

## TOOLS FOR SUPPORTING THE USE OF SDL

Edited by E.M. Dijkerman  
PTT - Dr Neher Laboratories  
P.O. Box 421  
2260 AK Leidschendam  
The Netherlands

During the third SDL forum a lot of tools to support the use of SDL were demonstrated. Tools are at least as important as research, since people will not use specification languages, unless they are assisted by editors, analysers, simulators, etc. What we intend to do in this paper is to present an overview of the demonstrated tools. For each tool we try to answer the following questions:

- Which are the functions the tool can perform?
- Is the tool commercially available?
- Is the date of supporting SDL'88 already known?
- What hardware and software environment are required for the tool?

### SDT (SDL DESIGN TOOL)

The present version of SDT [1] contains a graphical editor, a report generator, an analyser, a syntax converter and database utilities. The graphical editor handles all SDL information, block diagrams, process diagrams and associated text. The report generator derives and prints the SDL documents for a process, block or a complete system. The analyser makes an analysis of the SDL information and the syntax converter can convert between the different syntax representations. With the utility functions it is possible to transfer data between different computer environments.

SDT is developed and marketed by Telelogic AB, Sweden.  
Work on SDL'88 is ongoing.

At the forum SDT was demonstrated on a SUN/3 workstation and a TEKTRONIX 4111 connected to a VAX/VMS computer. In the near future SDT will be available on Apollo and VAX station. Work is also going on port the graphical editor to PC/AT.

### DASOM

DASOM [2,3] is a toolset supporting the SDL'84 language. DASOM contains a graphical editor covering the different SDL diagrams, and also does syntax analysis and to some extent consistency checking. It especially supports the specification and design phases, but it also has automatic program generation (to C, other languages are being considered), document management, archive facilities, version management and other features supporting the whole of the system lifecycle.

DASOM is available as a commercial product from A.S COMPUTAS in Norway. COMPUTAS offers a full support and service (including updating to new SDL versions) and training courses. Updating to SDL'88 is planned.

DASOM runs on all types of VAX computers under the VMS operating system. Versions for PCs and workstations are considered.

A lot of low-cost, high resolution graphical terminals may be used, and hard-copies can be generated on a number of printers/plotters/laser printers. DASOM is connected to Telox SDL Tools, and the results from the program generator can be used directly as input to Telox SDL Tools.

#### TELOX TOOLS

Telox SDL Tools (TST) is a toolset for implementing Extended Final State Machines (EFSM) supporting SDL'88. TST contains tools for implementation, testing and debugging of EFSM using signal charts, transition logging, signal state matrices, state machine manipulation, etc. The tools can easily be connected to many existing SDL design tools.

TST is commercially available and marketed by TELOX A/S, Norway.

The tool runs on VAX computers, HP workstations and IBM PC.

#### THE DANISH SDL-TOOL

The Danish SDL-tool [4] consists of several components, described below. The SDL/GR editor currently supports process graphs only, but during this year full SDL/GR will be supported.

The SDL/PR syntax analyser supports full SDL'88 now. Also an SDL'84 syntax analyser is available. The SDL/PR pretty printer now supports SDL'84, but during this year it will be updated to SDL'88. GR to PR and PR to GR translators will be completed this year.

The semantics for SDL'88 written in META IV have been completed. This specification will be used for the implementation of a semantic analyser. This analyser will transform the SDL/PR into the recommended abstract syntax which can be used for simulation and code generation.

The Danish SDL-tool is developed by the Telecom Research Laboratory in Copenhagen (Denmark). It will be available commercially at the end of 1987 or at the beginning of 1988.

The Danish SDL-tool will support full SDL'88 around the middle of 1988.

The tool is presently hosted on VAX/VMS with a TEKTRONIX 4lxx terminal or IBM PC/AT with emulator. During 1987 it will be available without emulator.

## AN ABSTRACT EXECUTION PROTOTYPING SYSTEM

This prototyping system [5] is a tool to support partial verification of specifications written in SDL. It initially translates the specifications into a program written in C, then executes the program and presents the results of execution to the user. This system is characterized by the fact that it can accept incomplete specifications (in addition to complete specifications) and produce their maximum execution traces making full use of the information in the specifications.

This prototyping system is developed by KDD, Japan, and is intended to be used within KDD only.

This tool requires the following environment. It runs on a VAX-11 with operating system UNIX. A TEKTRONIX 4107 is needed for graphical displays.

