

KOSMOS System Design Note 2.03

Title: KOSMOS User Software Selection
Author: Jay Elias
Revision: 3
Date: April 15, 2010

Introduction

This document discusses the selection of the KOSMOS User Software architecture. Basic requirements are discussed in the KOSMOS User Software Requirements document (SDN 2.02).

Detailed observing scenarios are laid out in the Operations Concept Document (OCD).

General Principles

We assume that the user software should be based on an existing framework, both because this should (in principle) reduce the overall effort, and because there is value to starting with a system that is familiar to a significant number of staff and users. However, the software must meet the requirements outlined in the requirements document.

During initial discussions, three alternatives were identified, as discussed below in more detail:

- Adapt the OSMOS software
- Adapt the NEWFIRM/MOSAIC1 software (NOCS)
- Adapt the WIYN Bench/WHIRC software (MOP)

Each option has pluses and minuses.

OSMOS Software:

Adaptation of the OSMOS software would be done primarily by OSU staff, with NOAO responsible for the detector control software and for documenting interfaces. A block diagram of the OSMOS software is shown below. Note that there are three alternative user interfaces, where the iTerm is an engineering interface and the OSMOS GUI is the final operational user interface. The Prospero interface is intended for commissioning and initial use and would not be used for KOSMOS since the OSMOS GUI will be available before anything is needed for KOSMOS.

If the OSMOS software were adapted for KOSMOS, the changes that would be needed are (1) changes to the telescope interface; (2) changes to the detector control interface; and (3) changes to the data handling. These are probably listed in order of increasing complexity – this is because data handling involves integration of header and pixels as well as passing the complete frames to the NOAO archive process.

The OSMOS GUI will be different from anything else at KPNO, although it will undoubtedly be simple enough to learn quickly.

The OSMOS software provides for scripting through editing files of command lines (one would undoubtedly start with a collection of useful templates). The command line also provides a means of doing unusual but safe actions.

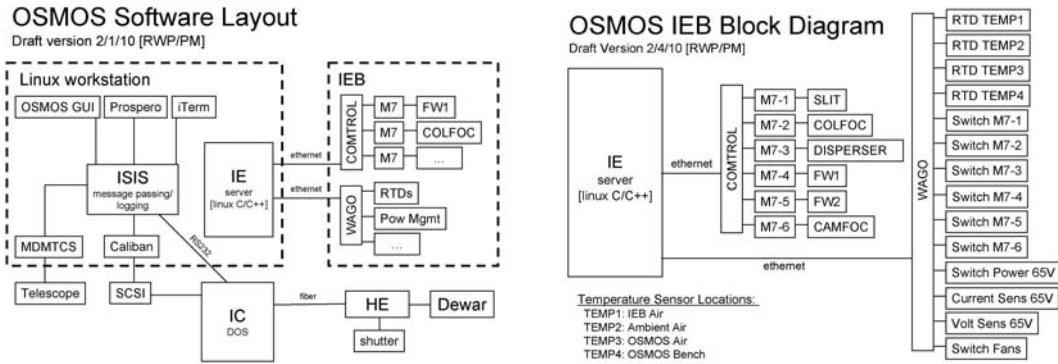


Figure 1 – OSMOS Software Layout (left) and IEB Block Diagram (right). Note that one of the “Ethernet” connections between the IE and IEB (on the left-hand panel) should in fact run between the ISIS and IEB.

NOCS

The NOCS software was originally designed for use with NEWFIRM but has a "observation"-centric data acquisition paradigm outlined by the message sequence chart of Figure 2. Higher level software combines variations on this sequence to achieve well-defined science recipes.

Inherent in the NOCS paradigm is scripting, preferred over other methods of data taking, especially for instruments like NEWFIRM which can take lengthy sequences of images for mapping. To the observer, a modest utility NGUI is provided with which they can prepare observing sequences (recipes/scripts) in advance or at the telescope. To execute a given observation, they run the script with no further command line parameters. Each script is self-contained, self-documented and (usually) contains multiple observations to achieve a particular science objective.

