Coping with data deluge – a data system for the Megacam

M. A. Conroy, J. B. Roll, Jr., W. F. Wyatt, D. J. Mink and B. A. McLeod

Harvard-Smithsonian Center for Astrophysics, 60 Garden St, Cambridge, MA 02138, USA

ABSTRACT

Megacam is a wide-field optical imager for the converted MMT that uses thirty-six 2048 by 4608 pixel CCDs to cover a 24′× 24′ format. We describe a computer architecture designed to accommodate the expected data volume and show benchmark results from prototype implementations that demonstrate the performance attained by each of the design decisions. We show that our time budget allotments can be met using a modular, scalable architecture design that exploits the natural parallelization of multiple, identical detector components.

Keywords: MMT, pipeline, computer architecture, mosaic, visualization

1. INTRODUCTION

At the converted Multiple Mirror Telescope we are developing several new CCD-based instruments to exploit the wide-field capability: Hectospec,[1] Hectochelle,[2] Binospec,[3] and Minicam and Megacam.[4] These instruments will share a common data acquisition and analysis system. Of these, Megacam[4] has the largest data volume and thus drives most of the hardware and software requirements.

Although CCD data analysis techniques are mature and relatively straightforward, the projected data volume from Megacam, 100 Gigabytes (Gbytes) of raw data per night, requires qualitatively new approaches to visualization, data quality monitoring, quick-look analysis and final data reduction pipelines. This data volume presents issues at the telescope as well as at data reduction time. Real-time, near real-time, batch and interactive modes, many of which perform comparable functions, must be developed. Our goal is to design a system that can be tuned to run efficiently in all these environments and modes.

The Megacam instrument acquisition system must read out, save to disk, and display an image from the 18K × 18K pixel array in 30 seconds. After each CCD frame is taken, an automated data reduction pipeline is invoked to reduce the data and provide status and housekeeping feedback to the observer. Interactive analysis tools will provide imaging and accurate sky coordinates, so that catalog objects may be overlaid. Standard quick-look analysis functions including image centroiding, data slices, radial profiles, contouring and statistics will also be provided.

We will address three components of the data volume problem: data acquisition, automated data reduction pipelines and interactive analysis tools. Both data acquistion and automated reductions are time critical. Here, we will define the throughput requirements, discuss the hardware and software design and conclude with the current and projected benchmarks based on our existing prototype implementations.

2. DATA ACQUISITION/OBSERVATORY SYSTEM

2.1. Throughput

Megacam (and all the next generation CCD imagers) will be composed of mosaics of CCD chips each with one or more amplifier read-outs. Megacam uses thirty-six 2048 by 4608 pixel CCDs to cover a 24′ × 24′ format. This results in an 18K × 18K detector array. Since each chip has two amplifiers, each integration produces a set of 72 essentially identical raw data images in FITS image format. Each of these components is approximately 19 Mbytes in size, producing an assembled archival FITS file of 0.68 Gbytes per integration. (For these calculations, we are ignoring the additional size contributed by the programmable overscan and underscan regions: these regions will contribute approximately 10 Mbytes to the total size). Often, each observed field will be composed of between three and five dithered observations to minimize the effects of cosmic-ray contamination and to fill in gaps caused by chip boundaries and bad CCD columns.

Other author information: (Send correspondence to mconroy) email addresses: mconroy,jroll,wwyatt,dmink,bmcleod@cfa.harvard.edu

Table 1. Time profile for typical observing night at full resolution

exposure type	exposure time	readout time	elapsed time	raw size	processed size	mosaiced size
20 flat fields	20×5 min	20×0.5 min	110 minutes	13.6 Gbytes	27.2 Gbytes	-
20 bias fields	20×0 min	20×0.5 min	10 minutes	13.6 Gbytes	27.2 Gbytes	-
10 sky fields	10×5 min	10×0.5 min	55 minutes	6.8 Gbytes	13.6 Gbytes	-
90 objects	90×5 min	90×0.5 min	495 minutes	61.2 Gbytes	122.4 Gbytes	50 Gbytes
Total			670 minutes	100 Gbytes	190 Gbytes	50 Gbytes

A typical observing night (see Table 1) might include 50 calibration images and 90 science images. Assuming 15 minutes per field divided between 3 5-minute exposures, this gives a total of 670 minutes or about 11 hours of data and produces an overall observing data volume of approximately 100 Gbytes of raw data when using the camera at full resolution. For many applications a binning factor of two or three will be common. Assuming a binning factor of three, the total data volume will be reduced by a factor of about 10, while also reducing the chip read-out time by a factor of 10, resulting in three second read-out times for the full instrument while still providing a pixel scale of $0\farcs25$.

