IST-2001-33149 (MRG)

Periodic progress report: year 4

26th June 2005

Project start date: 1 Jan 2002
Project duration: 40 months
Project coordinator: University of Edinburgh
Project partners: University of Edinburgh, Ludwig-Maximilians-Universität München

Funded by the European Community's "Information Society Technologies" Programme (1998–2002) under the FET proactive initiative on Global Computing.

Contents

1 Executive Summary

The MRG project started on 1st Jan 2002. The aim of the project is to develop the infrastructure needed to endow mobile code with independently verifiable certificates describing its resource behaviour. These certificates will be condensed and formalised mathematical proofs of a resource-related property which are by their very nature self-evident and unforgeable.

This report covers work carried out during a four-month extension that was agreed so that the project would still be running at the time of the planned final review. The extra time allowed work on some tasks originally planned for year 3 to be deferred in order to allow effort in year 3 to be concentrated on critical areas.

The MRG workplan is structured into ten technical workpackages, plus one administrative workpackage. Two of these (WP6 and WP8) were scheduled for activity during the extension period but work on two others (WP7 and WP9) was not complete after the end of year 3 so these have been active during this period as well. This was a period of wrapping up the project. All ten of the remaining technical deliverables were completed; in many cases this was mostly a matter of writing up work done during year 3.

The deliverables for this period are in the form of reports or software prototypes. All reports and software are downloadable from the project website. The content of the deliverables and key decisions are explained in Section 2.2 below, and Deliverable D11i compares progress with checkpoints and quality measures previously specified in the Self Evaluation Plan (D11f).

2 Work progress overview

2.1 Specific objectives for the reporting period

Our goal for the extension period is to complete all remaining tasks. These include tasks that were scheduled during year 3 but not completed during that period, and tasks that were scheduled for completion during the extension period. The tasks in question are listed in the table below.

Section 2.2 and the final self-assessment report (D11i) give an overview of the work done towards these objectives.

2.2 Overview of the progress of the project during the reporting period

The following table lists the deliverables that have been completed since the last report.

Del. no.	Deliverable name	WP no.	Est. person-months	Del. type	Planned delivery (month)	Complete?
D6g	Total correctness	6	18.7	prototype	4/05	3/05
D7a	Extension of type system by allowing user annotations [optional task]	7	4	report	6/04	1/05
D7c	Extension of basic type system to object-oriented language	7	6	prototype, report	9/04	6/05
D7d	Methods to infer type annotations automatically	7	10	report, maybe prototype	5/04	1/05
D7e	Extension of basic type system to mutable state and concurrency	7	2	report	9/04	3/05
D8a	Relationship between proof-based certificates and present-day security management	8	6	report	10/04	2/05
D8b	Certificate-based resource manager	8	6.5	prototype	4/05	3/05
D8d	Methods for estimating cost of native methods	8	2	report	8/04	1/05
D8e	Expressing and relating resource policies	8	4	report	4/05	5/05
D9b/c	Advanced techniques of certification	9	2.5	report	12/04	3/05
D11e	Workshop at end of year 3	11	0	workshop, proceedings	4/05	4/05
D11i	Final assessment of progress	11	0	report	4/05	6/05
D11k	Technological implementation plan	11	0	report	4/05	6/05

All remaining Workpackages (6, 7, 8, 9, 11) are thereby completed.

Below is a verbatim copy of relevant parts of the MRG workplan, augmented with a description of what work has been carried out under each workpackage and task and explaining significant departures from what was foreseen. Workpackages and tasks on which work was neither planned nor carried out during year 3 are not included.

WP6: Generation of certificates

Objectives: Define format of certificates and implement a certificate generator. Experiment with reducing size of certificates.

This package is concerned with generating mobile guarantees of resource boundedness. The guarantee, or *certificate*, is what will be shipped together with the code, as irrefutable evidence for the consumer that the code obeys the desired resource constraints. Here we consider the format of the certificates, and their generation.

g. From partial correctness to **total correctness** *(18.7 months)*
Deliverables: prototype implementation; technical report *Prerequisites: 2a/b,2c,3b,6a*
The Grail bytecode logic formulates and proves partial correctness assertions. This means that any such assertion will be true for a nonterminating program. Therefore, it is desirable to be able to detect and certify termination as well.

The reason for decoupling termination from correctness as we have done is that usually different methods are used to prove either so that the two properties are better dealt with in isolation. The alternative would have been a bytecode logic for total correctness with much more complicated rules for method and function invocation.

To address total correctness we will therefore implement a termination checker for Camelot. We hope to largely be able to use existing technology such as [LJBA03].

Output provided by the termination checker must be turned into a formal certificate of total correctness in the bytecode logic. As already mentioned, the bytecode logic in its current state cannot express termination. It must therefore be extended to cover this case. The idea is that we will have both partial correctness assertions and total correctness assertions (program terminates under a precondition and assertion is valid). We will prove a metatheorem allowing one to pass from a partial correctness assertion, obtained e.g. by certificate generation, to the corresponding total correctness assertion in the presence of a proof of termination, and implement this in the certificate generator.

