

# The MPlib Project

## *MetaPost as a reusable component*

As you probably know, MetaPost development has restarted approximately two years ago. After a period of investigating user demands, it has now become obvious that MetaPost is showing its age.

The problems lie not so much in the actual drawing language that is used, but in the 1980s Metafont legacy that is very noticeable in the way the program interacts with the user and in how it deals with the computing environment in general.

Some of the big user-side problems that resurface on a regular basis are:

- The model used for the handling of external labels is outdated.

Running a per-file preprocessor to create the labels was already problematic before, but it is getting worse now that both T<sub>E</sub>X and Troff are moving away from their traditional output formats.

- All number handling is based on fractions of a 32-bit integer.

User input often hits one of the many boundaries that are a result of that. For instance, no numbers can be any larger than 16384, and there is a noticeable lack of precision in the intersection-point calculations.

- MetaPost cannot be used as a system-level service.

In fact, MetaPost cannot even be used as a system-wide library, because the many global variables make it non-shareable.

- Lack of 3-D support.

Even technical drawings that are nominally considered to be two-dimensional, like the ones in highschool math and physics books, often need to handle projections of 3-D objects to a plane.

Much of the needed development to fix these issues can be done in the normal course of events, because the needed extensions or changes to the program are isolated to a small section of the source code (this is for instance true for 3-D projection support), or because the needed changes are so well understood that it is trivial to make many changes (this is true for upgrading the 32-bit internal calculus).

But the handling of labels and the lack of system integration require massive changes to the source code

as well as to the build system, and therefore it was very unlikely that this would ever get done without extra incentives: a significant amount of time and effort that has to be dedicated to those particular problems.

An estimate of the needed programming hours to turn MetaPost into a modern, re-entrant system library with a modern form of inter-process communication was created:

Converting from the use of hundreds of global variables into a data structure that is passed on from one function to another	200
Adding a unified redirection layer for the input and output, allowing files as well as buffers to be used	100
Designing and implementing a new subsystem for label typesetting	150
Adding an interface for configurable default error responses	50
Total	500

If writing documentation is included in that estimate, it makes for six months of full-time (40 hours a week) programming, time that simply could be allotted within anybody's free hours in any way. It was clear to us that, to get these tasks done within a reasonable time frame, at least some of the work would have to be done during office hours. And that requires money.

So, a funding proposal was written and at the Dante 2007 meeting Hans proposed this new project for funding. Dante immediately stepped in for 50% of the requested amount (6000 euro) and within a week other user groups joined in as well: TUGIndia (1000 euro), TUG (1500 euro), NTG (2000 euro) and CSTug (1000 euro). Currently 500 euro is still missing, but we are confident that this gap will be bridged or overcome.

Work will start in the autumn of this year, and it is our current estimate that the project will be complete by the summer of 2008. The actual programming will be carried out by Taco. Hans Hagen will lead the project, and Bogusław Jackowski will be in charge of quality control.

Hans Hagen & Taco Hoekwater