2.2. Computer Architecture

This vast quantity of raw data cannot be handled reasonably by the computer/network/tapedrive infrastructure currently in place. However, the high degree of parallelization among the data components lends itself well to a distributed system of multiple workstations, CPUs, and disks.

Current system architectures heavily penalize data reads and writes that must be done across the network, since the prevalent network speed is only about 1 Mbyte s^{-1} . Thus, network access speed and band-width are an important area for upgrade, both at the telescope site and at the observers' home desk-top. Megacam plans to use 100 Megabit, and later possibly Gigabit, ethernet at the MMT. At the Center for Astrophysics (CfA), 100 Megabit desk-top fiber connections are already in place. This network upgrade increases disk access speed by a factor of 10-100, and eliminates the requirement of having all data on local disks, thus allowing more flexible architectures.

An improvement to 100 Megabit ethernet means that data access speeds will now be limited by the disk rather than the network. It now becomes important to implement a data distribution plan that optimizes the disk performance. While it is possible to achieve high disk performance by writing to raw disk partitions, this has the disadvantage of producing files not accessible by a *UNIX* file structure. *RAID* hardware to implement a disk striping architecture is also available but is still quite expensive.

Four Mbyte s^{-1} is the nominal performance achieved by writing a single multi-extension FITS file to a single local disk file with blocking I/O. We can achieve most of the performance of *RAID* disk striping using conventional file storage when the data are distributed so that each disk contains data that are accessed asynchronously in large sequential blocks. The multiple-amplifier CCD data are naturally optimized for this approach when the data are organized on disk according to amplifier rather than by time or observation. In the Megacam scenario this results in 72 FITS images distributed across 16 disks. The disks must be assigned to SCSI ports in a way that allows each of them to achieve the maximum 10 Mbyte s^{-1} bandwidth: 2 disks per FastWideSCSI chain or 4 disks per UltraSCSI chain. This can be viewed as producing a factor of two to four increase in writing speed per disk, since 2-4 disks can now be written simultaneously. When this is coupled with another factor of two increase by using asynchronous I/O rather than blocking I/O, we are able to achieve an overall increase in speed to disk of about 8, thus attaining the required sustained speeds of 29 Mbyte s^{-1} .

The final component of the data acquisition system is the user archive of the raw data. The archive routine will assemble a single multi-extension FITS file from the individual component FITS images, if requested. Thus the archive has the convenience of a single tape file per observation, while the data acquisition system takes full advantage of data distribution and parallel processing. The archive will require some type of magnetic storage medium. Current