There are programs that deliberately do not terminate such as user interfaces which take the form of a single infinite loop. In this case one must analyse termination and resource behaviour of the loop body in isolation. We will investigate this phenomenon in more detail and in particular develop a suggestive example.

Work carried out: A termination logic for Grail has been developed within Isabelle/HOL, building on the existing formalised operational semantics. Validity expresses termination of Grail program expressions under a given precondition. Formalised soundness of the termination logic is established and completeness is in progress. Examples of manually generated termination proofs include typical mutually recursive examples and use a table to lookup a well-founded termination measure for a function. To separate the design of this logic from the specifics of termination, we have been working on a variant of the logic that formulates insecure behaviour via a set of axioms and reuses the rules of the logic to express security properties. This research was completed and delivered in March 2005 as Deliverable D6g.

The work in this package is mainly applied research, involving the design and construction of software components. Working software from this package will be an essential part of the final overall system.

Package 9 is dedicated to advanced research topics with the motivation of reducing certificate size. Parts 6c and 6d in this workpackage address this issue. We expect these approaches to be fruitful but not the final word; the results will be useful input for 9.

WP7: Advances in high-level type systems

Objectives: Improve expressiveness, user-friendliness, and accuracy of the type systems developed in WP4 and WP5.

The goal of this workpackage is to improve expressiveness, user-friendliness, accuracy of the type systems developed under 4 and 5.

a. Optional: Improve accuracy by **allowing for user interaction**. *(4 months)*
Deliverables: technical report *Prerequisites: 4a,4b/c,5a/b*
The aim here is to augment the basic type system from 4 with user annotations in the form of supplied proofs based, for example, on the derived rules from 4a. Also more abstract annotations such as loop invariants could be considered here.

If very restrictive resource bounds are imposed (e.g. hard limits on stack size) we might have to give the programmer the possibility to implement certain critical methods directly in bytecode with corresponding certificates obtained by hand, supported by a theorem prover. The task here will be to enable smooth integration of such user-supplied and -certified routines with high-level code.

Work carried out: This was completed and delivered as combined Deliverable D7a/d in January 2005.

c. **Extend** basic type system **to object-oriented language** *(6 months)*
Deliverables: research paper; extension of implementation *Prerequisites: 3d,4b/c,5a/b*
The basic type system developed in 4 will be likely to encompass only the first-order side effect-free fragment of our high-level language. The aim here is to generalise to account for object-orientation. While we may be able to draw inspiration from encodings of object-oriented languages in functional ones [BCP97], genuine innovation is required for two reasons. Firstly, these encodings invariable rely on advanced features such as higher-order functions and (mixed-variance) recursive datatypes; secondly the compilation of object-oriented features will make use of the object-oriented structure of the VM rather than following an encoding. Our strategy will be to extend the previous work to encompass higher-order functions (perhaps restricted to linear [Wad90, Hof00]) and then tackle object-orientation proper.

Another issue that might arise would be an extension of the previous work to more object-specific resources such as "number of class and interface loads required".

Work carried out: Hofmann and Jost began work on this topic late in 2004. While good progress has been made on a working draft it would be premature to publish the results at this point. Instead it was decided to submit the results to the Symposium on Principles of Programming Languages whose deadline is 18th July 2005. A preliminary report on this work is Deliverable D7c, delivered in June 2005.

The delay here is mainly due to the fact that during initial investigations it was found that moving to an object-oriented target opens the way for making much more far-reaching improvements and extensions regarding in particular circular data structures, arbitrary aliasing, and mutable state.

d. **Type inference** *(10 months)*
Deliverables: research paper; implementation (optional) *Prerequisites: 4b/c,5a/b,7a*
The basic type system developed under 4 and the extension developed under 7a may rely on any number of user-supplied type annotations, for instance, recursive functions might be annotated by suggested bounds on their resource usage which are merely certified, cf. [CW00].

The aim here is to develop ways to infer such annotations automatically to a certain degree based on decision procedures for arithmetic inequalities [HP99], automata-theoretic methods [PWO97], unification [Mil78], program analyses, fixpoint methods [PS91], etc.

Work carried out: This was completed and submitted in January 2005 as combined Deliverable D7a/d.

e. **Extend** basic type system **to mutable state and concurrency** *(2 months)*
Deliverables: technical report *Prerequisites: 3f*
We will merely assess and elaborate the requirements for this extension possibly leading to future work.

Work carried out: Mutable state is addressed as part of the extension to object-oriented features discussed under 7c above. Remarks on types for concurrent extensions are included in Deliverable D7e, delivered in March 2005.

This workpackage forms part of the scientific core of the proposal. Successful completion will demonstrate that our proposal extends beyond the basic feasibility validated in 1, 2, 4. The goals set out here are ambitious but realistic.

