Comparison of LO-VC model to AVC Model

Steve Schneider, Sr. Consulting Product Data Analyst

sschneid@steelcase.com
For more than 105 years, Steelcase Inc. has helped create great work, education and healthcare experiences for the world's leading organizations. Our family of brands, including Steelcase®, Coalesse®, Designtex®, PolyVision®, Turnstone® and AMQ™, offer a comprehensive portfolio of furniture and technology products and services. Steelcase is globally accessible through a network of dealers, including over 800 Steelcase dealer locations, and is a publicly traded company with fiscal 2018 revenue of $3.1 billion.
If you did not attend this presentation, much of what I spoke about is contained in the slide notes.
This presentation will compare 2 similar Product models. One built using the traditional LO-VC and the other using the new AVC (Advanced Variant Configuration). An overview of the master data differences between both models will be shown along with some important differences that are not master data related.

**All Data Models are built in a S/4HANA 1809 release On Premise system!**
Much of this presentation is based on what can be found in the newly released Improvement List noted in this Forum post.

While I actually built these models and presentation before the list was released, it follows along pretty closely.

Note that the presentation itself was reviewed and approved by the colleagues at SAP.
What Will Be In This Presentation

- Change of Material Length from 18 to 40
- Change of Characteristic Length from 30 to 70
- Actions going away (replaced with procedures but still working in the AVC model)
- Classification as a selection condition going away
- Overview page for VC Modeler
- Processing Modes
- Negative Domain Restriction syntax (including Variant Tables)
- All characteristics (Including Multi Value) being restrictable
- Pre-Conditions acting like constraints
- Characteristic groupings
- New AVC Syntax checks
- Alternative values
- More Precise rounding
- BOM Explosion
- Intermediate variable values
For reference, the comparison slides will be built in this manner, with the LOVC data on the left and the AVC data on the right. When compared in this manner each data column will be labeled as shown here.

This same Bicycle data model will potentially be utilized in future presentations on topics such as Loose Coupling.
What is in the next few slides are things that are not specific to the AVC. They are part of the general S/4HANA releaser but are relevant to understand when moving your VC model into S/4HANA
S/4 HANA can go up to 40 characters. Note that you will get a warning about EDI not functioning correctly if you exceed 35 characters.. Do so at your own risk!

All models shown in this presentation are no more than 35 characters.
Note that you must extend the material master past 18 characters before you will be able to use the new extended Characteristic Value length.
Existing actions will continue to work in the S/4HANA LOVC model. You cannot create new actions. Note that the screenshot labeled LOVC is actually from and ECC6 EHP 6 system. Only the right hand screenshot is from S/4HANA
Existing classifications used as a selection condition will continue to work in the S/4HANA LOVC model. You cannot create new instances of this solution. Note that the screenshot labeled LOVC is actually from and ECC6 EHP 6 system. Only the right hand screenshot is from S/4HANA.
AVC Related Differences
Overview Page for VC Modeler
The Overview page is something brand new. It is only available in Fiori. From this overview page you will be able to get a “one stop” shop of things that are relevant to you as a product modeler.

Note that things such as classes and characteristics are independent of the engine that processes them. Therefore they will show on this screen as changed objects if they are used in an AVC data model (assuming they have actually been changed).
Processing Modes
The processing mode is a critical piece of the new data model. It all starts with defining what processing mode will be used at the $Root Configuration profile.

When Classic is selected, the data model will be processed using the tried and true LO-VC engine.

When Advanced is selected, then the data model will be processed using the new AVC engine.

Note that the relevant model checks in the PMEVC (detailed later) are also based on this $root selection.

It is not possible to “Mix and match” processing modes in a multi level model. The engine will “inherit” the highest level processing mode. However I strongly recommend the processing mode be consistent throughout the entire data model to be exploded.
Object dependencies also need a processing mode based on the new and continued expansion of the AVC syntax. You will find the processing mode on constraint nets and constraints. Within constraint nets, the processing mode of the constraint will be inherited from the net.
You will also find the processing mode on “standard” object dependencies.