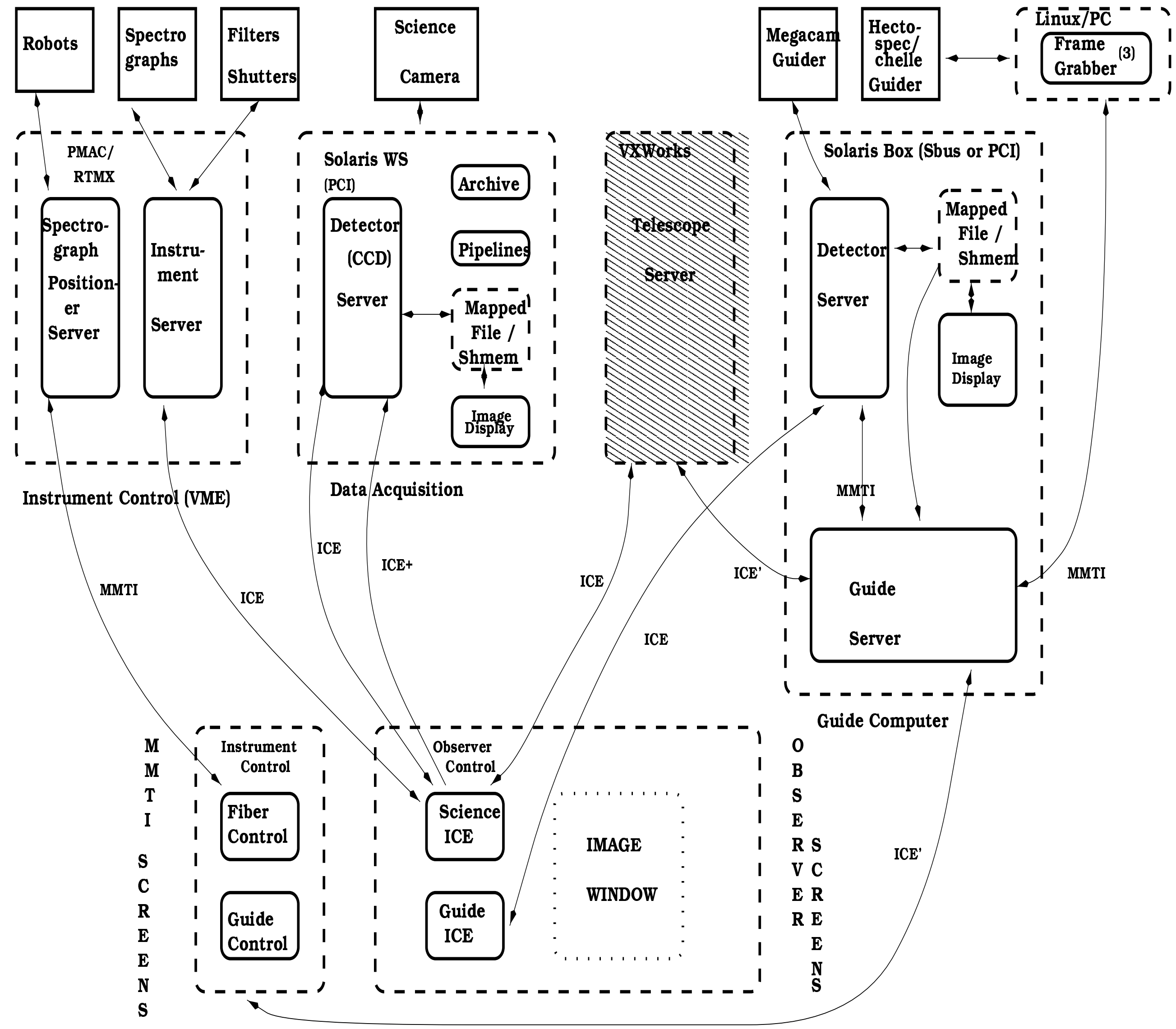

Figure 1.
Architecture for MMT Observatory Data System. Components that conform to the standard *ICE* implementations are labelled using the *ICE* protocol. *ICE+* denotes an ICE implementation where we've made small additions to the standard messages to accomodate multiamplifier chips and raw data that will remain on the server disks. *ICE′* denotes the components that will communicate with the ICE protocol but the messages differ significantly from standard ICE. Components that fall outside of current ICE implementations are labeled using the *mmti* message bus system.

Digital Linear Tape (DLT) 7000 technology achieves about 8 Mbyte s^{-1} , so 4 hours would be required to backup with this technology, and more than one cartridge is required. We will delay selection of the appropriate technology and we will evaluate the promised Advanced Intelligent Tape (AIT) technology.

2.3. Software Architecture

The control system has been designed to use a modified *IRAF Control Environment (ICE)* architecture.[5] The choice of *ICE* is motivated by its wide acceptance as a standard and the quick development time for a new detector implementation. We are following future developments of *ICE* as well as other data acquisition protocols, and will upgrade our interfaces as appropriate. We will supplement the *ICE* system with additional components to support

MMT instrument capabilities not currently included in *ICE*, most notably active focusing and guiding and computer control of the spectrographs, Hectospec and Hectochelle, fiber placement. The supplemental systems will use the MMTI *ascii* messaging protocol over sockets, designated *mmti* in Figure 1.