WP8: Integration with existing security model

Objectives: Implement resource manager and relate proof-checking infrastructure to present-day security management.

The preceding workpackages have detailed a proof-checking infrastructure which advances the state of the art in security management capabilities. This workpackage will enrich our understanding of this infrastructure by relating it to present-day security management.

a. **Relationship with existing security management** *(6 months)*
Deliverables: technical report *Prerequisites: 6a*

One direct advantage of a security infrastructure based on certificates and proof would be the ability to safely disengage the existing security manager for any downloaded code which can be shown not to offer any potential threat to the host cf. [Gil00]. Thus a *dynamic* (run-time) security management overhead would be replaced by a *static* (load-time) security assessment cost for those mobile code routines whose certificate guarantees that it satisfies the resource requirements of the host. Where the attached certificate *cannot* provide this guarantee (or cannot *be shown to* provide this guarantee) then either all or some parts of the existing dynamic security management code will be needed to supplement the static security assessment.

Work carried out: This work has been completed and delivered as Deliverable D8a in February 2005. This reports on the novel integration of a resource-aware certificate checker and a modern virtual machine (the J2SE5.0 JVM) via "hooks" in the JVM known as "agents".

b. **Experimental implementation** *(6.5 months)*
Deliverables: experimental prototype *Prerequisites: 2c,8a,6a*

A prototype implementation of a certificate-led resource manager will be produced. This will provide a platform for further speculative research on developments in static security assessment. Recent work on Java-based agent models which work with the Java 2 security model [GP01] would provide the basis for further development here.

Work carried out: A web-based demonstration platform that incorporates the entire MRG proof-carrying code infrastructure was completed and delivered in March 2005 as Deliverable D8b. Key components are a certifying compiler for Camelot on the producer side, a high-level proof checker on the consumer side, and a light-weight encoding of certificates to enable automatic and independent verification.

c. **Milestone: Implemented resource manager.** *Prerequisites: 8b*

This has been successfully achieved by the delivery of Deliverable D8b.

d. **Resource typing of native methods** *(2 months)*
Deliverables: operational techniques *Prerequisites: 1b*

It will be necessary to have a least a conservative estimate of the likely cost of invoking the native methods of the virtual machine (that is, those methods which have no bytecode representation). We will develop estimation techniques for this purpose.

Work carried out: Deliverable D8d was completed and delivered in January 2005. It reports on the use of Markovian analysis in estimating the impact of garbage-collection ignorant native methods in the JVM on managed object allocations.

e. Expressing and relating **resource policies** *(4 months)*
Deliverables: technical report *Prerequisites: 6a*

Our type-system technology will at first allow us to express resource properties of the result of compiling individual functions in the high-level language. We would like to have a way to combine these properties and relate them to user-level statements about the resource usage of a whole program, perhaps dependent on input parameters. An example statement might be "for positive integer inputs n and m, executing the `main()` method requires heap space $32*n+16*m$". This level of description is appropriate for expressing *resource policies* of the code consumer. In this task, we want to investigate ways of expressing and relating resource policies. The aim will be to allow either of two routes towards resource usage checking: generating certificates according to a given resource policy, or attempting to show that a given certificate satisfies an arbitrary resource policy.

Work carried out: Deliverable D8e, which describes a way of extending the MRG infrastructure to deal with resource policies, was completed and delivered in May 2005.

WP9: Reducing size of certificates; negotiation vs. proof

Objectives: Explore alternatives to 100%-guaranteed certificates when these are infeasible.

This package is concerned with exploring possible alternatives in situations where the generation and transmission of 100%-guaranteed certificates is unfeasible for one of the following reasons:

- certificates exist, but are prohibitively large
- certificates can in principle be obtained, but only at prohibitively high cost (time and human resource needed for theorem proving, runtime of program analyses)
- certificates can in principle not be obtained due to influence of unknown or merely estimable parameters.

This workpackage is more tentative and visionary than the other ones. Progress will be strongly dependent on the results obtained in the other packages and the particular qualifications of the research assistants.

A minimum deliverable will consist of visionary articles fleshing out the ideas thus enabling future interaction and perhaps collaboration with others working on these issues.

b/c. **Advanced techniques of certification** *(2.5 months)*

Deliverables: visionary paper *Prerequisites: 6a,2a/b*

Here we plan to depart from the requirement of endowing mobile code with mathematical proofs of resource bounds.

The first idea is to consider interaction-based probabilistic certification. The general scenario will be as follows: A sends B a piece of code to be executed remotely. On the basis of the code and possibly random data B computes a challenge (in the style of "please give letters 3 and 7 of your online password") to which A must respond. B may then accept the code, reject it, or set another challenge.

