



2

The Design and Development of Griffin
A Common Prototyping Language
Final Technical Report

Principal Investigators: Robert Dewar, Benjamin Goldberg, Malcolm Harrison,
Edmond Schonberg, Dennis Shasha
PI Institution: New York University
PI Phone Number: (212) 998-3495 (B. Goldberg)
PI E-mail Address: griffin@cs.nyu.edu
Grant or Contract Title: Griffin: A Common Prototyping Language
Grant or Contract Number: DARPA/ONR #N00014-91-J-1472
Reporting Period: 1 Oct 90 - 30 Sep 91

1 Productivity measures

- Refereed papers submitted but not yet published: 6
- Refereed papers published: 9
- Unrefereed reports and articles: 9
- Books or parts thereof submitted but not yet published: 2
- Books or parts thereof published: 4
- Patents filed but not yet granted: 0
- Patents granted: 0
- Invited presentations: 12
- Contributed presentations: 9
- Honors received: 0
- Prizes or awards received (Nobel, Japan, Turing, etc.): 0
- Promotions obtained: 0
- Graduate students supported: 5
- Post-docs supported: 0
- Minorities supported: 0

DTIC
ELECTE
MAR 02 1994
S F D

This document has been approved
for public release and sale; its
distribution is unlimited.

94 3 01 021

708

94-06795



**The Design and Development of Griffin
A Common Prototyping Language
Final Technical Report**

Principal Investigators: Robert Dewar, Benjamin Goldberg, Malcolm Harrison,
Edmond Schonberg, Dennis Shasha
PI Institution: New York University
PI Phone Number: (212) 998-3495 (B. Goldberg)
PI E-mail Address: griffin@cs.nyu.edu
Grant or Contract Title: Griffin: A Common Prototyping Language
Grant or Contract Number: DARPA/ONR #N00014-91-J-1472
Reporting Period: 1 Oct 90 - 30 Sep 91

2 Detailed summary of technical progress

2.1 Overview

The objective of the Griffin project at NYU is the design of a language, called Griffin, for prototyping large software systems. The success and cost-effectiveness of prototyping depends on, among other things, a prototyping language that demonstrates expressiveness, flexibility, and conciseness. Griffin is intended to satisfy these requirements.

The NYU approach to the design of Griffin is based on the premise that the prototyping of large software systems can be performed more efficiently in a language that is specifically designed to support software experimentation. This approach reflects experience with the Ada/Ed translator written in SETL at NYU (the first validated ADA implementation) and other medium-scale prototyping experiments.

The most important requirement of a prototyping language is that it provide the means to express the essential content of an algorithm, while permitting lower-level implementation details to be supplied by the system. Another basic requirement is that the language be easy to use and understand, so that program construction, debugging, and redesign are facilitated. An informal measure of the power of the language is the compactness of programs (compared with their implementation in a production language), though this is certainly not the only measure.

The Griffin language obtains its expressiveness from the adoption of such language paradigms as type polymorphism, inheritance, higher-order functions, powerful pattern matching primitives, generalized iterators and constructors, and expressive abstract data structures such as sets and mappings. Unlike other languages exhibiting these features, Griffin also supports parallel and distributed computation through tasks and generalized communication channels, and it facilitates transaction-based programming through its support for persistent objects.

In order to enhance the impact of the Griffin language on the large scale programming community, the Griffin project has formed a relationship with the Raytheon Company. During the design and preliminary implementation of Griffin, Raytheon, a developer of large scale Ada software systems, will constantly be reviewing the language design in order to judge its feasibility for prototyping. This review process includes the prototyping of components of actual Raytheon software systems. The final design of the Griffin language will reflect Raytheon's experiences using Griffin for software prototyping.

↓
<input checked="" type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

Availability Codes	
Dist	Avail and/or Special
A-1	

With the completion of the preliminary Griffin language reference manual, the Griffin project is entering a new phase of implementation and language review. Although the reference manual is still a working document, the language design is 90% complete. The first phase of the Raytheon review process has been initiated, with the selection of a set of sample programs to be coded in Griffin. One of these sample programs was recently implemented in Ada for a distributed on-board signal processor. At NYU, the implementation effort for Griffin is gearing up with the definition of the Griffin intermediate language already complete.