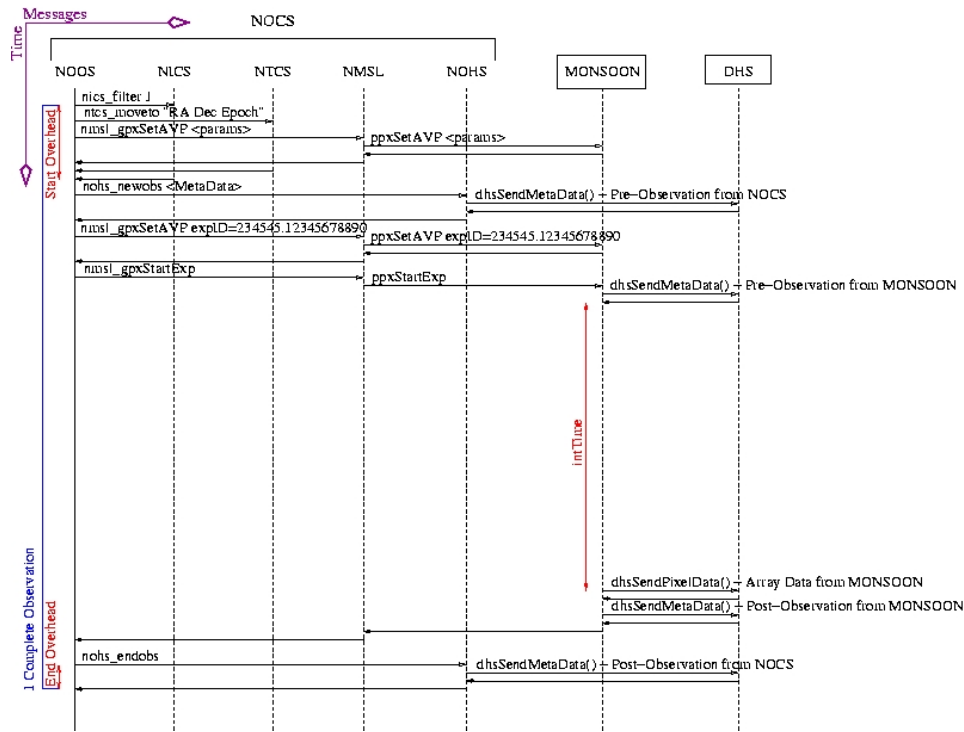


Figure 2 – NOCS observation paradigm.

The NOCS has 2+ years of telescope time under its belt and is well understood by mountain staff. In recent months, the NOCS has been selected as the platform of choice for the Mosaic Upgrade Project, and NEWFIRM itself has been shipped to Chile for operation on the CTIO Blanco 4-m telescope. Therefore, there is demonstrable expertise in handling the interface at several telescope on multiple sites.

The data flow diagram of the NOCS is shown in Figure 3 but this is more complicated than KOSMOS needs since only 1 Torrent system is required. The NOCS is already integrated into the 4-m telescope environment for guider, telescope and dome flat control. Additional interfaces will be needed for the 3 user-selectable "moving parts" of KOSMOS (slit wheel, filter and disperser). The KOSMOS software may need to control the focus stages but almost certainly not as a user-selectable parameter. The NOCS already has an instrument control element (for filter wheels in NEWFIRM and MOSAIC) which can be adapted with modest effort to control the other elements. (By this time next year, some variation of the NOCS will have run on the Blanco 4-m, Mayall 4-m and WIYN 0.9-m telescopes).

The NOCS scripting does not currently interact with images, which would be needed for initial acquisition scripts. None of the current implementation interact with the 4-m cass rotator/guider but this should be simple to include.