The theory of *probabilistically checkable proofs* [AMS+92, Aro94] shows how based on a concept of polynomially-sized and polynomial time verifiable certificate (which we can assume here) such a protocol can be devised so that challenges have logarithmic size, responses have constant size, and the probability that A can give a correct response to a challenge although no certificate exists is $<50\%$ so that k independent rounds lead to an error probability of $\leq 2^{-k}$.

In spite of some encouraging recent work [HS00] the overhead on the side of A remains considerable. We plan to investigate feasibility of this approach in our context and study possible relaxations, for instance allow for larger size responses. We emphasize that our concern is not to advance the theory of probabilistically checkable proofs, but exclusively to harness it for the purpose of certification of mobile code.

A different idea is to consider certification by negotiation in the absence of rigorous proofs. The above-described protocol still requires that the sender A actually possesses a proof that his program meets the required bounds. It only reduces the amount of information that is actually interchanged. For the case where proofs are too difficult or impossible to obtain, we propose to investigate more liberal negotiation processes like the following:

- B provides a range of input parameters, A sends certificate which works only for this range.
- B challenges specific parts of the program. E.g. "you've got a write command in line XXX. Please convince me that this is fine." A then responds with a proof with assumptions that B may challenge again or accept.
- To cope with resource usage depending on unpredictable extraneous factors such as interaction patterns in concurrent systems or number of cache misses, we propose to investigate the use of probabilistic models such as PEPA [GH94, Hil96]. A and B would agree upon a probabilistic model to be used; A would then carry out the modelling and would provide B with verifiable results obtained in this way, for instance in the form of probability matrices or selected rows/columns thereof.

Work carried out: A PCP framework for certification of arbitrary type systems has been formulated and implemented. This research was completed and delivered as Deliverable D9b/c in March 2005. This appears to contain the first actual implementation of PCP. While in principle feasible, it was found that the reductions in size brought about by this method only take effect with proof sizes well beyond what is currently being used and required.

This package is appreciably more risky than the previous ones as we move further away from well-understood terrain in programming language theory. However, the possible reward will be

high, as size of certificates and overhead in their production forms the biggest foreseeable obstacle against wide practical use of resource certification. If it can be successfully overcome or at least the necessary foundations laid, we will have paved the way for practical resource certification in the context of global computing.

WP11: Project management, dissemination and evaluation

Objectives: Project management, dissemination and evaluation.

The project requires close collaboration between the two sites and provides many opportunities for dissemination. The purpose of this workpackage is to ensure that collaboration proceeds effectively and with attention to internal and external evaluation, while being able to take advantage of a wide variety of forms of dissemination for the results.

The small size of the project enables decisions about the overall technical direction of the project to be taken in close consultation with all of the people involved. Milestones and periodic meetings provide checkpoints where progress can be reviewed and plans adjusted if necessary. Meetings for technical coordination will be as follows:

- A kickoff workshop in month 2 plus workshops in months 11, 23 and 35. These will be attended by all project personnel, insofar as possible. One prominent non-EU expert will be invited to speak at each of the first three workshops at the project's expense; this will give a useful source of comment and advice without the need for project staff to visit these people individually. All of these workshops will be open to people from outside the project, with the final workshop being publicized more widely.

- Internal project meetings in months 7, 17 and 29. Each of these will include technical meetings on all active workpackages, and will be attended by all project personnel involved with those workpackages.

Individual visits are also planned for collaborative technical work.

The results of the project will be made available to all through a website set up at the beginning of the project and maintained under the direction of the Project Coordinator for the duration of the work. This is intended to give interested parties a view of the results as they accumulate. It will include at least the following:

- An introduction to the project including title, partners, and summary, with links to appropriate European Commission websites (GC, FET, IST and/or FP5).

- All of the deliverables and other publications produced by the project, as they are produced. Those that are most appropriate for external consumption will be given special prominence.

- A section (protected from access by non-project personnel) for working drafts, internal project documents, etc.

The results of the project will also be presented at appropriate conferences and published in academic journals. Many of these conferences take place outside the EU and this is taken into account in the travel budget.

The project workshops provide an opportunity for external participants to learn about the project's progress and to contribute their views. The final workshop will be associated with an established international conference for increased visibility; this event is intended more for disseminating the results of the project than for technical coordination and a proceedings is planned.

Linking workshops with the annual project evaluation meetings will allow more efficient use of the travel budget.

For self-assessment, each Workpackage Coordinator will supply in advance measurable criteria of progress/success for the different stages of the workpackage which will later be used to assess progress. This assessment of progress will take place in connection with each of the end-of-year project workshops.

e. **Workshop at end of year 3** *(0 months)*
Deliverables: workshop, proceedings

Work carried out: The Symposium on Trustworthy Global Computing is is taking place at Edinburgh in April 2005 as a satellite event of ETAPS; we regard this as completing Deliverable 11e. Although this will give valuable opportunities for cross-fertilization between projects, and dissemination of results to the wider community, it is too late for in-depth technical discussion and collaboration between MRG project participants and so we organized additional internal workshops in November 2004 and February 2005.

i. **Final assessment of progress** *(0 months)*
Deliverables: report

Work carried out: Completed and delivered in June 2005 as Deliverable D11i.

k. **Technological implementation plan** *(0 months)*
Deliverables: report

Work carried out: Completed and delivered in June 2005 as Deliverable D11k.

