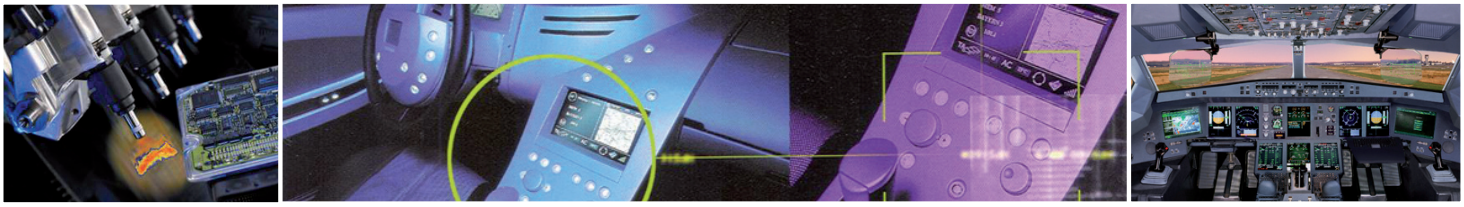
GENE-AUTO
(ITEA ~ 05018)

Tonu Naks, IB Krates, Estonia,
Xavier Olive, Thales Alenia Space, France and
Olivier Ssi Yan Kai, Continental, France

Automatic code generator speeds development of safety-critical real-time embedded systems

The ITEA Gene-Auto project has developed an open-source code generator that enables automatic transformation from high-level industry-standard Simulink, Stateflow and Scicos models to executable program code. The system enables automatic code generation for real-time embedded systems in safety critical domains and was evaluated successfully in nine industrial case studies. The Gene Auto toolset conforms to aviation industry standards.



Airplane cockpit picture © Airbus S.A.S.

Embedded critical systems now represent a large part of the final product cost in many European industries. This is particularly so in the safety-critical aeronautic, aerospace and automotive domains – some 25% in automotive and 35% in aeronautics. Moreover, such systems are responsible for much of the innovation in these sectors.

However, development of such complex systems has had to face up to two contradictory challenges: a demand for higher levels of integrations with lower costs, while at the same time the intrinsic complexity of these systems is exploding.

Some industries are compensating for this by outsourcing a large part of their software development to low-cost countries as such activities require a significant workforce to carry out all activities in compliance with current development standards. However, such outsourcing can create extra loops and costs and is critical in terms of know-how and competences.

GROWING USE OF MODEL-DRIVEN SIMULATION

An approach making it possible to reduce the workforce and shorten the development loop, used more and more widely for embedded systems, is model-driven development. Model-driven engineering is widely applied in the development of highly complex and critical systems as application-specific solutions can be more easily described using high-level graphical modelling rather than a computer programming language.

Abstraction of the implementation details makes it possible to verify and validate important system properties very early in the design. Subsequent model refinements add all the necessary details and, finally, the code generator helps to convert the model to working software code.

Increased productivity in the specification and design phases coupled with early verification and validation are already a major step forward when compared with classical development methods. However, there is more to gain! A certified code generator which guarantees lossless conversion from model to software code will allow the suppression of the large amount of verification required normally at code level. This was the goal of the Gene-Auto project – to develop a certifiable automatic code generator (ACG) allowing the conversion of high-level functional models to 'correct-by-construction' software code.

DEVELOPING AN OPEN MARKET-STANDARD APPROACH

Gene-Auto has implemented an open-source solution for automatic code generation based on the *de-facto* industrial standard Simulink/Stateflow modelling suite from The Mathworks and its open-source counterpart Scicos from INRIA. These are both graphical modelling and simulation tools that enable the user to create block diagrams to model and simulate the dynamics of hybrid systems and compile models into executable code.

Several commercial tools exist to convert Simulink/Stateflow models to embedded code. However, they are all proprietary with narrow freedom for the end users. When compared with the existing solution, Gene-Auto allows more freedom for organising the development process and for toolset customisation;

this enables the toolset to be certified and through that suppress code-level verification activities.

The Gene-Auto team gathered the best from existing European knowledge: aeronautics certification experience, aerospace and automotive know-how of efficient embedded code, European-wide software skills and applied research on formal methods from European research institutions.

The result is an open-source toolset that:

- Facilitates long-term support, maintenance and tool qualification;
- Is based on publicly available and well-defined meta-models for system and code modelling to ensure a strong theoretical basis;
- Supports multiple input formalisms and fully automatic transformation from input model to embedded code;
- Provides support for qualification through open architecture and suitable development processes within the scope of the DO178B standard;
- Contains components developed with formal methods; and
- Produces optimised code.

The toolset has been tested and validated by project partners including Airbus, Barco Avionics, EADS-Astrium, Thales Alenia Space and Continental that are the leaders of their specific industries.

IMPACTING COSTS AND PERFORMANCE

The initial impact for users of the Gene-Auto toolset is on the cost of ownership: no purchase is required and maintenance costs will be reduced by 20 to 30%.
A second and even more important effect will be the drastic increase of performance and efficiency of embedded software development. For example, Airbus has carried out a benchmarking exercise comparing two similar software developments – one coded using traditional methods, the other based on automatic code generation for 75% of the software code.

Resulting figures for these Airbus developments show an overall reduction in cost by a factor of three and a reduction in the development cycle by a factor of four. The second point is particularly important as it allows a faster lifecycle for 'on-time in-market' products, especially critical within all domains.

