Simply Consistent

By Franz Lohberger, Visu-IT! GmbH

ASCET connects with global data dictionary

Especially with the new XML interface, connecting ASCET with external data sources, such as data dictionaries, is a breeze. This opens the door to the use of externally managed data declarations also in simulations and code generation with ASCET. Not only does this make the development process in ASCET easier and faster; it also provides for consistency in data declarations across process boundaries. Because the data declarations are archived globally and managed in a central data dictionary, they are available for use in a variety of subsequent process steps (single source).

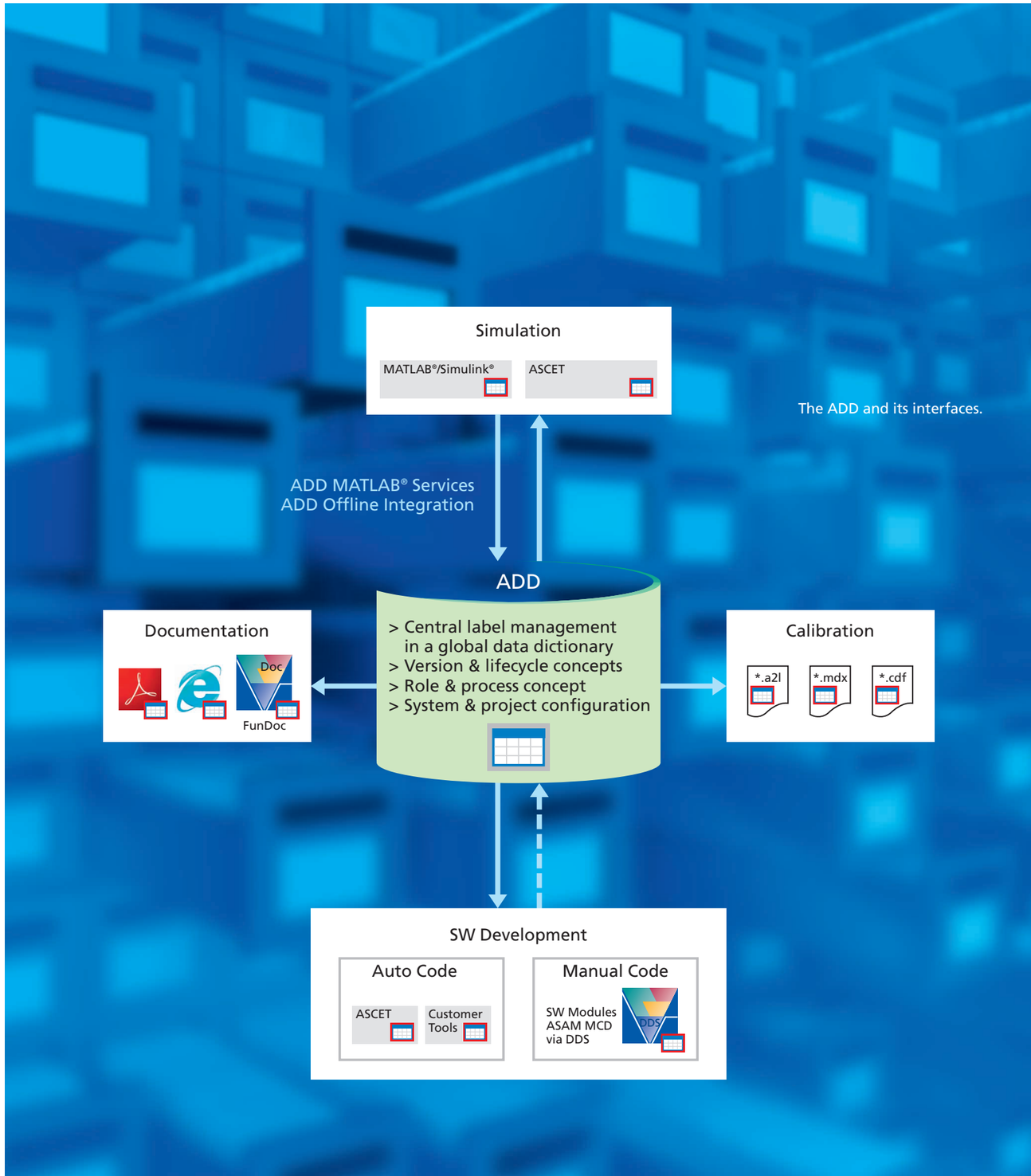
As a case in point, a connection with a central data dictionary is in productive deployment at Audi's Ingolstadt facility, where the preferred choice is the Automotive Data Dictionary (ADD) by Visu-IT! of Regensburg. The ADD supports a host of centralized data dictionary functions:

- Centralized database with global, company-wide accessibility (also across several company locations).
- Version handling and Lifecycle concept.
- Roles-and-rights concept.
- Data object, function, and project view.
- Project configuration by means of system constants, plus variant coding.
- Safeguarding of naming conventions & terminology (e.g., AUTOSAR naming conventions) by means of a configurable name checker/builder.