2.2 The Preliminary Language Reference Manual

At the January 1991 meeting of the Darpa Prototyping Technology effort, the Griffin preliminary language reference manual was released. It contained the first definition of the Griffin language to be made public, and included a number of examples to show how the features of the language are used. It is still a working document, and has been undergoing revision since its release.

2.3 The Griffin Implementation

A prototype implementation of Griffin has been under development for the past four months. It consists of an automatically generated parser, a type-checker and translator to an intermediate language, and an interpreter. The parser-generator (as well as the parser itself) is written in SETL. The type-checker and translator is written in Scheme, and the interpreter is written in Ada. This multi-lingual approach, often typical of prototyping, was used because a) the language chosen for each part of the implementation was the language of choice for the implementor, b) the parser generator was developed independently at NYU, and c) the interpreter had to be written in Ada to support Griffin's call-in/call-out interoperability with Ada.

2.4 New Developments in the Griffin Language

Many features of the Griffin language have been defined or revised over the past year. These features include:

- A flexible type system that includes: type signatures for bounded polymorphism, operator overloading and subtyping for object-oriented programming, and a type-inference mechanism that allows the user to omit most type declarations for conciseness.
- A model of persistent objects that retains the semantics of a program whether or not some of the objects are made persistent.
- A value semantics for each type in the language. This allows a clean semantic notion of parameter passing and assignment of abstract data types.
- A powerful exception facility that generalizes Ada exceptions by associating values with exceptions.

Each of these features are described in detail in the preliminary language reference manual.

**The Design and Development of Griffin
A Common Prototyping Language
Final Technical Report**

Principal Investigators: Robert Dewar, Benjamin Goldberg, Malcolm Harrison,
Edmond Schonberg, Dennis Shasha
PI Institution: New York University
PI Phone Number: (212) 998-3495 (B. Goldberg)
PI E-mail Address: griffin@cs.nyu.edu
Grant or Contract Title: Griffin: A Common Prototyping Language
Grant or Contract Number: DARPA/ONR #N00014-91-J-1472
Reporting Period: 1 Oct 90 - 30 Sep 91

3 Publications, presentations, reports, and awards/honors

3.1 Publications

- Cai, J., Facon, P., Henglein, F., Paige, B. and Schonberg, E., "Type Analysis and Data Structure Choice". IFIP TC2 workshop on transformational programming, Asilomar, CA. May 1991. (North-Holland, to appear)
- Hind, M. and Schonberg, E. "Efficient loop-level parallelism in Ada", TriAda conference, San Jose, CA, Oct. 1991.
- Goldberg, B. "Tag-Free Garbage Collection for Strongly Typed Programming Languages", 1991 ACM SIGPLAN Conference on Programming Language Design and Implementation, Toronto, June 1991.
- Park, Y.G. and Goldberg, B. "Reference Escape Analysis: Optimizing Reference Counting based on the Lifetime of References". ACM Symposium on Partial Evaluation and Semantics Based Program Manipulation, pages 178-189, 1991.
- Laufer, K. and Odersky, M. "Reflection in Type Systems." Proceedings, OOPSLA '91 Workshop on Reflection and Metalevel Architectures, Phoenix, Arizona.
- Laufer, K. and Odersky, M. "Type Classes are Signatures of Abstract Types." Proceedings, Phoenix/Esprit Workshop on Declarative Programming, Sasbachwalden, Germany, 1991.
- Fritz Henglein and Konstantin Laufer "Programming with Structures, Functions, and Objects." Proceedings, XVII Latin American Informatics Conference (PANEL '91), Caracas, also appeared as NYU-CS-TR 556. Presentation given at conference.
- Shasha, D. and Wang, "Optimizing Equijoin Queries In Distributed Databases Where Relations Are Hash Partitioned". ACM Transactions on Database Systems, Vol. 16, No. 2, June 1991, pages 279-308.
- Shasha, D. and Johnson, T. "The Performance of Concurrent B-tree Algorithms" ACM Transactions on Database Systems.
- Shasha, D. and Turek, J. "The Many Faces of Consensus in Distributed Systems". IEEE Computer (to appear)

