

# Creating an efficient and incremental IDE for TTCN-3

Kristóf Szabados

Ericsson Telecommunications Hungary  
H-1117 Budapest, Irinyi J. u. 4-20  
Hungary  
Kristof.Szabados@ericsson.com

In this article we present methods and algorithms for constructing an efficient IDE in the sense that the processing costs of re-analyzing source code after change is minimal. Moreover, we show that these methods and algorithms can be designed in a way that they support iterative realization, hence, they fit better to the current trends of iterative software development life-cycle. We also show how these algorithms can be built into an existing system and we show measurements on performance benefits. The proposed methods were validated in the telecommunication area for compiling Testing and Test Control Notation - 3 (TTCN-3 [1]) code.

## References

- [1] EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 1: TTCN-3 Core Language "[http://www.etsi.org/deliver/etsi\\_es/201800\\_201899/20187301/04\\_04.01\\_60/es\\_20187301v040401p.pdf](http://www.etsi.org/deliver/etsi_es/201800_201899/20187301/04_04.01_60/es_20187301v040401p.pdf)"
- [2] L. PETRONE Reusing batch parsers as incremental parsers *Proceedings of the 15th Conference on Foundations of Software Technology and Theoretical Computer Science*. London, UK, UK: Springer-Verlag, 1995, pp. 111123. Available: <http://dl.acm.org/citation.cfm?id=646833.708027>
- [3] J.-M. LARCHEVEQUE Optimal incremental parsing *ACM Trans. Program. Lang. Syst.*, vol. 17, no. 1, pp. 115, Jan. 1995. Available: <http://doi.acm.org/10.1145/200994.200996>
- [4] W. X. LI A simple and efficient incremental ll(1) parsing *Proceedings of the 22nd Seminar on Current Trends in Theory and Practice of Informatics, ser. SOFSEM 95*. London, UK, UK: Springer-Verlag, 1995, pp. 399404. Available: <http://dl.acm.org/citation.cfm?id=647005.712013>
- [5] A. M. MURCHING, Y. V. PRASAD, AND Y. N. SRIKANT Incremental recursive descent parsing *Comput. Lang.*, vol. 15, no. 4, pp. 193204, Oct. 1990. Available: [http://dx.doi.org/10.1016/0096-0551\(90\)90020-P](http://dx.doi.org/10.1016/0096-0551(90)90020-P)
- [6] C. GHEZZI AND D. MANDRIOLI Augmenting parsers to support incrementality *J. ACM*, vol. 27, no. 3, pp. 564579, Jul. 1980. Available: <http://doi.acm.org/10.1145/322203.322215>
- [7] F. JALILI AND J. H. GALLIER Building friendly parsers *Proceedings of the 9th ACM SIGPLAN-SIGACT symposium on Principles of programming languages, ser. POPL 82*. New York, NY, USA: ACM, 1982, pp. 196206. Available: <http://doi.acm.org/10.1145/582153.582175>
- [8] KRISTOF SZABADOS Structural analysis of large TTCN-3 projects, Proc. 21st IFIP WG 6.1 International Conference on Testing of Software and Communication Systems and 9th International FATES Workshop *Lecture Notes in Computer Science 5826: Testing of Software and Communication Systems*, Springer-Verlag Berlin, Heidelberg, 2009 pp. 241–246.