## SDL-DEBUG

SDL-debug [6,7] is a portable test-environment for systems specified in SDL. It allows extensive testing of complete systems as well as subsystems. The subject of the demonstration was a small system, which was designed in SDL and implemented by means of an SDL tool environment of the same company. This environment allows the commands of SDL-debug to be executed. The power and convenience of SDL-debug was demonstrated while testing the example system.

SDL-debug was developed at Philips Kommunikations Industrie AG in West-Germany.

This tool is commercially available only within Philips.

Update to SDL'88 will be done, but there is no fixed time schedule yet.

The tool run on VAX/VMS and requires a VT 100 compatible terminal.

## PROTEAN (PROTOCOL Emulation and ANALysis)

PROTEAN is a tool for verifying protocol specifications. It uses high level Petri nets, so-called numerical Petri nets. The verification is done by interactive simulation and automated exhaustive analysis of the global state space. PROTEAN can generate all global states reachable from a given initial state. The reachability graph, showing the relationship of these states, is displayed graphically. It is possible to reduce the graph to show only the events of interest. PROTEAN can also search for a given scenario in the graph. Furthermore deadlocks, livelocks and cycles can be highlighted.

PROTEAN has been used by Telecom Australia to verify many protocols, including the OSI class 0 and 2 transport protocols, and level 3 of ISDN Basic Access. Faults were found and corrected in all these protocols.

PROTEAN is distributed by Telecom Australia (International).

PROTEAN runs on VAX computer under VMS (a UNIX version is planned). Colour graphics are displayed on VT 241 terminals.

#### SDL-E

There are several ways to use SDL-E. One way is to use the personal computer "stand alone", e.g. for study groups within CCITT or specification of services required by a manufacturer.

In larger projects the personal computer is used as a front-end processor communicating with a host processor offering more features for analysis, simulation, code-generation, etc.

Furthermore SDL-E is an excellent tool for education purposes.

It consists of the following components:

- a process graph editor for SDL'84
- a syntax checker
- an SDL/PR code generator.

Furthermore general access to the standard Macintosh environment (including word processing, graphics, etc.) is possible.

SDL-E is developed by Functional Systems in Sweden.

The tool is commercially available. The release date for SDL-E supporting full SDL'88 is October 1987.

SDL-E requires the following hardware:

- Macintosh Plus or Macintosh SE or Macintosh II
- Imagewriter or Laserwriter.

#### SAS (Software Automation System)

SAS [8] contains an SDL/PR analyser for syntax and semantics. Furthermore there are translators from SDL/PR to SDL/GR and the other way round, and also from SDL to CHILL [9]. Finally there is a graphical editor.

SAS is developed at TELEBRAS R & D Center (Brazil).

Two versions are supported: a VAX/VMS version, which will be available in July '87 and an IBM PC compatible version, available December '87.

Whether SAS will be updated to SDL'88 has not yet been decided.

The VAX/VMS version needs a printer plotter LXY11, VT 125 or VT 240 terminal and VAX PASCAL.

The IBM PC compatible version requires professional graphics and Microsoft PASCAL.

## THE AT&T SDL TOOLS

This demonstration consisted of a graphical editor, a translator from SDL/GR to SDL/PR, a translator from SDL/PR to SDL/GR, and a report generator. The SDL/GR editor supports multiple pages, has text search and replace features, allows a group of symbols to be moved or copied as a unit, automatically connects symbols, and preserves connections when symbols are moved. The SDL/PR to SDL/GR translator automatically divides a diagram into pages, sizes symbols and text to fit, and automatically places symbols. The report generator produces symbol cross-references, tables of contents, reachability matrices, and adjacency matrices.

These tools have been developed by AT&T. They are not available outside AT&T. Updating the tool to SDL'88 is being considered. However there are no specific plans yet.

The tools run under UNIX System V. The graphical editor requires an AT&T terminal.

## KINDRA: COMPUTER AIDED DESIGN OF SOFTWARE

KINDRA - British Telecom's graphical software design and coding tool [10] - is an established product which has been used extensively for the development of major real time systems, including System X and Monarch exchanges. The suite of software has recently been extended to provide an interactive graphical interface for the production of the SDL/GR based charts. A pre-release version of the KINDRA SDL Toolset was demonstrated, showing the production and automatic coding of a working simulation using the State Machine Translator. This is an aid to ensure that the documentation reflects the system in the field. Additionally, conversions from GR to PR and back again were demonstrated with analysis of the SDL/PR form by the KINDRA PR Static Analyser. Analysis at the algorithmic or control flow chart level was performed using the KINDRA Chart Analyser, reporting faults in terms of the chart constructs.

