Multi-Paradigm Programming through Graph Rewriting
Final Report of EPSRC Grant GR/H 41300

John Glauert
jrwg@sys.uea.ac.uk
School of Information Systems, UEA Norwich NR4 7TJ, UK

The aim of the project was to explore the usefulness of graph rewriting as an implementation technique for multi-paradigm languages focussing mainly, but not exclusively, on declarative programming styles. The project built on experience using *Extended Term Graph Rewriting* to implement a range of programming languages and aimed for a secure theoretical framework for combining these languages, linked to an efficient implementation model. The stated objectives were:

Design a Core Implementation Model using Graph Rewriting
A model known as *Object Graph Rewriting* (OGR) was developed as planned. Early work used Dactl to simulate OGR, enabling the key design features to be identified.

Sequential Implementation of Multi-Paradigm Languages
The project focussed on the implementation of Facile, which combines functional and concurrent programming. The SML/NJ compiler from Bell Labs was used to implement a translator from Facile to OGR and thence to C. This system is known as OGRe for *Object Graph Rewriting Environment*.

Identify Primitives for Concurrent Computation
Towards the end of the project a design was produced for the *Parallel OGRe Machine* (POM). The design for POM was produced quickly, but was successful because a key design aim of OGR had been to allow highly efficient sequential evaluation while not hindering parallel implementation.

Parallel Implementation Experiments
The initial vehicle for implementing POM was PVM running on a network of conventional Linux workstations. This provided a straightforward development environment. Naturally the communication costs involved in PVM are too high to provide a real test of POM so a Transputer implementation using a Meiko Computing Surface is proposed for further work.

All of these original objectives were met successfully. Some unexpected, but highly significant, contributions also arose from the project:

Process Calculi and Graph Rewriting
In order to develop a semantics for OGR, work was done to simulate the model using π-calculus. With Thomsen and Leth, originally at ECRC, we are developing a new calculus, the π^{π}-calculus, based on direct communication between processes. This calculus is very closely related to OGR.

Theoretical Foundations of Multi-Paradigm Programming
A key issue in the implementation of multi-paradigm languages is the relationship between their models of computation expressed in terms of evaluation order and the forms of results expected. A very productive strand of work was pursued leading to a general theory of relative normalisation for rewriting systems.

The web page http://www.sys.uea.ac.uk/~jrwg/MultiPar/multipar.html will provide further details and point to many of the papers produced.