It is critical to note what engine a given object dependency will be used in.

For example, while the syntax of an existing Classic mode may be exactly the same as the AVC dependency, HOW that dependency is processed may be different. This is especially the case for pre-conditions, which are processed like a constraint in the AVC, meaning (for example) that order of execution is not relevant.

You might notice that my AVC screen changed to look like the ECC screen. My old eyes cannot tolerate the default SAP S/4HANA Screen color and layout... Sorry!!
Domain Restrictions Using Constraints
Within the LO-VC engine, the concept of “truth” is important. Meaning that in the LO-VC world, everything is a “positive” statement in order to restrict domains. This is a direct result of something called the TMS (Truth Maintenance System) in the LO-VC Engine.
However, in the world of AVC, you can use negative expressions in order to restrict domains. Notice that I use the Not Equal expression in the restrictions.

Also notice that there is no inference section necessary. At this point, ALL characteristics will ALWAYS be inferred.

By the way, the TMS is not in the AVC engine. It has been replaced with a new constraint solver and mathematical formulas.
Note Characteristic Reuse across both models..

By the way, characteristics do NOT need to be marked as “Restrictable” in the AVC engine. ALL characteristics are considered as restrictable in the AVC...
Notice what happens when I try to utilize a Not Equal domain restriction.... Fail...
In this case, I am using a negative table call. Notice the AVC syntax uses something called “FALSE IF”...

Next look at the tables involved
The result is that both domains are restricted in exactly the same manner.
Multi Value Characteristic Domain Restrictions
In the LO-VC world, the only way to do a domain restriction on a multi value characteristic is by using pre-conditions. In the AVC, multi value characteristics are restrictable... What does this mean to us?

Here is a demonstration of using constraints to restrict the domain.
Must set constraint C_AVC_BICYCLE_MV_OPT to Released
Must set constraint C_AVC_BICYCLE_MV_OPT_EQ to locked
Of course, if you REALLY, REALLY like preconditions on values.. They still work! But a tad differently..

It really, really needs to be experienced in the simulation environment.. For the purposes of this conversation, be very, very careful with preconditions using the SPECIFIED statement in the AVC.
You can use either the EQ or NE syntax to restrict the domains... And get the same results. But make sure you address ALL the values!!

Must set constraint \texttt{C\_AVC\_BICYCLE\_MV\_OPT\_EQ} to Release
Must set constraint \texttt{C\_AVC\_BICYCLE\_MV\_OPT} to locked
Pre-Conditions in the AVC
Here is a side by side syntax of preconditions in both LOV and AVC
Notice that the Frame Size has not yet been specified and therefore the value of HARDRACINGSEAT is not available.
Notice that the Frame Size has not yet been specified but the value of HARDRACINGSEATANDREALLYLIKEELONGVALUES is available for selection.
Pay really close attention to that Improvement listing. It has nice little tidbits like this in there...
Characteristic Groups For user Interface Design
Characteristic groups are a new concept designed to make creating an interface design easier. These are now NEW master data elements that can be mixed and matched as desired across models.

Note that in the AVC configuration profile, there is a new button. Clicking it will take you to
A new Fiori application that allows you to create new characteristics groupings or assign existing groupings.

You define the groupings of characteristics how you want including the sort order of the characteristics within the groups.

If you assign multiple characteristic groups, you can then define the sort order of the groups.

At this point in time, ECM is not enabled... But be advised that adding ECM control is in the planned enhancements.
Notice you get much the same result.
• Ability to maintain assignment of Characteristic Groups to Configuration Profiles with ECM
• Ability to maintain assignment of characteristics within characteristic groups with ECM

As noted, ECM is coming.. When?? Good question... 😊
AVC Syntax Check
Sometimes you will perform a standard Object dependency syntax check on an AVC model (from within the PMEVC), then try to go to simulation and you’ll get this wonderful screen.