Most of the deliverables are allocated 0 person-months because this work will be done by the main investigators rather than the researchers who are employed by the project.

2.3 Comparison of planned activities and actual work

The goal for the extension period was to complete all remaining tasks:

- "tasks that were scheduled during year 3 but not completed during that period": **all achieved (7a/d, 7c. 7e, 8a, 8d, 9b/c)**

- "tasks that were scheduled for completion during the extension period": **all achieved (6g, 8b, 8e)**

The following charts record the per-workpackage per-site per-annum activity. Manpower is given in person-months and refers only to the researchers paid by the project. The estimated manpower figures are incomplete, since the figures that were required for the Technical Annex were per-workpackage per-site totals, and per-task overall totals, but not the detailed per-workpackage per-site per-annum totals that would be most relevant here. Where it is possible to extract information from the figures in the Technical Annex (for instance, when a Workpackage is scheduled for activity in a single year) then this is recorded in the chart; otherwise the entry is left blank. In such cases, partial information may be obtained by comparing the cumulative totals with the planned totals over the entire project duration.

	Year 1		Year 2		Year 3		Extension		Full project	
WP1	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	12	12	0	0	0	0	0	0	12	12
LMUMUN	0	0	0	0	0	0	0	0	0	0
Total	12	12	0	0	0	0	0	0	12	12

	Year 1		Year 2		Year 3		Extension		Full project	
WP2	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	2	2	16	16	0	0	0	0	18	18
LMUMUN	9.5	9.5	10	10	0	0	0	0	19.5	19.5
Total	11.5	11.5	26	26	0	0	0	0	37.5	37.5

	Year 1		Year 2		Year 3		Extension		Full project	
WP3	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	3.5	3.5	8.8	8.8	0	0	0	0	12.3	12.3
LMUMUN	5.5	5.5	0	0	0	0	0	0	5.5	5.5
Total	9	9	8.8	8.8	0	0	0	0	17.8	17.8

	Year 1		Year 2		Year 3		Extension		Full project	
WP4	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN		2		9		4	0	0	13	15
LMUMUN		4		2		5	0	0	11	11
Total		6		11		9	0	0	24	26

	Year 1		Year 2		Year 3		Extension		Full project	
WP5	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	0	0	0	0	0	0	0	0	0	0
LMUMUN	1	1	3	3	0	0	0	0	4	4
Total	1	1	3	3	0	0	0	0	4	4

	Year 1		Year 2		Year 3		Extension		Full project	
WP6	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN		0		1.5		22.41		6.2	22.2	30.11
LMUMUN		0		9		15		0	24	24
Total		0		10.5		37.41		6.2	46.2	54.11

WP6 is now complete, with resource expenditure at the Edinburgh site somewhat greater than planned.

	Year 1		Year 2		Year 3		Extension		Full project	
WP7	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN		0		5		17	0	1.51	17	23.51
LMUMUN		0		0		0	0	3	5	3
Total		0		5		17	0	4.51	22	26.51

WP7 is now complete, with overall resource expenditure somewhat greater than planned and skewed towards the Edinburgh site. This is not a true picture since much of the progress that has been made here is the work of a PhD student in Munich (Steffen Jost) who is not employed by MRG.

	Year 1		Year 2		Year 3		Extension		Full project	
WP8	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN		0		0		8.46		7.1	12	15.56
LMUMUN		1.5		1		4		4	6.5	10.5
Total		1.5		1		12.46		11.1	18.5	26.06

WP8 is now complete, with resource expenditure at both sites somewhat greater than planned.

	Year 1		Year 2		Year 3		Extension		Full project	
WP9	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	0	0	0	0	2.5	1.5	0	3.42	2.5	4.92
LMUMUN	0	0	0	0	0	0	0	0	0	0
Total	0	0	0	0	2.5	1.5	0	3.42	2.5	4.92

WP9 is now complete, with resource expenditure at nearly twice the (small) planned level.

	Year 1		Year 2		Year 3		Extension		Full project	
WP10	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	0	0	3.3	3.3	0	0	0	0	3.3	3.3
LMUMUN	0	0	0	0	0	0	0	0	0	0
Total	0	0	3.3	3.3	0	0	0	0	3.3	3.3

WP10 is complete, since the remaining task has been eliminated from the revised workplan.

	Year 1		Year 2		Year 3		Extension		Full project	
WP11	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	0.5	0.5	0	0	0	0	0	0	0.5	0.5
LMUMUN	0	0	0	0	0	0	0	0	0	0
Total	0.5	0.5	0	0	0	0	0	0	0.5	0.5

WP11 is now complete. Most of the tasks are allocated 0 person-months because the work is being done by the main investigators rather than the researchers who are employed by the project.