The Megacam instrument poses the most demanding performance challenge of the MMT instruments. The Megacam specification requires that the 18K × 18K array be read-out in 24 seconds and stored to disk in a maximum of 30 seconds. This puts serious constraints on both the CCD hardware controller, the detector server data acquisition software and the disks. The server and disk performance have been discussed above. The CCD controller design is presented elsewhere in this volume.[6]

Filter wheel, shutter, telescope and detector control, and data acquisition functions will be implemented with standard *ICE* servers. The Megacam instrument will also include an active guiding and focusing mechanism. This capability is outside the normal *ICE* functions. The Megacam guiding system is composed of 2 CCD chips which will be operated as a mini-ICE detector sub-system for control and data acquisition. While we will maintain the capability of operating these devices in a conventional *science* mode, for routine guiding they will be free-running and they will be programmed with 2 independent windows determined by the location of the selected guide stars. For these reasons additional client tasks must be written to control this operation and different server functions are required to capture data in free-running mode and to de-multiplex the multiple windows. The data acquired by the *ICE* guide server from this system are analyzed by a guide server process which derives positioning and focusing correction values for the telescope. These corrections will be communicated to the telescope server using a messaging system we've designated *ICE'*. This messaging system uses the *ICE* protocol but involves some extensions and modifications to the existing message definitions to allow communication of these additional values and because it may well require that the telescope server support multiple, simultaneous client connections, which is a feature not currently supported by *ICE*.

An *ICE* implementation of the CCD detector server has been completed. The baseline *ICE* client and server have been modified to support the multi-chip detector and to allow data storage on the server machine (rather than the client machine). We have designated this modified system *ICE+* in Figure 1. The CCD data acquisition system uses an EDT SDV-FOI interface connected via fiber optic cable. The EDT-provided library contains a standard remote camera interface implemented with off-the-shelf hardware as well as routines to read the CCD data via DMA directly into ring buffers in the detector server memory and manages the buffer lock-downs and multiple threads. The data from the multi-amplifier CCD detectors is multiplexed such that the pixels from each amplifier are interleaved. with the corresponding pixel from all the additional amplifiers in use. The detector server acquires data in 0.5 Mbytes DMA buffers. These data are demultiplexed by multiple threads, each of which cycles through the buffer to extract one amplifier's data, stores it in shared memory and executes an asynchronous write to disk. The DMA buffer size is a tunable parameter via *ICE* and we expect that this optimal buffer size may change for different detector sizes and configurations. Our prototypes using memory-mapped files were not successful; we achieved performance figures 50% slower than the shared memory implementation. Apparently, the asynchronous writing allows the detector server more direct control over the I/O and thus allows us better tuning options.

Real-time image display is a fundamental requirement for any observatory system. The Megacam detector server writes the complete detector read-out to shared memory, including geometry and world-coordinate keywords. The image display program SAOtng and its successor, DS9,[7] can display data from shared memory files and also support cursor-tracking read-out of world-coordinates (e.g. RA/Dec) as well as pixel coordinates. An extensible analysis menu accesses most of the quick-look analysis algorithms currently available in the NOAO IRAF **imexamine** task. We will discuss the features of image display in more detail in section 4. The special feature required for real-time imaging is the ability to request frequent updates of the display from the shared memory amplifier images.

2.4. Benchmarks

We currently have two systems in-house which we have used for prototyping: a Sun Ultra 2/2170 SBus based system and an Ultra 30/300 PCIBus based system (Figure 2). The Ultra 30/300 system will be used soon to evaluate the PCIBus version of the EDT DMA interface. Up to the present it has been used for pipeline reductions benchmarks. Our current Ultra 2/2170 machine with two 167-MHz CPUs, two Fast-Wide SCSI chains, four 10 Mbyte s^{-1} disks and 128 Mbytes of RAM memory currently can achieve sustained performance of 20 Mbyte s^{-1} when running the detector server to capture DMA data from the CCD controller and write the multiple FITS primary images to disk. This performance is limited by overall computer BUS speed/bandwidth, rather than disk performance, since we can

