



Status of the DAPSE Project:
A Distributed Ada Programming Support Environment*

Stowe Boyd

Massachusetts Computer Associates, Inc.
(COMPASS)
26 Princess Street
Wakefield MA 01880

The DAPSE (Distributed Ada Programming Support Environment) is an on-going engineering research project directed toward the synthesis and application of contemporary environment technology. The goals of the project are several:

- to construct an environment and initial set of tools distributed across a local area network of high-performance, raster-graphics workstations;
- to develop techniques by which a particular DAPSE may communicate with other environments (DAPSEs or otherwise);
- to develop capabilities — embodied in the environment and tools — which will support the development and maintenance of Ada systems, and in particular, distributed Ada systems;
- and to build the environment and tools by means of a DAPSE construction methodology, exploiting high-level specification language and allied generators to build tool and environment components automatically.

The DAPSE project is jointly funded by Ft. Monmouth (U.S. Army), Rome Air Development Center (U.S. Air Force), and the STARS program. Other reports on this work have been published [1,2].

- The initial toolset for DAPSE has been prototyped, and includes
- a graphical, Ada structured editor and an Ada library manager based upon grammar-driven graphical manipulation [3];
 - and a graphical, role-based user interface.

Use of high-level specification languages and allied generators has yielded components of the prototype DAPSE; this DAPSE construction methodology is based upon attribute grammar techniques [4,5] and recursive data structures [6]. The methodology has been used to generate the prototype environment, and certain development tools, and is based upon the use of a common specification language. Other generators — the front end generator, data (object and relationship) management

system generator, and the structured editor generator — may extend the semantics of the common specification language into their domains.

The DAPSE architecture provides a framework in which users are provided access to a set of typed objects and a well defined set of operations (tools) for those objects. These objects have attributes, and relationships to other objects. Several innovative aspects of the DAPSE architecture bear mentioning.

- Configuration control (CONCON) supports an integral, implicit mechanism based upon tracking named sets of objects. Its primitives include reconciliation (a means to block multiple updates), as well as creation, deletion, update, etc.
- User interaction with objects is role-centered and contextual, with defined contexts — user views of the database — called activities, and with operations known as actions. An action may be associated with pre- and post-conditions, which must be satisfied, respectively, before or after the action's execution.

During the next year, the DAPSE team will direct its efforts toward

- the extension of existing tools and generators (such as structured editors);
- the integration of a commercial, validated Ada compiler into the DAPSE;
- the integration of high-performance, entity-relationship-attribute methods;
- implementation of a remote procedure call toolkit, which provides a compile-time binding for multi-processor Ada applications, while conserving strong-typing [4];
- and the complete definition and prototyping of activity and action modeling.

References

- [1] M. Marcus, K. Sattley, S.C. Schaffner, E. Albert, "DAPSE: A Distributed Ada Programming Support Environment," Proc. IEEE 2nd Int. Conf. Ada Applications and Environments, 115-125, 1986.
- [2] M. Marcus, K. Sattley, C.M. Stefanescu, "Configuration Control in an Ada Programming Support Environment," Proc. Joint Conf. National Ada Technology and Washington Ada Symposium, March 1987, in press.
- [3] S.C. Schaffner, "Grammar-driven graphical interfaces", Massachusetts Computer Associates, CA-8609-0901, 9 September 1986.
- [4] N. Habermann, et al. *A Compendium of Gandalff Documentation*. Carnegie Mellon University, 1981.
- [5] T. Reps. *Generating Language-Based Environments*. MIT Press, 1983.
- [6] C.A.R. Hoare, "Recursive Data Structures," Int. J. Comp. Inf. Sci., 4(2), June 1975.
- [7] S.A. Schuman, E.M. Clarke Jr., C.N. Nikolau, "Programming Distributed Applications in Ada: A First Approach," Massachusetts Computer Associates, CA-8108-1201, presented at Int. Conf. Parallel Processing, 1981.

*This work was performed under contract DAA807-85-K572 for the U.S. Army, Ft. Monmouth, N.J.
(COMPASS Document CA-8702-2701)