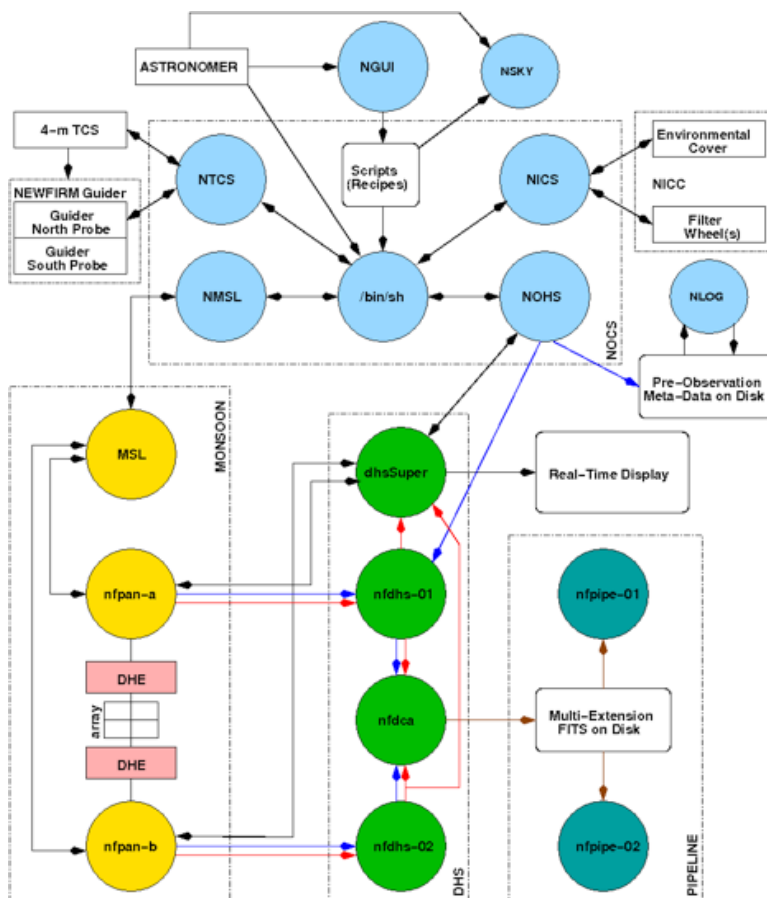


Figure 3 – NOCS data flow. The MONSOON/DHS sections can be simplified for KOSMOS, which requires only a single controller.

The NOCS is fully integrated into the NOAO E2E (end-to-end) data system and provides pipeline-ready products if required and/or automatic ingestion into the NOAO science archive.

If the NOCS is chosen as the preferred interface, observers at the 4-m using NEWFIRM, MOSAIC1.1 or KOSMOS will, in effect, be presented with the same observing interface and need not re-learn instrument specific stuff. Indeed, the only part of the system they need care about is NGUI which is separable and available as a downloadable utility for teaching new users about the system.

MOP (Monsoon Operating Platform)

The MOP is currently in use at KPNO on two WIYN instruments (Bench Spectrograph and WHIRC) and is being implemented on the KPNO 2.1-m telescope for imaging.

A block diagram is shown below.

