



Palamos House #104  
66-67 High Street  
Lymington  
SO41 9AL United Kingdom

www.embecosm.com  
jeremy.bennett@embecosm.com  
Tel: +44 1590 610184

## **MILEPOST II Proposal**

### **Migrating the Compiler to Commercial Robustness**

#### **Background**

MILEPOST was a project funded by the EU through HiPEAC to add machine learning to GCC and other compilers to aid in the selection of compiler optimizations. It was particularly intended to help with generating optimal code for embedded processor cores. The project ran from 2006-2009, with MILEPOST GCC 4.4 released in June 2009.

By training the compiler on a selection of representative code, it could learn which optimizations best suited particular source code structures. Even by 2008, the developers were reporting average improvements in compiled code performance between 11% (ARC 725D) and 40% (Intel Xeon), with the best cases more than doubling their speed<sup>1</sup>.

However this was fundamentally a research project, and while it arouse a great deal of interest from potential commercial users, it is not yet ready for adoption.

This proposal would make MILEPOST a technology suitable for adoption by any commercial organization.

#### **Issue Addressed by this Proposal**

We can summarize the deficiencies in two groups.

1. Mainline GCC development has diverged significantly from MILEPOST in its implementation of a plugin infrastructure. For MILEPOST to be widely adopted it must be compatible with mainline GCC. This requires changes both to GCC and to MILEPOST.
2. It is very hard to deploy MILEPOST GCC, and particularly to port it to new target architectures. Tutorial material is needed for deployment and the porting techniques need to be documented. A number of features need to be added or modified for commercial use: for example it must be possible to use a private machine learning database, it must be possible to report bugs without having to supply the machine learning database and incompatible optimization sequences must be excluded.

#### **Project Structure**

We propose a three phase project

- Phase One will make the changes required to GCC and MILEPOST, and submit those proposals to the Free Software Foundation for adoption. Of necessity the changes will need to be compatible with GCC's design objectives, so a major part of the work will be updating MILEPOST. This will build on work already completed during 2010 by Jörn Rennecke and Embecosm. It is estimated this represents 6 months further full-time work by Jörn Rennecke, although this could be spread over 9 months to accommodate the requirements of the partners. A single reference architecture from a

---

<sup>1</sup> Fursin et al. *MILEPOST GCC: machine learning based research compiler*. GCC Summit, Ottawa, Canada, 2008.



project partner will be chosen for this stage of the work. This architecture must be different from any of the existing MILEPOST architectures to highlight issues with portability.

- Phase Two follows on from Phase One and manages the adoption of the changes within GCC mainline distribution. This is largely an exercise in negotiation and explanation, but it is certain to require further GCC changes, and in turn MILEPOST changes. This represents 4.5 months further work by Jörn Rennecke, but spread over 18 to 24 months.
- Phase Three will run in parallel with Phases One and Two, and involves preparation of tutorial material for deployment, documentation of the porting process, changes to installation technology to make deployment easier and MILEPOST modifications to support private databases and easy reporting. The approach will be proved by working with further project partners to port MILEPOST to their architecture. This represents a further 6 months work by Dr Jeremy Bennett of Embecosm, comprising 4 months on the documentation and implementations changes and 1 month each working with partners to port MILEPOST to their architecture. This is flexible, and further partners can be accommodated (indeed are encouraged).

### **Project Costs**

Embecosm will provide the technical expertise in partnership with INRIA. Other partners will be clients and provide technology and financial support as appropriate.

A gross cost of €160k/engineer year is used, which includes equipment and all overheads, other than travel to partners. All phases of the project are fixed price.

Negotiations with HiPEAC will provide up to 50% funding of the project.

- Phase One: Total €80k. Up to 50% from HiPEAC, 50% from lead technology partner.
- Phase Two: Total €60k. 50% from HiPEAC, 50% from lead technology partner.
- Phase Three: Total €80k. 33% from HiPEAC, 33% from each technology partner.

In the event of HiPEAC being unable to provide support at this level, Embecosm and the partners will share the balance of the costs.

### **Partners**

Embecosm provides skills in GCC porting and experience in negotiating with the Free Software Foundation's team for GCC adoption.

The *lead technology partner* will need an existing GCC port and commitment of €40k to Phase One and €30k to Phase Two. As well as early access to the technology, they will have the benefit of being the reference architecture for all documentation of the MILEPOST technology, which will represent the highest performance version of GCC available.

Other *technology partners* will need an existing GCC port and a commitment of €27k each during Phase Three. They have the benefit of early inclusion in the targets for which MILEPOST can be used.

Academic partners will contribute their expertise and be able to reuse the outputs of the project in their research and publications. However this is primarily a project to move MILEPOST into the commercial viability, and it is the commercial partners who are our primary performance.

### **Further Information**

Those interested in joining this project should contact Dr Jeremy Bennett at Embecosm (email: [jeremy.bennett@embecosm.com](mailto:jeremy.bennett@embecosm.com), tel: +44 1590 610184).