	Year 1		Year 2		Year 3		Extension		Full project	
ALL WP	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual	Plan	Actual
UEDIN	30	20	30	43.6	30	53.37	0	18.23	90	135.2
LMUMUN	24	21.5	24	25	24	24	0	7	72	77.5
Total	54	41.5	54	68.6	54	77.37	0	25.23	162	212.7

When the workplan was revised, the budget and the allocation of money remained unchanged. Thus the *total* planned resource per site per year for the project remains as it was in the *original* workplan, including the lack of planned resource for the four-month extension, despite the fact that the planned resource per year for most workpackages changed significantly in the revised workplan. The difference between planned man-months and actual man-months, without a change in the budget, is accounted for mainly by the appointment of more staff at the Edinburgh site that are more junior than anticipated in the original budget. In particular, a 50% post for a senior researcher at professorial level was included in the budget and was planned to be filled by the Project Coordinator who would be released from 50% of his normal duties to devote this time to technical work on the project, but this turned out to be impossible under Commission rules. This increase in manpower was an important factor in getting the project back on schedule during year 2 when some tasks turned out to be much more difficult than anticipated, and it has also been important in keeping the project on schedule during year 3.

2.4 World-wide 'state-of-the-art' update

There are no particularly relevant technical developments world-wide during the past few months that have not been mentioned in previous Periodic Progress Reports.

2.5 Previous review reports

Comments made by the reviews in their report on the second year of MRG are addressed in the Periodic Progress Report for year 3.

2.6 Planned work for the next reporting period

This is the final four months of the project; there are no further reporting periods.

3 Project management and co-ordination

Cooperation within the consortium has been excellent. The fact that there are only two partners has meant that collaboration is strong.

Project meetings. MRG organized one internal project workshop during the four-month extension period. This was held in Edinburgh on 2 March 2005 and was attended by Phil Wadler of Edinburgh University, who spoke about his plans for a programming language called Links for building web-based systems, in addition to project personnel. All projects in the Global Computing pro-active initiative will attend ETAPS in Edinburgh during 2-9 April 2005.

Collaboration. Most tasks involve contributions from both sites. Most collaboration is via e-mail, and in most cases researchers travelling to project workshops spent extra days before or after the workshop at the host site for collaboration.

Personnel changes. The following personnel changes have taken place during the four-month extension period:

Hans-Wolfgang Loidl left MRG at the beginning of March to work on the EmBounded project, while remaining at the Munich site.

Cooperation with other projects. Connections with other projects are as described in the Periodic Progress Report for year 3.

4 Cost breakdown

At the time of writing, detailed financial data is unavailable.

5 Information dissemination and exploitation of results

Dissemination of the results of MRG has been via publications, research talks and presentations, courses, the project website, software, organization of workshops and joint dissemination activities, and involvement with other projects at national and international level.

Publications. The following publications by MRG members have been appeared or been accepted during January–April 2005.

Invited papers: [AGH+05], [AH05]

Refereed journals and conferences: [BHMS05], [GP05], [GS05]

Non-refereed journals and conferences: [Cam05]

Other publications by MRG members that have appeared or been accepted during January–April 2005 are: [ALW05a], [ALW05b], [BCGH05a], [BCGH05b], [BCGH05c], [CGH05], [LMO04], [LS05], [OBF+05], [PS04], [Sta05], [ZTLM05]

Research talks and presentations. Talks to present the conference papers listed above, plus the following:

- Alberto Momigliano gave a talk on "Automatic certification of Resource Consumption" at Ecole Polytechnique in Palaiseau.
- Kenneth MacKenzie gave a talk on "Functional Programming and Resource Bounds" at the Scottish Programming Languages Seminar in Edinburgh.
- Hans-Wolfgang Loidl gave a talk on "Certified resource bounds for the functional language Camelot" at the Scottish Programming Languages Seminar in Edinburgh.
- Robert Atkey gave a talk on "A Calculus for Resource Relationships" at SPACE 2004 during POPL 2004 in Venice.
- David Aspinall gave an invited talk on "Proof Carrying Code for Resource Guarantees" at CASSIS 2005 in Nice.
- Don Sannella gave a talk on "Mobile Resource Guarantees" at Beihang University in Beijing.

In June, Don Sannella will give talks at Kyoto University and at several government and industrial research labs in Japan on the results of MRG.

Courses. Martin Hofmann will give a series of lectures on certification of quantitative properties of mobile code at the NATO Summer School in Marktoberdorf in summer 2005, in which he will survey MRG results and related work.

Project website. The project website is at `http://groups.inf.ed.ac.uk/mrg/`.

Software. Grail and Camelot are available for download from the project website, and all other software produced by MRG will also be made available there.