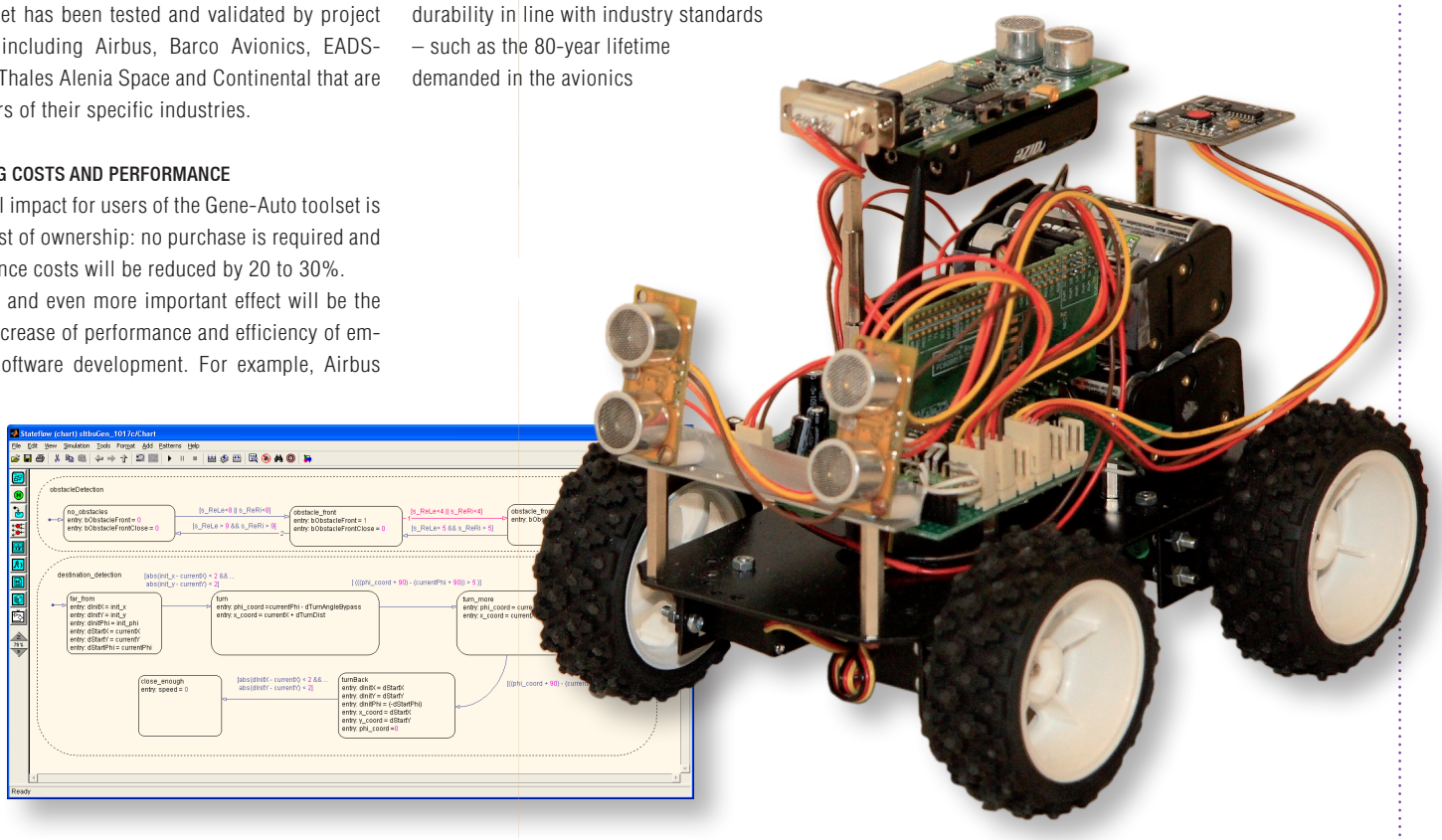
Moreover, the good co-operation between the industrial partners from safety-critical domains, SMEs and academia in the Gene-Auto project ensured that common requirements were worked out from several key European industries. It has also resulted in the creation of an embedded systems community based on aeronautics excellence and maintaining Europe's leading position in these industries.

Furthermore, the open-source strategy deployed by the project has also shown a satisfactory scheme for ensuring the openness of the tools, the end-user influence on the development roadmap and the required durability in line with industry standards – such as the 80-year lifetime demanded in the avionics industry – with the possibility of different uses and contributions.

Last and not directly foreseen from the project set-up, due to the strong theoretical basis of the project in terms of meta-modelling, academic partners have demonstrated the use of Gene-Auto components for bridges between Simulink and its open-source counterpart Scicos.

Following the end of the Gene-Auto project in 2008, a first public release was made under the GPL licence in early 2009. Work is continuing in an open community to extend the tool. The next major development in the tool functionality is adding conversion to the Ada language – currently the only available output language is C. Some project partners have already started this development in co-operation with AdaCore; the results are expected before the end of 2009.

More information
Gene-Auto community: *www.geneauto.org*
Gene-Auto commercial support:
http://geneauto.krates.ee/



This funny-looking vehicle is the public demonstrator of the GENE-Auto ITEA project. It is a small robot car with a smart autonomous navigation system and fuzzy-logic motion control. The essential part of the control an positioning algorithms were modelled in Simulink/Stateflow and code generated with GENE-Auto.