This screen is a tip...
That you have done something in your model that is specifically not allowed in the AVC.. You can find out what those things are by running the AVC model check.. This is executed from within the standard Model Check tool (menu path EXTRAS\CHECK MODEL or the keyboard combination of CTRL+F2)
If you have done something wrong (like bind a multi value cstic to a variant table) you will see an error message as shown. Fix you model, re-run the check tool and your simulation will execute fine.
Alternative Values
In LO-VC, the user initially sees the restricted domain of values. Then, after selecting a value, if the user presses the F4 key, they get the selected value.

Of course, you can see the entire domain of values by selecting No entry, then hitting the Enter key.
Notice in this case that the original display is the domain of restricted values. Once the user has selected a value, using the F4 key shows them the entire set of domain restricted values (not the global set). What is also pretty cool is the inspector shows you the global set, including those values that were excluded by domain restriction.
Improved Mathematics & Rounding
Note that for this particular item, there are no master data differences between LO-VC and AVC.

It’s all in how the underlying engine does math.
Notice the exact same formula
Note that the target ctsic is a numeric format at 15 digits. The new engine rounds the last digit based on the new math algorithms.
More Precise Rounding

**Precision and Rounding**

AVC uses up to 34 digits for computation results

- $1.23456789876543 \times 3.45678987654321$ can be computed exactly
- That's $4.2676418143575609510750492303$ (in case you wondered)

Obviously, that is not always enough: $X = 1/3$

- AVC applies advanced algorithms to further control rounding errors (based on IEEE'1788-2015 standard)

This slide was taken from a presentation given at ASUG. For whatever reason, this slide is not in the posted deck.
Bill of Material Explosion
In LOVC, there are really 2 explosions happening..

One for the high level configuration

Another for the low level configuration. Think BOM explosion and Routing determination.

While AVC still does both, HOW it does the low level configuration has changed..

This impacts what you can do with procedures in the BOM.

Note the cstice “BOM_CSTIC”

In LOVC the value was set by the procedure in the BOM

In AVC, the value was not set
Notice that in the AVC simulation trace, you can filter on high or low level explosion.

In this example, I have it set to high level
Notice the 2 executions on the LOVC side

And the 0 executions on the AVC side...
Now notice the AVC side. I changed the trace detail to show procedures executing in low level configuration.
And look.. In the AVC, the procedure in the BOM is only executed by low level configuration

As a result, no values are set into the BOM_CSTIC.. Because value assignment is not possible in low level config
Intermediate Variable Values
Some of you may use temporary variables in high level config. Or as I called them in the presentation referenced here, Intermediate values

You can do this in LOVC all day long...

But not in the AVC...

In the AVC, in order to do a value assignment, all characteristics must be assigned to a class which is assigned to the configurable material.
Things I Didn’t Cover
The items highlighted in Green on this roadmap are what I discussed. Along with many of the innovations already released in 1709 that are NOT on this slide.

What is NOT highlighted in green are those items I did NOT discuss.

You need to pay attention to this slide during every CWG conference as the results will be different (especially for the Cloud).
In case you are wondering, the roadmap has changed... Notice we are out to 2021 now, along with a couple of new items... Like that “Configuration Object” thingy....

Watch for further updates!
Questions?
Thank You!
Please remember to fill out the Conference Survey
CWG Information (Use for non-CWG presentations)

- The **ONLY** international user group for SAP Configuration technology
- **THE** best place to influence SAP on the direction of their Configuration technology
- Multiple forums, technical articles, and blogs to get answers to your pressing business questions about utilizing SAP Configuration technology
- **Two** annual conferences, a spring conference in Europe and a fall conference in the Americas
- Membership is **FREE**, but restricted to SAP employees, customers, and partners which accept, respect and follow our bylaws.
  - [http://www.configuration-workgroup.com/node/1850](http://www.configuration-workgroup.com/node/1850)