IfA/MHPCC Image Processing Pipeline Pilot Project: IfA Overview

Nick Kaiser
Pan-STARRS Principal Investigator
Institute for Astronomy
4 December 2003
## Revision History

<table>
<thead>
<tr>
<th>Version/Revision</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4 Dec 2003</td>
<td></td>
</tr>
</tbody>
</table>
IPP Pilot Project Review: IfA Overview

1. Comments on the Process

IfA's understanding of the process is the following sequence:


This is based on the "Scrum" philosophy: a rapid/agile development cycle. The goal was to get moving fast, possibly cutting corners, and to terminate within a fixed time, rather than to push to complete fulfillment of requirements. The process is terminated by a review and analysis, followed by a re-launch of the cycle with new goals.

As this is the first step on a large project, the IfA anticipated that the product might be prototype at best and might possibly be "throw-away". As well as making a start on developing the system, part of the motivation was sociological; for the IfA algorithms team to get a feel for what can reasonably be expected from MHPCC and vice versa.

1.1 Requirements

The final instructions from IfA were supplied 10/14 as in the following form (also, see PSDC-410-001-02):

- High level specification of minimal.desired functionality.
- Data format specification.
- Coding standards.
- Meta-data requirements.
- Documentation - full documentation.
- Module I/O specifications - minimal I/O requirements + anticipated needs.

In retrospect, requirements should have been lower level, with more concrete specification of what should be provided in public the API.

1.2 Design review

IfA expected delivery of a "mini-SDD" specific to the pilot project. Instead, the design review process consisted primarily of mapping of requirements to sections of the existing MHPCC design document. While adequate to associate algorithms/modules, this did not allow review of any architecture issues such as control interfaces, code organization, accessibility of modules, utilities nor any detailed review of algorithms. Design review is really happening now.
1.3 Coding/Documentation Issues

The MHPCC team are to be commended on delivery of working code in a very short time from a very small team of SW engineers. While the algorithmic content of pilot project was quite small, IfA had anticipated more interaction between coders and scientists on algorithmic details. Review of the code showed widespread failures to adhere to coding standards. That, in itself, should be easily remedied. However, there were also problems with the delivered software at many levels. These are addressed in detail elsewhere (see PSDC-400-002-01) and in Dr. Lupton's presentation.

Regarding documentation, the User manual was found to be clear, concise and useful, but current documentation of the actual code and algorithms is minimal.

There was some disappointment that meta-data generation and handling was not implemented in the released version.

2. Major Design Concerns

The IfA team had expected the API to provide access (SWIG) to modules and functions developed for the pilot project. The MHPCC team vision was for pipeline programmer interface to be entirely through the IPP "engine" via a process control language. While IPP engine is adequate to test functionality of pilot project modules, and provides an effective way to link inputs and outputs of modules it is a batch process oriented interface and, as it stands, cannot be directly implemented within the existing prototype parallel system. The IfA team have concerns that this interface may be too restrictive, or that to make it usable in a broader context would be a major effort in designing a new language, when existing languages may be more efficient.

IfA expected a more layered approach to the library design and code organization consisting of:

- Low-level external libraries (cfitsio, fftw...).
- PS interfaces to external libraries (so we are not wedded to specific FFT etc).
- Low level utilities (error handling, memory allocation etc).
- Toolkit of functions for e.g. list statistics, low level
- Image manipulation routines, image arithmetic, data modelling,
- Image display tools etc.
- Modules
- Control system and user interface

Some of this was done, but there was little distinction between general purpose tools and modules. IfA expected that public API would descend at least to the level of toolkit. However, this was not made specific in the requirements.
3. Major Lessons Learned - IfA Perspective

We have obtained a useful metric of what can be achieved by a small team of SW engineers in a limited time. This will be invaluable in planning future scope of work. IfA team have a much better understanding of the process of interacting with MHPCC and the MHPCC approach to design philosophy (process and architecture). There is clearly a need for the requirements to be pitched at a lower and more detailed level and to be more rigorously defined.

The IfA feels that a major problem has been communication between the groups; this being a human rather than technical problem --- while SourceForge was probably not used very effectively, it seems a promising mechanism. IfA team need to be more involved in the design review process, API scope definition, algorithm selection and implementation. Many of the problems we have identified could have been avoided if the design ideas and code had been made available at an earlier stage. Also there is not yet a real sense of this being a team effort: more face-to-face interactions are needed to cement the working group.

MHPCC need to more strongly enforce coding/documentation/CVS standards (this should not be an IfA responsibility). Future expansion of the MHPCC team should bear this in mind. We had hoped to be able to review documentation standards, but in the absense of much aside from the users manual, this will have to wait for the future.