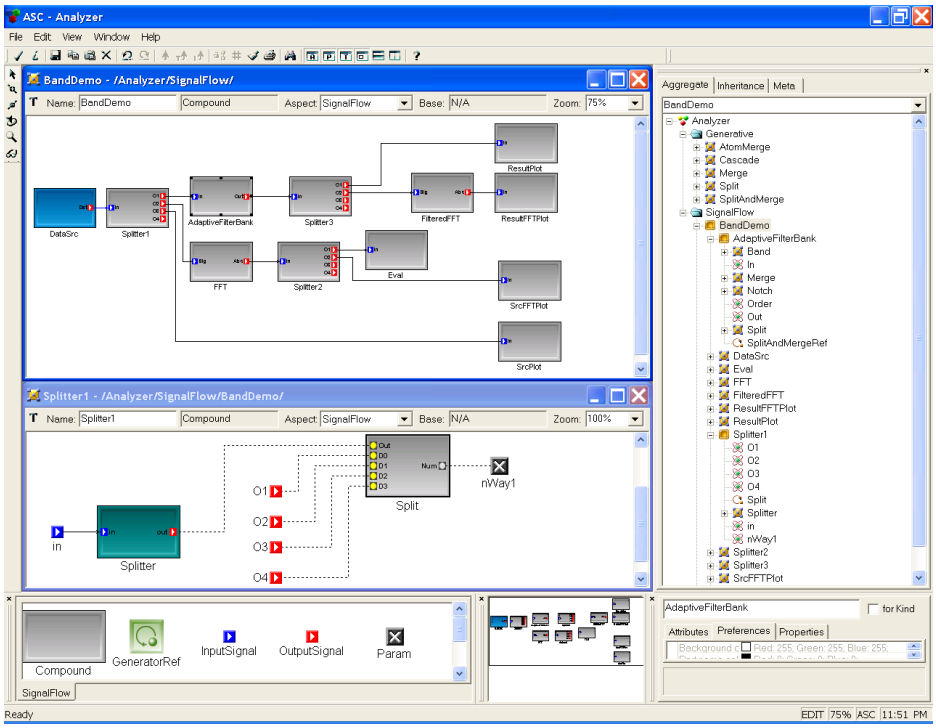
GME: Generic Modeling Environment
Summary of Features

VANDERBILT UNIVERSITY
INSTITUTE FOR SOFTWARE INTEGRATED SYSTEMS

The Generic Modeling Environment is a configurable toolkit for creating domain-specific modeling and program synthesis environments. The configuration is accomplished through metamodels specifying the modeling paradigm (modeling language) of the application domain. The metamodeling language is based on the UML class diagram notation and OCL constraints. The metamodels specifying the modeling paradigm are used to automatically generate the target domain-specific environment. The generated domain-specific environment is then used to build domain models that are stored in a model database or in XML format. These models are used to automatically generate the applications or to synthesize input to different COTS analysis tools.

GME has a modular, extensible architecture that uses MS COM for integration. GME is easily extensible; external components can be written in any language that supports COM (C++, Visual Basic, C#, Python etc.). GME has many advanced features. A built-in constraint manager enforces all domain constraints during model building. GME supports multiple aspect modeling. It supports distributed multiuser access and model versioning. It provides metamodel composition for reusing and combining existing modeling languages and language concepts. It supports model libraries for reuse at the model level. All GME modeling languages provide type inheritance. Model visualization is customizable through decorator interfaces.



GME has been applied to several real-world applications. The Saturn Site Production Flow (SSPF) system monitors the manufacturing process at the Saturn Corporation providing key production measures to managers in real-time. The primary objectives of the State Space Analysis Tool (SSAT) developed for Sandia National Laboratories involve increasing the reliability, safety, and security of high-impact systems by performing analysis of design models. MIC has also been used by Boeing to develop tools for Fault Detection, Isolation and Recovery (FDIR) applications for the International Space Station. ESML, the Embedded System Modeling Language and Toolchain, is used to construct applications running on a Real-Time CORBA platform, such as the Bold Stroke Framework. GRATIS is a graphical development environment for wireless sensor network applications running TinyOS.

Contact point: Akos Ledeczi, akos.ledeczi@vanderbilt.edu, http://www.isis.vanderbilt.edu/Projects/gme/
Tel: (615) 343-8307, Mail: ISIS, Vanderbilt University, PO Box 1829B, Nashville, TN 37235.

Last updated: 5/5/2008