

ATNF ATUC MEMORANDUM

To: ATUC
From: Neil Killeen
Date: 7 June 2004
Subject: **Marsfield Scientific Computing Group Report**

1) AIPS++ International Collaboration

An MOU has been signed by NRAO, ATNF and ASTRON (three of the four active aips++ consortium members). The purpose of the MOU is to recognize the mutual dependencies and fashion a new collaboration regarding maintenance and development of AIPS++.

The MOU became effective on April 22, 2004. Core elements are

- We do not fork the code
- Changes to core areas require agreement between the parties
- Maintenance of modules is continued by the previous maintainers
- Development targets are available to all parties. Only development of core targets must be agreed to. Tradeoffs can be made for 'in kind' development'
- Weekly reports are continued
- Representatives of the parties meet monthly by phone (2 meetings so far)

Getting this MOU has been vital. Without it, probably NRAO would have changed the system as they saw fit to support their ALMA activities (for which AIPS++ is the offline reduction package).

So far this is working reasonably well. The switch to cvs for code management was managed successfully. Currently we are in the process of jointly working out how to replace the 'Tasking' system of AIPS++ by a modern Corba-based system bound to Python. This is much more complex activity and is driven largely by the imperatives of ALMA (although we have all wanted it for a long time); this will stress the MOU more. We are yet to really negotiate any 'in kind' tradeoffs, although some may eventuate in the next few months.

2) Application Development

Single Dish

SPC Replacement (Malte Marquarding). The Requirements document has now been finalized and the community engaged with us pretty well in doing this. There is a lot of requested functionality! As well as this milestone, the project met the large majority of its targets for the June ATNF Projects meeting.

The system is being built with Python as the scripting language, connected via the Boost wrappers to AIPS++ C++ classes as appropriate. Plotting will probably be

done via the package matplotlib.

At the time of writing, we can fill data (from SDFITS, RPFITS, MeasurementSet) into an internal structure based on AIPS++ Tables (memory or disk) select some data and plot it. This rapid development was enabled by the high reuse factor (generic AIPS++ modules and multibeam AIPS++ modules).

We are working on producing a 'demonstrator' which will form the basis of a report requested by the Steering Committee. DJ Pisano and Juergen Ott will work with us to partly reduce some Mopra data they plan to take in July. They have produced a good requirements document. This plus our existing prioritization of the items in the Requirements document drives our activities for the next development cycle (to September).

By the end of this cycle, other friendly astronomers will be asked to help in the evaluation process. We anticipate making the first public release by the end of the year.

To speed up delivery, Mark Calabretta is now also partly working in this project in the areas of interface and archiving (SDFITS).

Multibeam.

We have moved Multibeam development into the ATNF Projects structure now, providing a more visible mechanism for managing its activities. Roman Feiler (works on Faraday s/w at the University of Torun) has been visiting us for a few weeks to work on finalizing the AIPS++-based multibeamview display module. We will deploy this initially in parallel with the Karma-based module with the plan being to replace the Karma one (which causes a lot of maintenance/installation headaches).

Remote Visualization Server.

The initial year of development (Anil Chandra) for this project was largely about prototyping ideas to assess whether the concepts and technology were viable. We have decided they are and we are continuing this project for another year. We have hired a second person (Praveena Tokachichu) to work in this area too.

The RVS project met the vast majority of its targets in the first year. By the end of the next Projects cycle (early September), we anticipate deploying RVS to existing ATNF image archives (e.g. SGPS). By the end of the year we anticipate packaging RVS so that it can be made publicly available (i.e. other people can use it in their archives).

RVS takes a request to display an image (specified by a URL to a FITS file) and transmits it from the users browser to a server via the SOAP protocol. The server fetches the data, and renders it via the AIPS++ display library (using CORBA). The rendered result is displayed back to the user's desktop via Java.

What makes this different from other display applications is that it is largely server-side (the data goes to the server, not the client) rather than client side. This means its niche is for visualization of images in archives.

ATCA online archive.

The ATCA archive is hosted by the ICTCentre in Canberra. The plans are:

- Make it (i.e. download RPFITS files) available publicly by end July (Christopher Owen)
- Attach a basic automatic pipeline by September (Tara Murphy, see below)
- We also have some activity via Aus-VO to make and maintain a mirror copy (at APAC). This is just so we can start to explore the idea of having a Data Centre look after our major archives.

ATCA Pipeline

A software pipeline is being constructed (Tara Murphy) to

- Attach to the ATCA archive to automatically make images
- Attach to the online system at Narrabri
- Be available generically offline

This is partly in collaboration with the ICTCentre. The initial deployment to the archive should be available by the end of the next cycle (early September). This will be for limited modality (continuum, no mosaicing) only. After that we will start to push into the more complex areas (spectral line to follow) and the other deployment areas.

The pipeline is being developed with AIPS++ modules. Of our projects using AIPS++, this one is probably riskiest. This is because the synthesis software is not in our maintenance domain (although we do have expertise in Mark Wieringa), is the most complex part of AIPS++ and is still not as robust as it needs to be in this area. Mitigating against this are 1) the fact that NRAO is heavily focussed on making the synthesis s/w robust and efficient (driven by ALMA) and have made a lot of progress in the last year 2) We now have our MOU to provide a structure to ensure we can cooperate as need be and 3) The design of the pipeline is generic and if we really had to, we could fall back to Miriad to implement some parts of it (but there are areas such as automatic editing, vital to the pipeline, that Miriad does not support). At this point, progress is satisfactory in my opinion.