UDC 681.3

COMPUTER SCIENCE AND INFORMATICS

T.H. SHAHINYAN

APPLYING SCRIPTING LANGUAGES AT DEVELOPING SYSTEMS OF AUTOMATED DESIGN OF INTEGRATED CIRCUITS

Scripting languages (SLs) are used in electronic design automation (EDA) software development and VLSI design flows. There are also EDA tools that contain embedded SLs. This article reveals various tasks and specialities that use SLs. It gives information on the purpose and strengths of SLs. The possible swapability and complementarity of SLs for various tasks in EDA software development and VLSI design flows are described too.

Keywords: scripting languages, electronic design automation, designing integrated circuits.

Introduction. SLs [1] are programming languages that do not require a preliminary compilation of the code before execution. The compilation of the code is implemented runtime by the interpreter. This feature saves time and provides the code portability between different operating systems. The SLs are usually higher level languages compared with the C programming language. Using a higher level language provides quick development of the required functionality.

There are many SLs. Different SLs have been designed and developed for different purposes, but there are 3 main categories of SLs [2]:

1. Command languages.
2. Markup languages.
3. General purpose languages.

This work will focus on the open source SLs used by EDA leader companies, such as Synopsys and Mentor Graphics. The EDA software tools of the market leaders are mainly developed in the environment of Linux operating system (OS). The Linux OS mainly determines the used SLs. The Linux package contains a number of open source SLs, but 3 of them are the most popular in the leading EDA companies. These 3 languages are Perl, TCL and Python. These are general purpose SLs used for various tasks. The general analysis of SLs can be found in other articles [2,3].

Specialities and areas of usage of SLs in EDA software development and VLSI design flows. There are several specialities that use SLs in EDA software development and VLSI design flows widely. These specialities are:

1. Software Configuration Management (SCM) Engineer.
2. Software Quality Assurance (SQA) Engineer.
3. Computer Aided Design (CAD) Engineer.
4. Corporate Application Engineer (CAE).

The engineers of the above mentioned specialities have tasks that are mainly solved with SLs. One of the areas of usage of SLs is a software development flow, which contains the following stages that use scripting widely:

1. Building software.
2. Running regression tests and writing run results to a MySQL database.
3. Viewing results using web browsers.

Building software is a non-trivial, multioptional task [4] that requires a SCM engineer to configure and watch after the process. To organize automated and regular builds in the environment of Linux OS, it is required to have a build system that setups the environment, checks out the appropriate branch from repository, compiles and builds software components with required options, validates and reports the results. All these steps require scripting.

Running regression tests is used by QA and SCM engineers for testing the software. This is done with test drivers that distribute thousands of tests among available computers and use multiprocessor parallel runs in order to finish testing in an acceptable time interval. For the minimization of the total runtime, there are several rules, which are discussed in [5]. All these tasks are solved by using scripting.

Steps 1-2 are mostly implemented either with Perl or by Python. Step 3 is mostly implemented with PHP, but it can also be implemented with Perl or Python. Perl and Python are perfect for distribution of multiple runs against servers on the grid [6].

VLSI design flow contains many stages and uses many formats (views) of representation of integrated circuits (IC). Many of these formats are ASCII files such as SPICE, verilog, LEF, DEF, liberty etc. For processing the text data, the best languages are Perl and Python. There are also binary formats like GDSII, OASIS, Milkyway, Open Access etc. which can be processed by Perl and Python. VLSI design contains simulation stages which can also have multiple corners and require parallel distribution on the grid [6].

During VLSI design SLs are used for automated analysis, verification, comparison and modification of the design views and other files. Typical tasks include but are not limited to:

1. Reading and reporting the design data.
2. Verifying the data against the technical specification.
3. Comparing the common data of different views.
4. Modifying the data to meet the requirements.

The VLSI design tools usually have embedded SLs, which allow to automate the design flow and integrate different EDA tools for interoperability (Fig.1) [7]. Actually, many languages can be used as an embedded language if designed accordingly but the most popular open source SL embedded in EDA tools is the TCL because it has been specilally designed for this purpose.