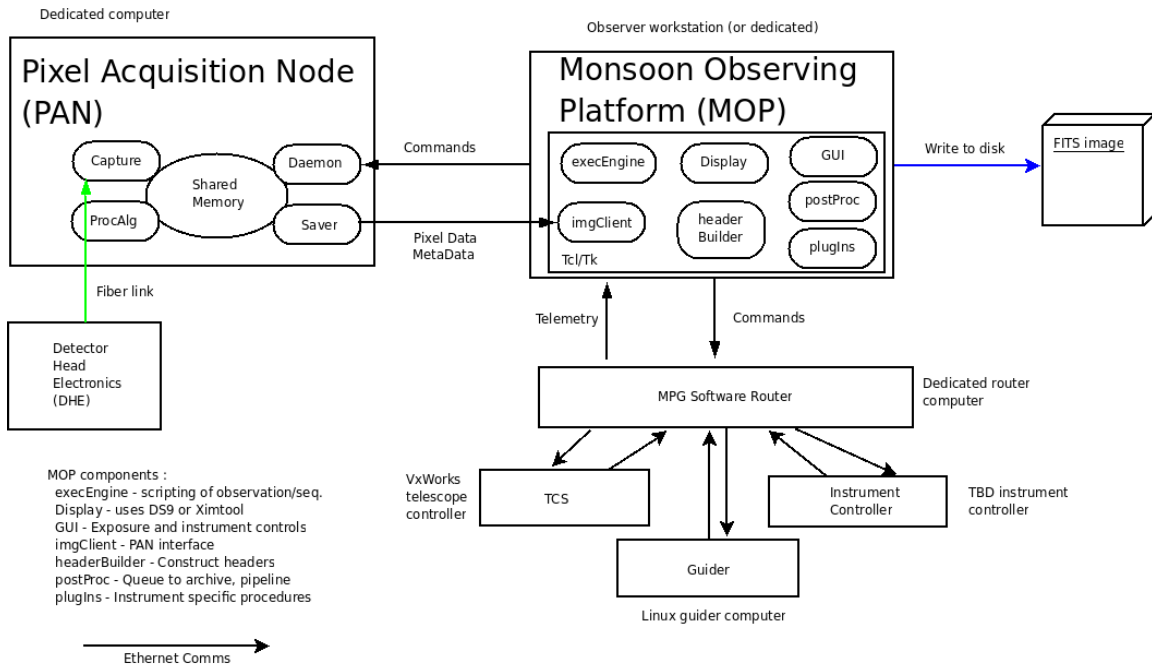


Figure 4 – MOP block diagram.

The Bench Spectrograph implementation only runs the detector, via Monsoon, but the WHIRC and 2-m versions do interact with the instrument software and also provide a limited scripting capability.

Thus the MOP already has the basic functionality to run the KOSMOS detectors. However, it currently does not deal with the NOAO data handling system, and the scope of this effort needs to be investigated. What it would also need to add is the functionality to communicate with the KOSMOS spectrograph hardware – probably through the IE server shown in Fig. 1 rather than replacing it – and to talk with the 4-m TCS instead of WIYN or the 2-m.

The MOP already has the ability to do simple scripts, but the ability to interact with images as part of the scripts would be a new capability.

The interface is already familiar to mountain staff, and probably to more potential KOSMOS users than the NOCS. In any case, like the OSMOS GUI, it is simple and should be easy to learn.

Discussion

Of the three alternatives outlined above, any one would clearly be satisfactory, so the choice between them is partly a matter of taste as well as an assessment of which one involves the least effort.

Discussions with the people who would do the work strongly suggest that the level of effort to implement the different solutions is roughly similar, roughly a few person-months; more accurate estimates would require carrying out a design study for each option, with some sort of “scored” down-select. Given the overall scope of the project and of the user software work, this would be excessive.

Two considerations lead us to choose the NOCS:

- Ideally, there would be a single NOAO standard for user software, but unfortunately there are several, taking both KPNO and CTIO into account. Both the MOP and the NOCS are already used on multiple NOAO instruments, which would make them preferable to adding the OSMOS software to our suite of pseudo-standards. The NOCS has the advantage that it will be familiar to both KPNO and CTIO staff, which is a consideration if COSMOS (the CTIO copy of KOSMOS) is funded.
- Current man-power allocations are better matched to implementing the NOCS.

Issues

Efficient use of KOSMOS involves procedures where the user interacts with an image to calculate positional offsets of the telescope and adjustment to the guider. This requires that the software be able to call IRAF (or IDL) routines and accept arguments from those routines. If this is not possible, then alternative software should be considered.

The most complicated such routine is the full mask alignment process, which determines precise XY offsets and rotations; processes to do this have been developed for other telescopes (including OSMOS at MDM, presumably, as well as MARS on the 4-m and many others). The ability to import or adapt such a routine would influence choices but should not be definitive by itself.

Operation of the KOSMOS detectors in stand-alone mode on other spectrographs should be examined as part of the development effort. However, the actual development of any additional software or software interfaces is *not* part of the overall KOSMOS project; the idea is to come up with a rough estimate of the scope of such additional development, and nothing more.

Versions

Version	Date	Changes
1	Mar. 22, 2010	First draft, initial release
2	Mar. 22, 2010	Figure 2 added
3	April 15, 2010	Additional material on NOCS added; discussion on selection revised; approved for release.