- Shasha, D. and Turek, J. "Problems and Assumptions in Database Fault Tolerance" BD3, Lyon, France, September 1991.
- Shasha, D. and Rozen, S. "A Framework for Automating Physical Database Design". VLDB, September 1991, Barcelona, Catalonia, Spain
- Shasha, D. and Anderson, B. "Persistent Linda: Linda + Transactions + Query Processing". Workshop on Research Directions in High-Level Parallel Programming Languages, Mont Saint-Michel, France June 1991
- Sepe, R. "Very High Level Languages: Integration of Functional and Logic Paradigms, A Survey" 1991. *To appear in the Rivista di Informatica of AICA.*
- Pellegrini, M. and Sepe, R. "SETLOG: A Tool for Experimenting with New Semantics." *ACM SIGPLAN Notices*, 26(2):67-74, Feb 1991.
- Dewar, R. "Impact of Innovative Architectures", Chapter 10 in *Aerospace Software Engineering*, 1991, American Institute of Aeronautics and Astronautics, Inc.

3.2 Reports

- Dewar, R. et al. "Reference Manual for the Griffin Prototyping Language." Proceedings, DARPA CPL Working Group Meeting, May 1991.
- Dewar, R. Draft Report of ISO/TC22/WG9/CRG (Ada Character Rapporteur Group) on proposed internationalization of character support in Ada 9X.
Dewar, R. Proposed Ada binding for IEEE P754 (Floating-Point Arithmetic). Version 1.05. Submitted to ACM/SIGAda/NUMWG and ISO/TEC/SC22/WG9/NRG (Numerics working groups).

**The Design and Development of Griffin
A Common Prototyping Language
Final Technical Report**

Principal Investigators: Robert Dewar, Benjamin Goldberg, Malcolm Harrison,
Edmond Schonberg, Dennis Shasha
PI Institution: New York University
PI Phone Number: (212) 998-3495 (B. Goldberg)
PI E-mail Address: griffin@cs.nyu.edu
Grant or Contract Title: Griffin: A Common Prototyping Language
Grant or Contract Number: DARPA/ONR #N00014-91-J-1472
Reporting Period: 1 Oct 90 - 30 Sep 91

4 Transitions and DoD interactions

The interaction between the NYU Griffin group and its industrial partner, Raytheon, is beginning to prove very fruitful. With the completion of the preliminary language reference manual, Raytheon has begun to explore how Griffin could be incorporated in the prototyping of large software systems currently being proposed and/or developed. Recent visits by NYU team members to Raytheon proved extremely constructive and informative.

The Griffin project has had a substantial interaction with the Ada9X effort currently underway. In particular, one of the members of the Griffin project, Robert Dewar, is also a Distinguished Reviewer for Ada9X. In addition, Ed Schonberg and Robert Dewar head a project to investigate the implementation implications of language design decisions made by the Ada9X team. Through these connections, the design of Ada9X has had an effect on the design of Griffin. In particular, the Griffin team members are studying the proposed Ada9X object oriented programming facility in order to compare it with that of Griffin.

The Griffin team has been in contact with the other teams comprising Darpa's Prototyping Technology effort. Together with the University of Maryland/Honywell group, the Griffin team has explored issues of communication and program interconnections, such as communication between Griffin programs and programs written in other languages. There has also been interactions with the Yale/Software Options group to explore type models, with the Stanford/TRW group to explore real-time programming issues, and with the Duke/UNC/Kestrel group to explore models of parallelism.

**The Design and Development of Griffin
A Common Prototyping Language
Final Technical Report**

Principal Investigators: Robert Dewar, Benjamin Goldberg, Malcolm Harrison,
Edmond Schonberg, Dennis Shasha
PI Institution: New York University
PI Phone Number: (212) 998-3495 (B. Goldberg)
PI E-mail Address: griffin@cs.nyu.edu
Grant or Contract Title: Griffin: A Common Prototyping Language
Grant or Contract Number: DARPA/ONR #N00014-91-J-1472
Reporting Period: 1 Oct 90 - 30 Sep 91

5 Software and hardware prototypes

A prototype Griffin implementation is currently being developed. The implementation is about half completed. It consists of a parser, type checker, translator to an intermediate language, and an interpreter for the intermediate language. The implementation itself is an exercise in multi-lingual prototyping - three different languages are used: SETL, Scheme, and Ada! Ultimately, the implementation will be reengineered in Griffin, which encompasses the features of the above three languages that proved desirable for the Griffin implementation.