A Generic Data Exchange Format for Variant Management

Michael Himsolt
Daimler AG
Wilhelm-Runge Str. 11
michael.himsolt@daimler.com

1. EXTENDED ABSTRACT

Tools for variant management frequently interact with artifacts such as model based specifications, program code, or requirements documents. This is often a two-way communication: variant management tools import variability information from an artifact, and in return export variant configurations. For example, they need to gather information about the variation points that are contained in the artifact, need to know which variants are already defined, and then modify existing or define new variants ("this variation point stays, that one goes away") or even define new variation points ("items a,b and c are alternatives"),

There is currently no standardized API available for such operations. Hence, a variant management tool needs to implement a separate interface – and possibly a new data format as well – for each new artefact. Worse, each variant management tool needs to do this separately. With m variant management tools and n artefacts, this may require the implementation of up to m×n different interfaces.
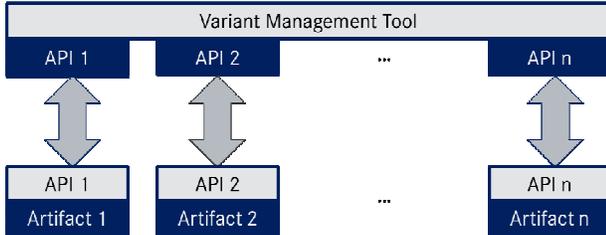

Figure 1 Without VariabilityAPI: Many APIs need to be implemented

In this document, we present a generic API (VariabilityAPI) that allows variant management tools to communicate with artifacts through a standardized interface. If implemented across both variant management tools and artifacts, this may reduce the number of required implementations significantly. A generic interface typically also lowers the barrier for adding new artifacts, and fosters the introduction of new tools for variant management.
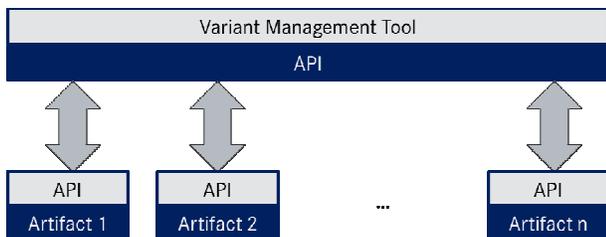

Figure 2 With VariabilityAPI: only a single API needs to be implemented

The VariabilityAPI serves two purposes. First, it provides a generic description of the variation points that are contained in an artifact. Variation points may come in two flavors:

- Variation points may be locations in an artifact which are removed or set inactive in a binding process. This is implemented by defining a condition for each variation point.

- Variation points may be parameters. Such variation points provide expressions which are used by the binding process to assign a value to the parameter.

Variation points may also exhibit dependencies; for example a set of variation points may be designated as a set of alternatives, which means that all but one of them will be removed during the binding process (although the actual semantics of the binding process is beyond the scope of this concept).

Second, the VariabilityAPI can define specific variant configurations. In our context, a variant configuration is an assignment of fixed values to the conditions or expressions that are associated with variation points.

Finally, the VariabilityAPI defines a number of operations for exporting and importing variability information. It is not expected that every artifact or variant management tool implements the full set of operations, hence the API provides means to state the capabilities of a specific implementation.

The VariabilityAPI uses an XML based serialization as its data exchange format. The specification is scheduled for release in 2015 within the SPES XT project.

2. ACKNOWLEDGEMENTS

3. REFERENCES

[1] Große-Rhode, M., Himsolt, M., and Schulze,M., 2013: Generische Variantenschnittstelle, in: *Initiale Methodik für ein durchgängiges Variantenmanagement und Wiederverwendung im Engineering von Embedded Systems*, SPES XT Project Deliverable