

**An Executive Summary of Research in
Integrated Software Development Environments
Gene Fisher
16 January 2008**

Keywords: software development, development environment, IDE

An integrated development environment (IDE) is a collection of tools for developing software. The basic components of an IDE are a source code editor, compiler/interpreter, and debugger. Tight integration of IDE components makes the cycle of edit/compile/debug significantly more convenient than using separate, unintegrated tools.

Advanced IDE features include support for multiple languages, code browsing, graphical program views, testing, and integrated version control. Many advanced IDEs provide a well-defined framework for adding new tools. These frameworks are typically called &”plug-ins”, implying that new IDE functionality can be readily added.

Among the earliest IDEs were those provided for the Lisp programming language [7]. Integrated development has been part of Lisp environments since as early as interactive terminal access became available in the early 1970s. When graphical workstations came into use, Lisp IDEs were among the first to provide graphical code browsers and diagramming tools.

The period of the mid-1980s was the heyday of IDE research. Pioneering environments such as PECAN [6] introduced new forms of graphical program viewing, including trees, graphs, data diagrams, and tables. The BALSAM environment [3] was among the earliest to introduce dynamic program views in the form of algorithm animations.

Many of the features that were researched in the 1980s began to appear in commercial and open-source products in the 1990s, and continue to be refined to the present. The Eclipse environment [4] has risen to recent prominence. It provides a good set of basic tools, and a well-defined plug-in framework for the addition of new tools.

IDE research in the 1990s focused on the addition of software engineering features into integrated environments. IDEs that support a broad range of development activities are called process-centered software engineering environments (PSEEs) [1]. A PSEE provides support for all aspects of the software development process, from requirements engineering to maintenance. They also support software project and product management.

Research in the 2000s has provided some incremental improvements to IDE features introduced in the 1980s and 90s. Recent new developments have focused on expanding the scope of IDEs, with support for hypermedia (Chimera [2]), collaborative work (Jazz [5]), and ubiquitous computing (iRoom [8]).

References

- [1] AMBRIOLA, V., CONRADI, R., AND FUGGETTA, A. Assessing process-centered software engineering environments. *ACM Trans. Softw. Eng. Methodol.* 6, 3 (1997), 283–328.
- [2] ANDERSON, K. M., TAYLOR, R. N., AND E. JAMES WHITEHEAD, J. Chimera: hypermedia for heterogeneous software development environments. *ACM Trans. Inf. Syst.* 18, 3 (2000), 211–245.
- [3] BROWN, M. H., AND SEDGEWICK, R. A system for algorithm animation. In *SIGGRAPH ’84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1984), ACM Press, pp. 177–186.
- [4] ECLIPSE OPEN SOURCE COMMUNITY, T., 2007.
- [5] HUPFER, S., CHENG, L.-T., ROSS, S., AND PATTERSON, J. Introducing collaboration into an application development environment. In *CSCW ’04: Proceedings of the 2004 ACM conference on Computer supported cooperative work* (New York, NY, USA, 2004), ACM Press, pp. 21–24.
- [6] REISS, S. P. Pecan: Program development systems that support multiple views. In *ICSE ’84: Proceedings of the 7th international conference on Software engineering* (Piscataway, NJ, USA, 1984), IEEE Press, pp. 324–333.
- [7] SANDEWALL, E. Programming in an interactive environment: the “lisp” experience. *ACM Comput. Surv.* 10, 1 (1978), 35–71.
- [8] SCHOLZ, J., GRIGG, M., PREKOP, P., AND BURNETT, M. Development of the software infrastructure for a ubiquitous computing environment: the dsto iroom. In *CRPITS ’03: Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003* (Darlinghurst, Australia, Australia, 2003), Australian Computer Society, Inc., pp. 169–176.