LOCUS Requirements Document

There is a current need for a new software system to allow control of the CCDs used in the instruments at the DCT. The old LOIS system is difficult to understand and contains a lot of unused legacy code. In addition, software systems are needed for handling the writing of FITS format data files, focusing the telescope, and guiding the telescope. All of these software systems need to communicate using the ActiveMQ message broker which will also be used to communicate with the user GUI software, the “Moving Parts” software(JOE), and the various aspects of the telescope control system. All of this software should be broadly configurable using configuration files and should produce various levels of log files that report progress as well as errors.

System and Software Requirements

These requirements will need inputs from the the commissioning scientist, the instrument scientists, the telescope operators, the software developers, and the astronomers/users.

- The new software will be written in Python, its structure will be well documented, and APIs will be provided to allow libraries to be easily used in future software development.
- The software will be object oriented and classes will conform the the “single responsibility” principle documented in Robert C. Martin’s “Clean Code”.
- The software development will be guided by unit tests when possible, which will provide a detailed testing suite when the project is done.
- A programmer's guide and a user’s guide will be developed in parallel with the software.
- The software will be designed with extensibility and modifiability in mind.

- The system will respond to commands sent from various other software systems via the ActiveMQ broker.
- The system will allow the user to initialize the camera CCDs, set up CDD parameters, and take images. These images will be sent to the FITS writing software via the ActiveMQ broker where they will be written to disk files.
- The system will also allow some automated sequences and repetitive observations to be configured via some type of scripts. These observations include telescope focus sweeps and guiding

Project Constraints

The new system will need to work within current overall structure of the software currently running at the DCT except for LOIS. This software will replace LOIS. Eventually, other parts of the current software system at the DCT will be replaced. The software will run on Linux and,

perhaps, OS X. Since the software will be written in Python, it should, generally, be portable to most computers.

Interface Requirements

The system will communicate with other software using the ActiveMQ message broker. FITS files will be written to disk and documentation will be generated on the fly to allow users to easily find the data they are looking for. The various parts of the system will display an optional console that can show progress and/or errors in real time. The new software will use the ARC device driver to interact with the ARC controllers. The new software will interact with the current LOUI user software. The new software will interact with the current JOE software. The new software will interact with the TCS, AOS, and other LabView software.

Software Users

LOCUS will be used, in the main, by other software to acquire data from the DCT instruments and provide the resulting data back to those other software systems. There will be some direct access provided to LOCUS for testing and debugging either by a command line interface that is scriptable or by a test GUI program.

Improvements over current LOIS system

What things do we need to do better in the new system relative to the current system? We know that we will have to provide more accurate start times for images (in the FITS headers) than the current system. What other things can we improve as we replace the current software? Do we need to improve guiding on non-sidereal objects, for example?

Current Thoughts About Structure

There will be a library, currently called ArcCam, that encapsulates the communications with the CCD controllers. There will be a test system, currently called CamCom for direct testing of ArcCam and the DSP codes.

LOCUS will use an ArcCam object to control the CCD and get data from it and will communicate on the ActiveMQ broker to listen for commands and configuration and to send the resulting data to other parts of the system.

LoFITS will be in charge of producing and writing FITS files. It will also fill in some sort of log or database describing these files, what the target is, when it was taken and by whom (etc).

LoFocus will be in charge of running focus sweeps and calculating the best focus value for the current telescope/camera configuration. It will communicate best focus to the TCS.

LoGuide will take images and guide the telescope by providing information to the TCS.

Specific Requirements

This section will contain more specific requirements once the general requirements are known. Specs, Design, Classes, Broker topics and message contents, etc.

Use Cases

To be provided by scientific staff and instrument specialists. For example, should we be able to guide the telescope using LMI instead of RC1?

Future Iterations

Discuss future updates to LOUI and perhaps JOE.
Should the software be updated to allow the control of current slit-viewing cameras?