What kind of data does the ADD manage?

The ADD manages both data objects, i.e., measurement values and calibration variables as well as their attributes. This includes physical units and conversions as well as ASAM MD-specific properties. These data objects are typically used by modules or functions as I/O values or local variables, respectively. The inputs and outputs define the interface of the function. Because the ADD is able to deploy the project view to check these interfaces, it not only guarantees the compatibility of all function interfaces within a given project environment but also ensures their consistency across version boundaries.

In addition, the ADD facilitates the definition of system constants for the purpose of effecting module and/or project configurations, e.g., by means of defining code variants at the module level. In the event that the system constants are provided with the governing project values at the project level, the ADD will instantly show the effect of the resulting configuration on the code variants, to name but one example. It is therefore immediately apparent how the configuration affects the project's modules and their interfaces. These configurations, too, are centrally versioned, which means that a given configuration can be called up with the push of a button – even if the attendant project is no longer active. This comprises an essential contribution to ensuring reproducible versions, whether releases or intermediates.



The ADD and its interfaces.

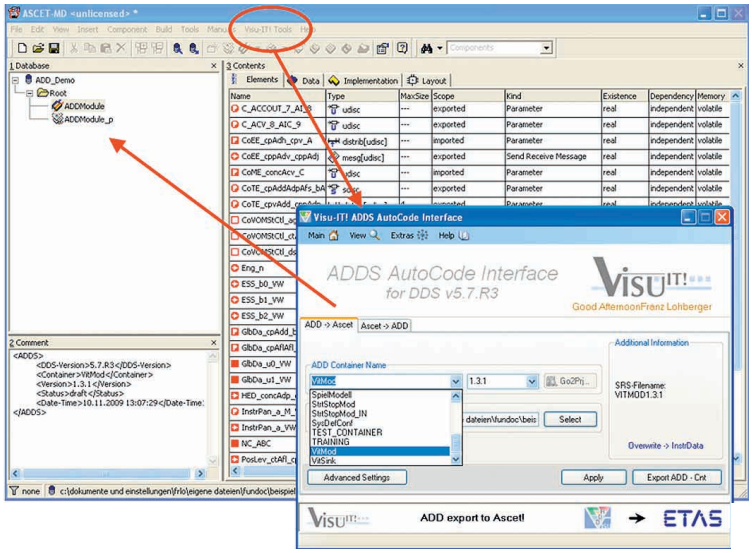
The connection

The interconnection between ASCET and ADD functions equally well via the ASCET COM API and through the preferred - and recommended – XML interface. It is specifically this new interface that helps eliminate the existing performance issues experienced with the COM interface. The XML interface has proven far more powerful and flexible.

An ADD import dialog can be started by means of ASCET's Executor mechanism. It lists all functions available in the ADD database, along with their individual version labels. Once the desired function has been selected in the required version, the declarations can be imported into the ASCET database.

Part of the import function also includes the creation of required projects, modules or classes, if these are not yet present. The associated attributes and/or implementation are taken from the ADD database. These attributes become necessary, e.g., for preparing ASCET-based models for automatic code generation.

Integration of the ADD in the ASCET database.



Reimport and data reconciliation

The XML interface provided by ASCET shows yet another benefit: in the same manner in which data is exported from the ADD to ASCET, it can also be moved in the opposite direction, i.e., from ASCET to a data dictionary.

This type of reimport is an essential function in some situations:

Data migration

Provided the customer already possesses existing ASCET models, the variables contained therein can be easily imported into the ADD database.

Model development

When developing a new or extended model, it may be useful to start by defining the variables in ASCET, and to synchronize these in the ADD database at a later time. Here, too, the ADD provides support: Using a dedicated ADD Offline Editor, data from the central database can be modified offline, and the resulting data can then be imported (and synchronized) in the ADD.

Another ADD feature makes it possible to synchronize an ADD function with an ASCET function (modules).

Summary

ASCET's native XML interface provides an excellent means of interconnecting ASCET with global data dictionaries. This empowers ASCET with the automatic use of globally managed information, which in turn brings about a tangible reduction of manual effort. It is an efficient way to ensure data consistency both company-wide and across process and project boundaries. In a nutshell, the ADD represents an essential building block for quality assurance.

For more information, please go to www.visu-it.de

THE CHALLENGE

To make externally managed data declarations consistently available for use by both simulation and code generation in ASCET.

THE SOLUTION

ASCET's native XML interface easily connects to external data sources, such as global data dictionaries.

THE BENEFIT

Information globally managed in ASCET can be automatically used, resulting in a tangible reduction of manual effort. It is an efficient way to ensure data consistency both company-wide and across process and project boundaries. In a nutshell, the ADD represents an essential building block for quality assurance.