

# Brief overview of SDT

*SDT comprises the following graphical tools:*

- *The SDL Editor is used for creating, editing and printing specifications and descriptions using the graphical SDL notation defined in the standard Z.100. The SDL Editor also performs various syntax checks at editing time.*
- *The Type Viewer visualizes the impact of the inheritance and specialization mechanisms in your SDL-92 system. The Type Viewer produces a graphical tree that is of great assistance to understand and take full advantage of the SDL types that you have defined in an SDL system.*
- *The Index Viewer presents listings of definitions and cross-references in a clear and easy-to-understand graphical notation. The Index Viewer is provided with filtering and navigation functions, with a trace-back to the source SDL or MSC diagrams.*
- *The Coverage Viewer is a test coverage and profiling tool that displays the results of a simulation or validation as a graphical transition or symbol tree. The tool can present an overview of the system, coverage or a detailed view on a part of the system.*

*The following additional tools and facilities are available:*

- *The ADT Library (library of Abstract Data Types) features a number of general ADTs that provide the basic services that are often needed when you design an SDL system. The ADT library is distributed in source code so you can tailor the ADTs to fit your specific requirements, if needed.*
- *The SDT Analyzer performs several functions. It performs syntactic and semantic analysis of your SDL descriptions, generates error reports and warnings in appropriate cases, and has the ability to produce information about definitions and cross references in an SDL system. The Analyzer also converts SDL information from the Graphical Representation (SDL-GR) to the textual Phrase Representation (SDL-PR). The reverse conversion is also possible, allowing you for instance to import PR files from other tools than SDT.*
- *The Cadvanced/Cbasic Code Generator transforms your SDL system into a number of C source files that are compiled and linked with an SDT runtime library. The C code can be used for a number of purposes, depending on what libraries are available in your configuration (see below). The C Code Generator is available as a Cbasic code generator (for simulation and validation purposes), and a Cadvanced code generator (for building any kind of application).*
- *The SDT Simulator library, you can make an executable program, a simulator, which helps you to understand and debug the behavior of a system specification. The simulator can be controlled from a graphical user interface (SimUI).*
- *With The SDT Validator library, you can make a validator, an advanced "self-exploring" simulator that may be used for finding errors and inconsistencies in an SDL system and for verifying that a system is consistent with a Message Sequence Chart. The validator can be controlled from a graphical user interface (ValUI).*
- *The Performance Library allows you to create a performance model of your SDL system that you run on your host computer. The library is optimized with respect to performance, so that a large amount of statistical data can be produced during a reasonable execution time.*
- *The Cadvanced code generator can be used for Building an Application for both host and target environments. Precompiled Application libraries are available for specific host environments. The Master Library is the SDT run-time library in source code format, which can be customized to fit different needs and operating systems.*

*You can choose to focus on the external view of a system specification, where you are interested in the signal interface, or on the internal behavior of a system specification. The execution of a simulator can be traced in a graphical mode in the source SDL diagrams and can be logged graphically in terms of Message Sequence Charts. Target simulation is also supported.*

*Integration with Operating Systems supports most of the commercially available real-time operating systems. You can also build applications where the runtime library schedules the system and sets the real-time pace.*

***You can browse the html help files for SDT from within the Help Menu Option if you are using the Lintula machines. Essentially they are the same as the ones you would find from within SDT itself.***