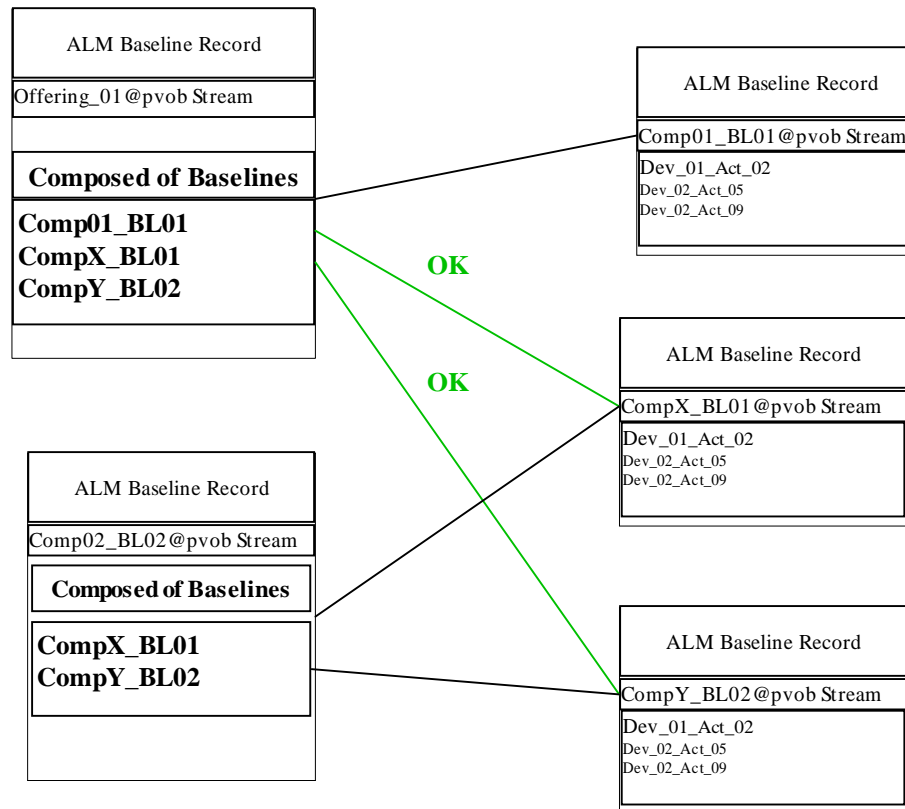
- ▶ **Start the Build**
- ▶ Review Build steps
 - UCM mkbl – Standard UCM Baseline is made
 - Create_Baseline_Record
 - Runs a diffbl between this new baseline and the last ALMBaseline record stored in ALM.
 - Minimal extra information need be supplied as script can gather information from both ALM and ClearCase UCM
 - Mock source build – builds a web page example
 - Create_Build_Record
 - Given the above Baseline, one or more builds may be created
 - ALM reuses BTBuild record
 - Review in ALM the new Baseline, Activities in the Baseline, BTBuild



Only UCM Composite Baselines of Ordinary Baselines are currently supported. Composite of Composite baselines are not currently supported



- ▶ It is required that any Composite ALM Baseline record you want to include into another Composite first be decomposed into ordinary ALM Baseline records, then a Composite of these ordinary ALM Baselines be made.



Methods to decompose Composite baselines

■ In UCM

- ▶ Set foundation baselines, and baseline dependencies to point to the originating ordinary baselines for the composite baseline
- ▶ Can be time consuming and will likely need another project

■ In ALM

- ▶ Use the options to create `create_baseline_record.pl`
 - add_composed_of_baseline <baseline>
 - add_pvob <pvob>Must be supplied in pairs.
Supply one pair for each originating baseline



How do ALM Activities from *Composed of Baselines* (UCM Member Baselines) appear on the ALM Task records?

- An ALM BTBuild record is created off the ALM Baseline Record.
- Each BTBuild record has a unique ALM ID. The display of ALM Activities on the ALM Task record will include a header of Composite.BTBuild.Build_System_ID. Entries in this column will display the ALM Build_System_ID (name) of the BTBuild created off the ALM Composite Baseline.



Release oriented versus Component Oriented projects.

- UCM Composite Baselines can be used in either type of orientation.
- ALM Composite Baseline records can also be used in either type of orientation, but with the restriction that composites of composites are not currently supported
- For a release oriented project, ordinary baselines used on a main UCM integration stream will pickup all activities delivered to it from any substream. Therefore on the activity you would see the builds created off the substream and later, the builds created off the top integration stream
- Activities are never lost. They will report to any baseline they appear in as long as all projects use the UCM ALM integration.
This includes:
 - Activities delivered via *interproject rebases*
 - Activities delivered via *deliver-baseline*
 - Activities delivered via *Alternate deliveries*