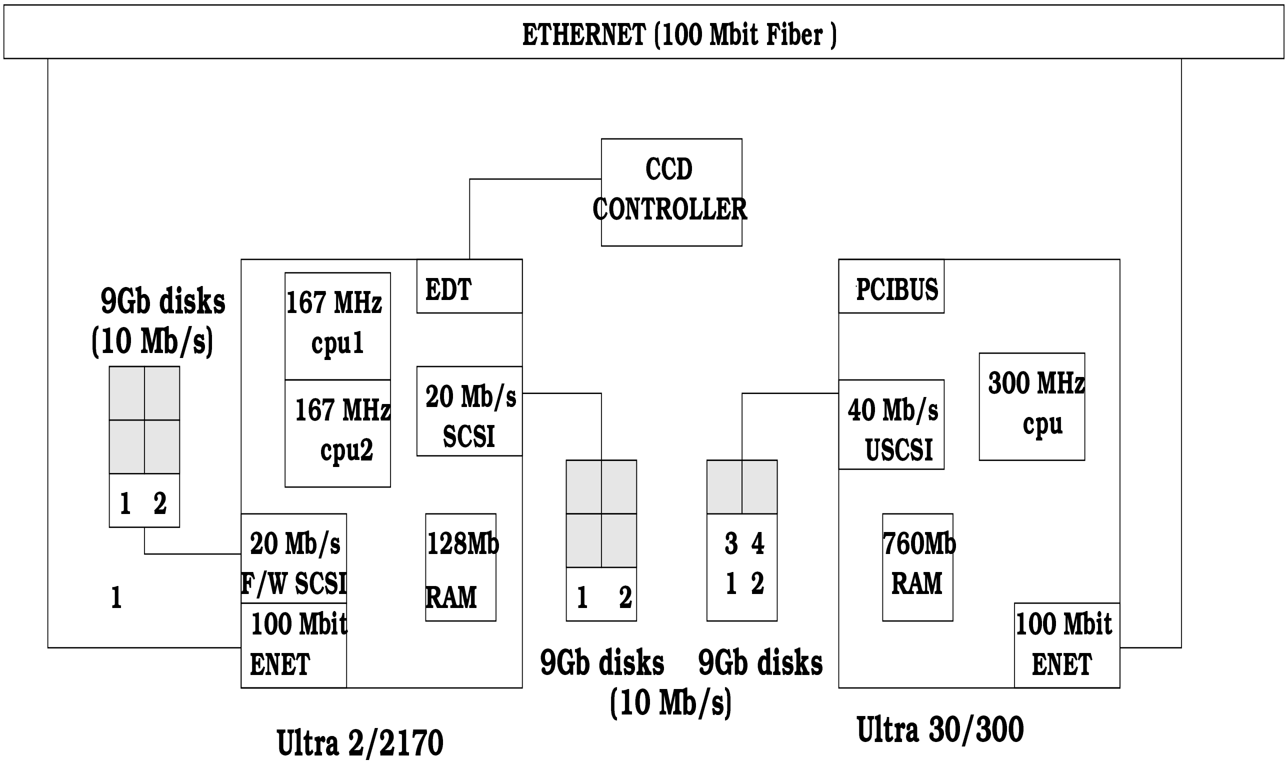

Figure 2. Existing computer configurations for Minicam for benchmarking both data acquisition and pipeline processing

Table 2. Benchmark hardware speeds for the current Ultra 2/2170 prototype as well as projected speeds for future Ultra 30/60 and Ultra 450 systems

Component	Ultra 2170		Ultra 30/60-300		Ultra 450-500	
	Model	Qty	Model	Qty	Model	Qty
CPU	167 MHz	2	300 MHz	1/2	500 MHz	4-8
SCSI	20 Mbyte s^{-1} (Fast-Wide)	2	40 Mbyte s^{-1} (Ultra)	2-4	80 Mbyte s^{-1} (Ultra2)	4-8
network	100 Mbit/s (Fast)	1	100 Mbit/s	1	1Gbit/s (Gigabit)	1
disk	10 Mbyte s^{-1}	4	10 Mbyte s^{-1}	4-16	10 Mbyte s^{-1}	20+
tape	2.5 Mbyte s^{-1} (DLT 4000)	1	2.5 Mbyte s^{-1} (DLT 4000)	1-2	8 Mbyte s^{-1} (DLT 7000)	2-4

already demonstrate sustained write speeds of 40 Mbyte s^{-1} to disk when not running the DMA data acquisition. Because the EDT/SDV SBus-based system cannot achieve the full 29 Mbyte s^{-1} speed that we require, we will most likely replace the Ultra 2/2170 with an Ultra 60/2300 (the dual CPU version of the existing Ultra 30.) We expect that with an Ultra 60 (or faster) workstation we can easily achieve the necessary 29 Mbyte s^{-1} speed, as the CPUs will be at least twice as fast as the Ultra 2/2170 and the PCIBus will be about three times faster than the SBus. We have already demonstrated the full 29 Mbyte s^{-1} output with the existing disks and SCSI's on the Ultra 30 system.