Organization of workshops.

- Don Sannella was General Chair of ETAPS 2005 in Edinburgh. Ian Stark was Finance Chair of ETAPS 2005 and David Aspinall was Publicity Chair.
- David Aspinall was organizer and programme co-chair of the User Interfaces for Theorem Provers (UITP 2005) workshop in Edinburgh.
- Don Sannella served on the Programme Committee of the Symposium on Trustworthy Global Computing in Edinburgh.

Involvement with other projects. See Section 3 above.

Exploitation of results. In the closing stages of the project, we have been attempting to cement relationships with some of the companies that have shown interest in MRG by setting up collaborative follow-on projects. So far, the following attempts to secure funding have been successful or seem likely to succeed:

- EmBounded is an FET-Open STREP project on resource bounds in embedded systems that will start in early 2005, building on the results of the MRG project and specifically work on Grail and the Grail bytecode logic. Partners include LMU München and AbsInt.
- MOBIUS is a FET-GC2 Integrated Project proposal that will build on the results of MRG. Partners include Edinburgh and LMU München and industrial partners include Trusted Logic, France Telecom and SAP. An industrial User Panel includes about a dozen companies from a range of relevant sectors of industry.
- The Edinburgh site have a proposal for a collaborative project with Helixion (a local SME in the mobile phone sector) on digital rights management and secure multimedia.

The Edinburgh group are in discussion with Motorola and colleagues at Edinburgh concerning a possible large project on security in automotive networks in which the PCC techniques developed in MRG would play a role. An EPSRC project in Edinburgh called ReQueST on applying MRG techniques to the Grid will begin in May 2005.

References

[AGH+05] David Aspinall, Stephen Gilmore, Martin Hofmann, Donald Sannella, and Ian Stark. Mobile resource guarantees for smart devices. In *Construction and Analysis of Safe, Secure, and Interoperable Smart Devices: Proceedings of the International Workshop CASSIS 2004*, number 3362 in Lecture Notes in Computer Science, pages 1–26. Springer-Verlag, 2005.

[AH05] David Aspinall and Martin Hofmann. Dependent types. In Benjamin C Pierce, editor, *Advanced Topics in Types and Programming Languages*. MIT Press, 2005.

[ALW05a] David Aspinall, Christoph Lüth, and Daniel Winterstein. Eclipse proof general: A generic interface for interactive proof. In *International Workshop on User Interfaces for Theorem Provers 2005 (UITP 2005)*, 2005.

[ALW05b] David Aspinall, Christoph Lüth, and Daniel Winterstein. Parsing, editing, proving: The PGIP display protocol. In *International Workshop on User Interfaces for Theorem Provers 2005 (UITP 2005)*, 2005.

[AMS+92] S. Arora, R. Motwani, M. Safra, M. Sudan, and M. Szegedy. Proof verification and intractabilitity of approximation problems. In *Proc. 33rd IEEE Symp. on Foundations of Computer Science*, pages 13–22, 1992.

[Aro94] Sanjeev Arora. *Probabilistic Checking of Proofs and Hardness of Approximation Problems.* PhD thesis, UC Berkeley, 1994. UCB Technical Report: CS-TR-476-94.

[BCGH05a] A. Benoit, M. Cole, S. Gilmore, and J. Hillston. Analyse quantitative de programmes applicatifs à base de squelettes. In *Proceedings of Journées Francophones des Langages Applicatifs (JFLA2005)*, Obernai, France, March 2005.

[BCGH05b] A. Benoit, M. Cole, S. Gilmore, and J. Hillston. Enhancing the effective utilisation of grid clusters by exploiting on-line performability analysis. In *Proceedings of Cluster Computing and Grid 2005 (CCGrid'05)*, 2005. To appear.

[BCGH05c] Anne Benoit, Murray Cole, Stephen Gilmore, and Jane Hillston. Realistic performance evaluation of skeleton-based grid applications using the Network Weather Service. *Computer Journal*, pages 1–19, 2005. To appear.

[BCP97] Kim B. Bruce, Luca Cardelli, and Benjamin C. Pierce. Comparing object encodings. In *Theoretical Aspects of Computer Software (TACS), Sendai, Japan*, September 1997. An earlier version was presented as an invited lecture at the Third International Workshop on Foundations of Object Oriented Languages (FOOL 3), July 1996.

[BHMS05] Lennart Beringer, Martin Hofmann, Alberto Momigliano, and Olha Shkaravska. Automatic certification of heap consumption. In Andrei Voronkov Franz Baader, editor, *Logic for Programming, Artificial Intelligence, and Reasoning: 11th International Conference, LPAR 2004, Montevideo, Uruguay, March 14-18, 2005. Proceedings*, volume 3425 of *Lecture Notes in Computer Science*, pages 347–362. Springer-Verlag, Feb 2005.

[Cam05] Brian Campbell. Folding stack memory usage prediction into heap. In *3rd Workshop on Quantitative Aspects of Programming Languages*, 2005.

