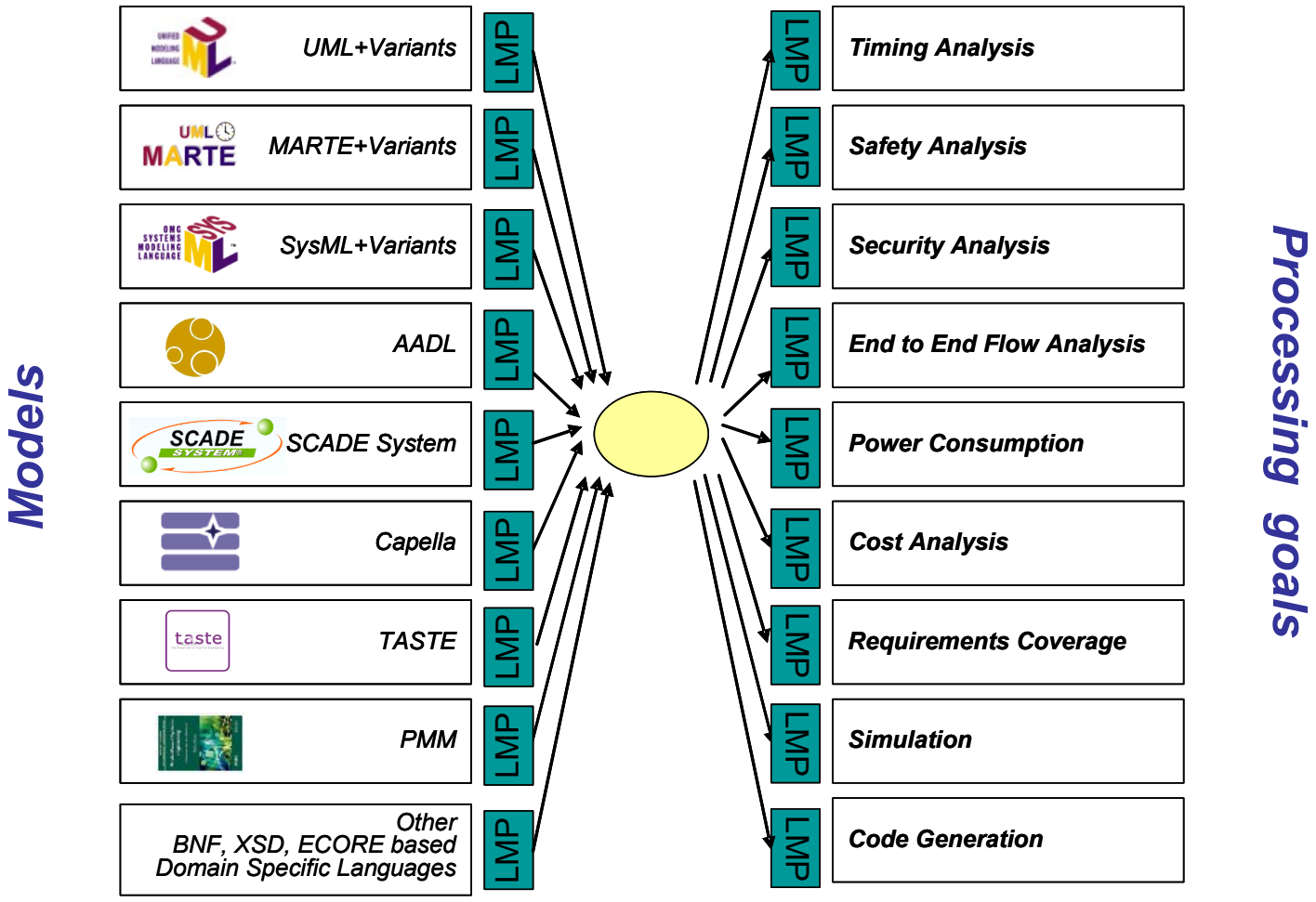
Ellidiss
www.ellidiss.com

LMP Dev Kit

Development Toolbox
for Logic Model Processing



Model Queries
Model Constraints
Model Transformations (M2M, M2T, T2M, T2T)
Model Exploration