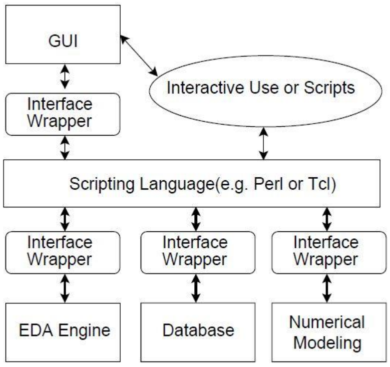

Fig. Tools integration by a scripting language

Features, swapability and complementarity of SLs. TCL (Tool Command Language) was specially designed for IC design applications. The author of the TCL is John Ousterhout who created the Magic VLSI layout tool [8], in the 1980's at University of California Berkeley and designed TCL in 1987 for embedding in such tools. Currently TCL is used in many tools of EDA market leaders. TCL also has a widget library for creating window based applications called TK [9]. TK is an open source, cross-platform widget toolkit that provides a library of basic elements for building a graphical user interface (GUI). TK can be used to create a GUI for reading input arguments of the existing scripts developed with other languages like Perl or Python. This provides an integration of the user friendly GUI with the existing command line applications. If embedded in an EDA tool, TK can be used for customization of GUI. Typical tasks include creating special menus with appropriate commands for a special purpose. These types of tasks are implemented either by CAD or CAE engineers.

Perl (Practical Extraction and Report Language) was originally designed in 1987 for text processing and it is very strong in it. Besides Perl has a wide range of applications. It is a general purpose language including features from C, bash, awk and sed languages. It has more than 25 thousand modules for various purposes at the famous comprehensive Perl archive network [10]. It is also widely used in EDA software development and VLSI design flows.

Python implementation was started in December 1989 by Guido van Rossum. Python is a general purpose language influenced by ABC, ALGOL, C, C++, Dylan, Haskell, Icon, Java, Lisp, Modula-3, Perl and includes but is not limited to the following features:

1. Very clear, readable syntax.
2. Strong introspection capabilities.
3. Intuitive object orientation.
4. Natural expression of procedural code.
5. Full modularity, supporting hierarchical packages.

6. Exception-based error handling.
7. Very high level dynamic data types.
8. Extensive standard libraries for virtually every task.
9. Extensions and modules easily written in C, C++ (or Java for Jython, or .NET languages for IronPython).
10. Used within applications as a scripting interface.

More information on Python usage and ready modules can be found at official Python web page [11]. The Python is also widely used for programmable layout cells (PYCELLs) development [12] for layout editors.

Conclusion. The usage of SLs provides quick solutions for common tasks in the EDA software development and VLSI design flows. SLs provide effective solutions for interoperability of different EDA tools. Python and Perl provide equal capabilities for solving tasks such as automating software builds, creating driver scripts for running automated tests, processing all types of ASCII files, using system commands and working with databases. TCL is mostly used as embedded scripting language in EDA tools and is used for automation of design flow and integration of different tools. The TK is used for customization of GUIs of EDA tools, as well as creation user friendly GUIs for existing scripts which can be written in other languages.

REFERENCES

1. **John K. Ousterhout.** Scripting: Higher-Level Programming for the 21st Century // IEEE Computer Society Press Los Alamitos. – March, 1998.- Vol. 31, issue 3. - P. 23-30.
2. **Kanavin Alexander.** An overview of scripting languages / Lappeenranta University of Technology. – Finland, December, 2002. - 10 p.
3. **Lutz Prechelt.** Are Scripting Languages Any Good? A Validation of Perl, Python, Rexx, and Tcl against C, C++, and Java. - A chapter for Advances in Computers. - August 18, 2002. - Vol. 58. – 62 p.
4. **Nick Jenkins.** How to Make Software. –eBook. – 2005. - 54 p. http://www.nickjenkins.net/prose/makingSoftware.pdf
5. **Shahinyan T.H.** Method for parallel testing of electronic design automation applications // Изв. НАН РА и ГИУА. Сер. ТН. - 2013. -Т. 66, №3. – С. 223-228.
6. **Shankar U.** Oracle Grid Engine User Guide. Release 6.2 Update 7. E21976-02. 2012. 154 p.
7. **Pinhong Chen.** Fast Integration of EDA Tools and Scripting Languages. http://embedded.eecs.berkeley.edu/Alumni/pinhong/scriptEDA/scriptEDA_pinhong.pdf
8. Magic VLSI layout tool http://opencircuitdesign.com/magic/
9. http://www.tcl.tk/
10. http://www.cpan.org
11. http://www.python.org
12. http://www.synopsys.com/Tools/Implementation/CustomImplementation/Pages/pycell-ide.aspx

SEUA (POLYTECHNIC). The material is received 11.10.2013.