KINDRA has been developed at British Telecom Research Laboratories. It is commercially available.

KINDRA runs on the DEC VAX family of mini-computers under the VMS or ULTRIX operating system, and on several other mini- and micro-computers under UNIX. High resolution graphical terminals or workstations are required.

## SDE (SYSTEMS DESIGN ENVIRONMENT)

SDE is a combination of the SAL (Service Addition Language) and KINDRA tool sets to support efficient maintenance of telecommunications software. SDE allows full definition of systems by formal specification of message sequences using the language SAL. Services are described independently. I.e. new services to be added can be specified without needing to refer to the existing services.

Furthermore SAL allows hierarchical verification of specifications. I.e. correctness of a set of SAL specifications is verified by checking hierarchical properties.

SDE can transform a SAL specification into an SDL specification and into source code in several programming languages.

The demonstration consisted of the transformation of a simple service described in SAL into process descriptions in SDL and executable programs. Also the automatic combination of additional service descriptions and the generation of new executable programs was shown.

SDE is developed by NTT in Japan and British Telecom Research Laboratories (KINDRA support group) in U.K. SAL is to be commercially available in 1988, while KINDRA is available now.

SDE runs on VAX/VMS or UNIX.

#### SDL-ORIENTED GRAPHICAL ENVIRONMENT

The SDL-oriented Graphical Environment [11] consists of several components. The graphical editor supports full SDL'88. It is template-driven and has a user-friendly interface. It can draw large diagrams and performs automatic scaling of symbols during text insertions.

The printing of diagrams can be done both by a screendump and/or phototype setting using pic/troff. Also automatic shrinking is performed when large diagrams are printed.

Furthermore there is a translator from SDL/GR to SDL/PR, and also a translator from SDL/GR to CHILL.

The SDL-oriented Graphical Environment has been developed at the Electronics and Telecommunications Research Institute in Korea. It is commercially available.

The required hardware consists of a SUN/2 or SUN/3 workstation and laser printers supported by troff/ditroff.

#### MELBA

The MELBA [12] system provides automatic generation of compilable CHILL code from SDL specifications. The SDL version used is 1984 SDL/GR.

The MELBA methodology distinguishes the system designer, working at the block interaction diagram level, the process designer, working at the process diagram level, and the abstractions designer, working with the abstract data types (ADTs).

MELBA generates SDL/PR as an intermediate stage in the generation of CHILL. Furthermore the ADTs are stored in a library and can be used both by the system and the process designer.

The MELBA system is well suited to rapid prototyping.

MELBA has been developed by Telecom Australia Research Laboratories and the Royal Melbourne Institute of Technology. It is commercially available.

Two versions of MELBA exist: a version for an ICL PERQ workstation and one for an IBM PC/AT (called P.C. MELBA).

## AN SDL-BASED OPERATING SYSTEM

The demonstration of the Delft University of Technology showed an operating system that has been developed for the implementation of SDL specifications [13]. It provides a high level run-time environment with primitives that are close to the SDL model. A bottom-up approach is assumed for the specified software; this makes it possible to modify and reconfigure the specification at runtime. The system supports the hierarchical structuring of SDL: a user can interact with the running specification on various levels of abstraction. These features are demonstrated with an implementation of CCITT signalling system no.7, level 2.

The operating system has been developed at Delft University of Technology. At this moment it is not a commercial product.

The operating system runs on a 68000 microcomputer.

## YAST (Yet Another SDL Tool)

YAST [14] will contain several components.

YAST-DESKTOP enables easy navigation among various components. The components are presented by icons which can be selected by a mouse. (This component is not yet implemented).

YAST-PRED will be a syntax-driven text editor (not yet implemented).

The major part of a graphical editor, YAST-GRED, has been developed. It will support the creation of process, procedure and macro diagrams as well as block tree and block interaction diagrams.

A translator from PR to GR has been implemented and a translator from GR to PR will be developed soon.

Finally YAST will contain an analyser to check the syntactic and semantic correctness of SDL documents.

YAST is developed at the Faculty of Electrical Engineering Zagreb in Yugoslavia. It is not yet clear whether it will be available commercially.

The first prototype runs on a personal computer ATARI ST, using GEM user interface functions and a standard C library.

It has been designed to be fully portable.

## SPECS SDL-LOTOS INTEGRATION

The RACE-project SPECS [15,16] has proposed an architecture for a specification environment, that provides integration of several specification languages through a common semantics, expressed by an internal language, called MR (Model's Representation).