LMP

Model Driven Engineering

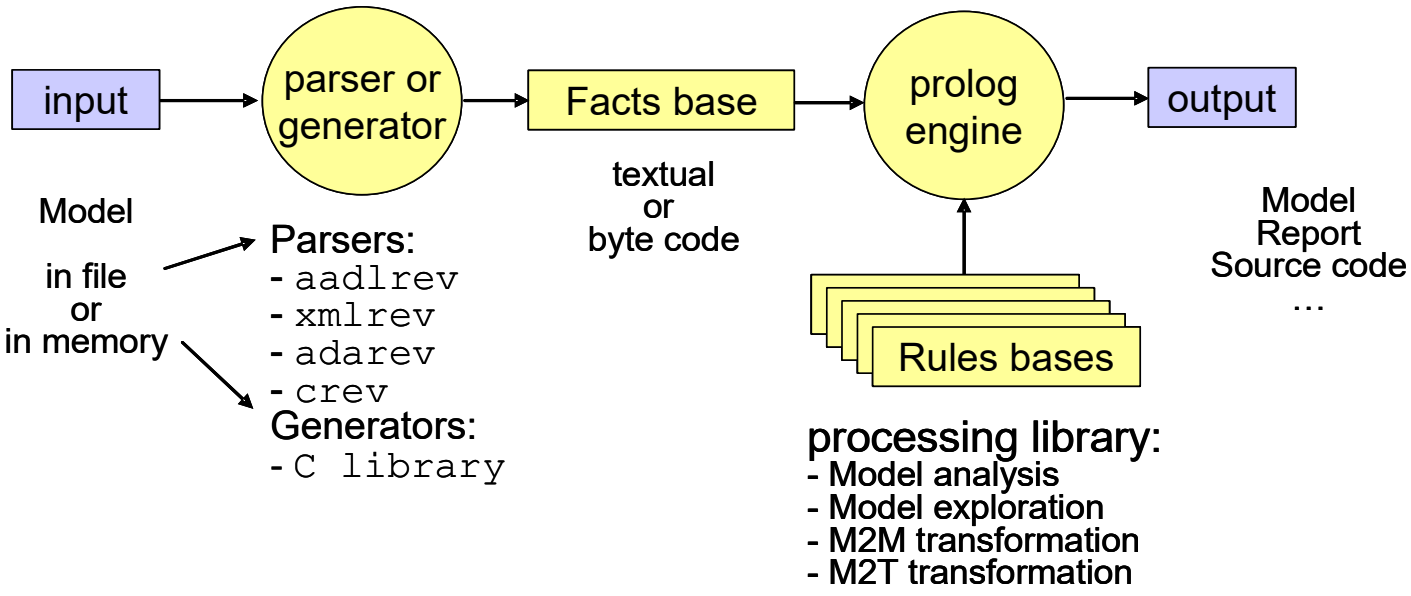
Model Driven Engineering is now recognized as a way to significantly improve the development process of industrial systems and software. This approach leads to the production of various kinds of models associated to each modelling and verification step of the life cycle. All these concrete models may differ in their abstract definition (meta-model) and in their syntactic expression. Such diversity cannot be easily avoided as each modelling language brings its own specific benefit or is fundamentally associated with a particular tool or technique. However, merging and processing heterogeneous models to support all the required development activities can become a real engineering issue in the context of industrial projects.

Logic Model Processing

The Logic Model Processing (LMP) methodology provides a unique, standard and easy to process representation of each model that is involved in a given project. Using this solution leads to the realization of a global homogeneous repository from syntactical conversion of each input model, without altering their semantic diversity. It then dramatically facilitates the development of model processing tools, such as model explorations, model verifications, model transformations and architectural reasoning.

LMP Methodology

LMP is based on the use of the Prolog language. Prolog is a declarative programming language that is used to express rules applying on predicates. The syntax of the Prolog language is very simple and most programs can be specified using only AND, OR and NOT logical operators.The development of a LMP application requires three main steps:
- Development of a Prolog Facts base generator for the input model or source code.
- Development of the Prolog Rules base to implement the required processing function(s).
- Merge of the Facts and Rules bases and use of a Prolog interpreter to produce the output model, analysis report or source code.



LMP Dev Kit

Tools to support the LMP methodology

The LMP Dev Kit provides the LMP designer with a set of tools and libraries to increase the productivity and the quality of the development of LMP applications. This toolbox is composed of three compartments:
- Tools for the generation of Facts bases
- Tools for the development of Rules bases
- Tools for the run-time execution of the LMP application.