[CGH05] M. Calder, S. Gilmore, and J. Hillston. Automatically deriving ODEs from process algebra models of signalling pathways. In *Proceedings of Computational Methods in Systems Biology (CMSB'05)*, Edinburgh, Scotland, April 2005.

[CW00] K. Crary and S. Weirich. Resource bound certification. In *Proc. 27th Symp. Principles of Prog. Lang. (POPL)*, pages 184–198. ACM, 2000.

[GH94] Stephen Gilmore and Jane Hillston. The PEPA workbench: A tool to support a process algebra-based approach to performance modelling. In *Proceedings of the Seventh International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Springer LNCS vol. 794*, pages 353–368, 1994.

[Gil00] Stephen Gilmore. Deep type inference for mobile functions. In P. Trinder G. Michaelson and H.-W. Loidl, editors, *Trends in Functional Programming (Volume 1)*, pages 40–48, 2000.

[GP01] Stephen Gilmore and Marco Palomino. BabylonLite: Improvements to a Java-based distributed object system. In *Proc. 4th CaberNet Plenary Workshop, Pisa*, 2001.

[GP05] Stephen Gilmore and Matthew Prowse. Proof-carrying bytecode. In *Proceedings of First Workshop on Bytecode Semantics, Verification, Analysis and Transformation (BYTECODE '05)*, Edinburgh, Scotland, April 2005.

[GS05] Stephen Gilmore and Olha Shkaravska. Estimating the cost of native method calls for resource-bounded functional programming languages. 2005. Submitted for publication.

[Hil96] Jane Hillston. *A Compositional Approach to Performance Modelling*. Cambridge University Press, 1996.

[Hof00] Martin Hofmann. Linear types and non size-increasing polynomial time computation. To appear in *Theoretical Computer Science*. See `www.dcs.ed.ac.uk/home/papers/icc.ps.gz` for a draft. An extended abstract has appeared under the same title in Proc. Symp. Logic in Comp. Sci. (LICS) 1999, Trento, 2000.

[HP99] J. Hughes and L. Pareto. Recursion and dynamic data structures in bounded space: towards embedded ML programming. In *Proc. International Conference on Functional Programming (ACM). Paris, September '99.*, pages 70–81, 1999.

[HS00] Prahladh Harsha and Madhu Sudan. Small pcps with low query complexity. *Electronic Colloquium on Computational Complexity*, 2000. Report No. 61.

[LJBA03] Chin Soon Lee, Neil D. Jones, and Amir Ben-Amram. The size-change principle for program termination. In *Proceedings of the 28th ACM Symposium on Principles of Programming Languages*, London, 2003.

[LMO04] Kung-Kiu Lau, Alberto Momigliano, and Mario Ornaghi. Constructive specifications for compositional units. In *LOPSTR 2004 post-proceedings*, 2004.

[LS05] Sam Lindley and Ian Stark. Reducibility and $\top\top$-lifting for computation types. In *Typed Lambda Calculi and Applications: Proceedings of the Third International Conference TLCA 2005*, Lecture Notes in Computer Science. Springer-Verlag, 2005.

[Mil78] Robin Milner. A theory of type polymorphism in programming. *Journal of Computer and System Sciences*, 17:348–375, August 1978.

[OBF+05] Mario Ornaghi, Marco Benini, Mauro Ferrari, Camillo Fiorentini, and Alberto Momigliano. A constructive modeling language for object oriented information systems. In *Constructive Logic for Automated Software Engineering*, Electronic Notes in Theoretical Computer Science, 2005. To appear.

[PS91] Jens Palsberg and Michael Schwartzbach. Object-oriented type inference. In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*, pages 246–161, 1991.

[PS04] John Power and Olha Shkaravska. From comodels to coalgebras: State and arrays. In *CMCS'2004: 7th Intl. Workshop on Coalgebraic Methods in Computer Science 2004*, volume 106 of *Electronic Notes in Theoretical Computer Science*, 2004.

[PWO97] Jens Palsberg, Mitchell Wand, and Patrick O'Keefe. Type inference with non-structural subtyping. *Formal Aspects of Computing*, 9:49–67, 1997.

[Sta05] Ian Stark. Free-algebra models for the π-calculus. In *Foundations of Software Science and Computation Structures: Proceedings of FOSSACS 2005*, number 3441 in Lecture Notes in Computer Science, pages 155–169, 2005.

[Wad90] Philip Wadler. Linear types can change the world. In *TC 2 Working Conference on Programming Concepts and Methods (Preprint)*, pages 546–566, 1990.

[ZTLM05] A. Al Zain, P.W. Trinder, H-W. Loidl, and G.J. Michaelson. Managing heterogeneity in a grid parallel haskell. In *Second International Workshop on Practical Aspects of High-level Parallel Programming (PAPP 2005)*, 2005.