A critical question concerned the feasibility of this proposal. In order to answer this question SPECS constructed a Feasibility Model of the architecture. The Feasibility Model had to deal with specifications in both SDL and LOTOS. Translators from a subset of SDL and a subset of LOTOS were made. A simulation tool was built and demonstrated at the forum on a small protocol example.

The SPECS consortium consisted of the following companies:  
GSI-TECSI, CNET, DNL, IBM France S.A., STET-CSELT and Telettra S.A.  
The demonstration at the SDL forum is only a prototype and not commercially available.

#### GRAPHICAL EDITOR FOR SDL IN "SPRITE"

At the forum the graphical editor [17] which is a part of the graphical environment of the SPRITE system [18] was demonstrated. SPRITE is an Integrated Environment for SDL and CHILL. The editor will assist the user in designing correct specifications using the rules and algorithms discussed in one of the papers of the forum [19].

The editor is developed by TELEKOM ZWUT and the Warsaw University of Technology in Warsaw (Poland).  
It will be released in November 1987.

The hardware requirements are IBM PC/XT or AT, 640 kilobytes and a Hercules card. Furthermore MS DOS is needed.

#### PSDL PACKAGE

The PSDL package supports transition from a system specification in SDLSIM to its performance analysis [20]. SDLSIM is an extended version of SDL which incorporates performance analysis capabilities. The analysis is done by using discrete event simulation techniques.

The tool has been developed by TELEKOM ZWUT in Poland. It will be released in July 1987.

The required environment is the same as that of the previous tool.

#### CONCLUSION

The demonstrations were very interesting. It was difficult to get a complete impression of every tool within the time available, because of the large number of tool demonstrated. There were twice as many tools as were shown at the second SDL forum. This increase in the number of tools illustrates the wider acceptance and use of SDL.

In addition tools with new functions such as debugging, simulation, performance analysis and report generation are now available. Also a prototype of a future specification environment, capable of supporting more than one specification language, was shown.

Finally it is expected that many tools are going to support SDL'88. Tools that adhere to SDL'88 will most easily be accepted by the user community.

## REFERENCES

- [1] Belina, F., Nilsson, G., SDT: SDL Design Tool, this volume
- [2] Vefsnmo, E., DASOM - A SDL-tool, this volume
- [3] Vefsnmo, E. Johansen, U., Automatic Program Generation of SDL-specifications: Principles and Solutions, this volume
- [4] Jensen, A., The Danish SDL-tool, this volume
- [5] Wakahara, Y., Ito, A., Prototyping System for Telecommunications Software Based on Abstract Execution of Requirements Specifications, this volume
- [6] Krechel, C., Successful use of a highly tool-supported SDL-environment, this volume
- [7] Jobes, B., Irlner, W., SDL-DEBUG a Portable Test Environment for SDL-systems, this volume
- [8] Batista Junior, J., Software Automation System, this volume
- [9] Franco, J.H.A., Haim, J., Lima, H.M., Going from SDL to CHILL: "The Tropico Approach", this volume
- [10] Norris, M.T., Orr, R.A., Tinker, R., Supporting the SDL user, this volume
- [11] Hong, J.P., Lee, J.S., Shin, Y.S., Choi, W., SDL-oriented Graphical Environment, this volume
- [12] Jackson, L.N., Cheng, K.E., Choong, T.S., Pascoe, R.S.V., Cain, G.J., MELBA at the Age of Eight: An Automatic Code Generation System, this volume
- [13] Venemans, P.H.A., An Experimental Operating System for the Implementation of Telecommunications Software, Based on SDL, this volume
- [14] Zorić, M., An Approach to SDL Tool Development, this volume
- [15] The SPECS Consortium, Evaluation and Comparison of Three Specification Languages: SDL, LOTOS and ESTELLE, this volume
- [16] The SPECS Consortium, The SPECS Architecture: Towards an Integrated Specification Environment, this volume
- [17] Kimbler, K. Skorupinski, A., Wojtal, M., Gumulinski, P., Kakol, B., Dabrowski, M., Sredniawa, M., Development of an SDL Graphic Environment, this volume
- [18] Kimbler, K., Lubacz, J., Sredniawa, M., A Project of an Integrated Environment for SDL and CHILL, this volume
- [19] Pstrowski, P., Dziong, Z., New Features in SDL - Correctness by Constructions, this volume
- [20] Sredniawa, M., Kakol, B., Gumulinski, P., SDL in Performance Evaluation, this volume