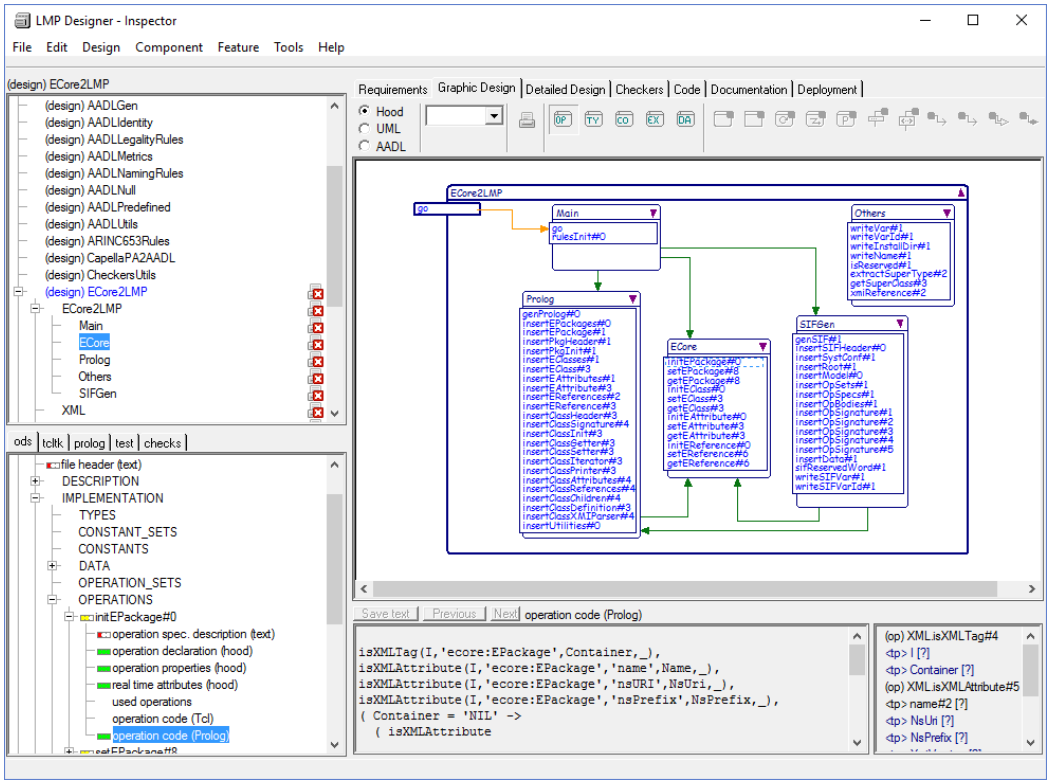
The LMP Dev Kit currently uses the sbprolog runtime environment

Facts bases generation

Input models have to be syntactically transformed into a Prolog Facts base. This can be done either by extending the source modelling tool with a LMP predicates generator, or by parsing files for models that have been serialized in standard languages. The LMP Dev Kit contains the following tools :
- a C library to generate predicates in sbprolog byte code format (.sbp)
- a XML parser
- an AADL parser
- an Ada parser

Rules bases development

The LMP Designer graphical editor can be used to facilitate the development of new rules bases. Prolog source files are automatically generated from the graphical designs and translated into sbprolog byte code for better efficiency. A set of existing prolog libraries can be easily included into a design and advanced facilities like the automatic generation of a new Prolog design from an Ecore meta-model description are also provided :
- Ecore2LMP generator
- AADLModel processing library



Summary

Fully featured: a single solution for:
- Model queries
- Model constraints
- Model transformations: M to M, M to T, T to M or T to T
- Model exploration and architectural reasoning

Standard prolog language (ISO/IEC 13211-1)

Open: compatible with the main meta-modelling formats (BNF, XSD, Ecore)

Declarative: incremental development

Modular:
- separate fact and rules bases
- facts bases merge
- rules bases associativity

Formal (boolean logic): appropriate for tool qualification

Flexible:
- Supports heterogeneous models
- Supports incomplete models (subsets)

Industrial return of experience:
- Airbus: involved in the development of DO-178 certified projects (A380, A350)
- European Space Agency: used in the TASTE graphical editors
- Ellidiss: AADL Inspector model adaptors and Stood code generators

Support to developers:
- LMP Dev Kit
- Training
- Consultancy

Ellidiss

www.ellidiss.com

UK office:
Mountbatten Court
Worrall Street
Congleton
CW12 1DT
UK
+44 (0) 1260 291449

support site :
www.ellidiss.fr

France office:
24 quai de la douane
29 200 Brest
France

lmp@ellidiss.fr
+33 (0) 298 451 870