We expect that rapid technology advances will improve the performance numbers that we currently achieve. It seems reasonable to expect a factor of 4 improvement in the next 2 years for both processor speed and data throughput speed (ethernet, disk, bandwidths and speeds). These system component ratings and projections are summarized in Table 2.

3. DATA REDUCTION PIPELINE SYSTEM

Pushing the 100 Gbytes of raw data per night through automated processing pipelines is the second component of the data deluge problem. The goal is to complete the CCD calibrations, mosaicing of the multiple exposures, source extraction and catalog generation at the telescope site. We plan to send the observer home with catalogs of extracted data objects that can serve as the starting point of scientific analysis. We must provide sufficient disk and processing capacity to support these reductions as well as the necessary software infrastructure. This goal aside, we realize that we must still provide a means for users to redo data reductions at their home institutions where the computer configuration may be different than the highly optimized system provided at the MMT. This constraint is an important motivation for ensuring a scalable system architecture. As above we will define the throughput requirements for the reduction pipelines, describe our hardware and software architecture design and conclude with the benchmark results from our current prototypes as well as the projected preformance with newer technology.

3.1. Throughput

The starting point of the reduction pipeline is the raw data. At the telescope the raw data will be stored on the disks served by the data acquisition computer. At the observer's home institution the data are likely to be on tape. In this case the de-archiving program will either unload the tape files into the designated directory, or it will split and distribute each multi-extension FITS file to specified directories as it is de-archived. Just as the individual chip data were distributed over several disks at the telescope, this is also the optimal data organiziation for efficient pipeline processing. However, the more traditional option of loading and storing data according to time or observation is still available.

Each processed observation will produce another 72 FITS images of size 38 Mbytes each or an assembled archival FITS file of 1.36 Gbytes. An initial 100 Gbytes of raw data thus produces 136 Gbytes of processed data plus about 70 Gbytes of calibration files. One or more processes can be run simultaneously to do the CCD reductions of the data. CCD reductions have two components. First, the multiple calibration files must be reduced and combined to produce single CCD correction files. Next, these correction files must be applied to the object files (probably producing 32-bit real images rather than 16-bit integer images). The optimal configuration would be for each CPU to be assigned a set of amplifiers (and therefore disks) that it will process. These processes are completely parallel and independent. The results of this processing are written back to disk (and optionally to tape) in a way completely analogous to the raw data (section 2). Because of the large amounts of I/O the CPU can easily process 2-8 (depending on CPU speed) files concurrently to overlay the I/O optimally thus allowing the maximum CPU loading.

Often, each target field will be composed of 3-5 dithered observations to minimize the effects of cosmic-ray contamination and to fill in gaps caused by chip boundaries and bad CCD columns. Once the overlapping fields are combined there will be a single, final, regridded mosaic image of approximate size 1.5 Gbytes. The above 136 Gbytes of processed data will thus produce about 50 Gbytes of mosaiced data. To build the final mosaic requires determining very accurate coordinates for each of the dithered exposures. A final astrometry solution for each amplifier is determined based on existing calibrations and is stored as a WCS in each amplifier file. Each of the dithered exposures is then regridded to a common coordinate grid so that the mosaicing task then assembles the components based on this coordinate information. The mosaic assembly also generates exposure map and background maps calibrations as auxiliary information to the mosaiced field.

The final step of the processing is to extract sources with accurate fluxes and positions. The output source catalogs are tiny in comparison to the actual data fields and will be stored in either FITS or ASCII tables compatible with the Starbase database system.[8]

3.2. Computer Architecture

At the observatory site we will provide a system both for the data acquisition and for the data reductions and analysis. For the initial instruments we expect to use a 2 workstation system, where one workstation is dedicated to data acquisition and real-time display while the second workstation is available for pipeline data reduction processing. For the final configuration to support the full 36-chip Megacam, it may be reasonable to replace these 2 workstations with a single high-end server class machine that has sufficient CPU and bus bandwidth, thus minimizing the need for high speed / bandwidth connections between the workstations. For the purposes of this discussion we will examine the options for the 2-3 work-station configuration (see Figure 3). The configuration of the processing machine(s) is expected to be 2-4 CPUS, at least 20 9-Gigabyte disks and at least 1 tape archive device. This system would need to

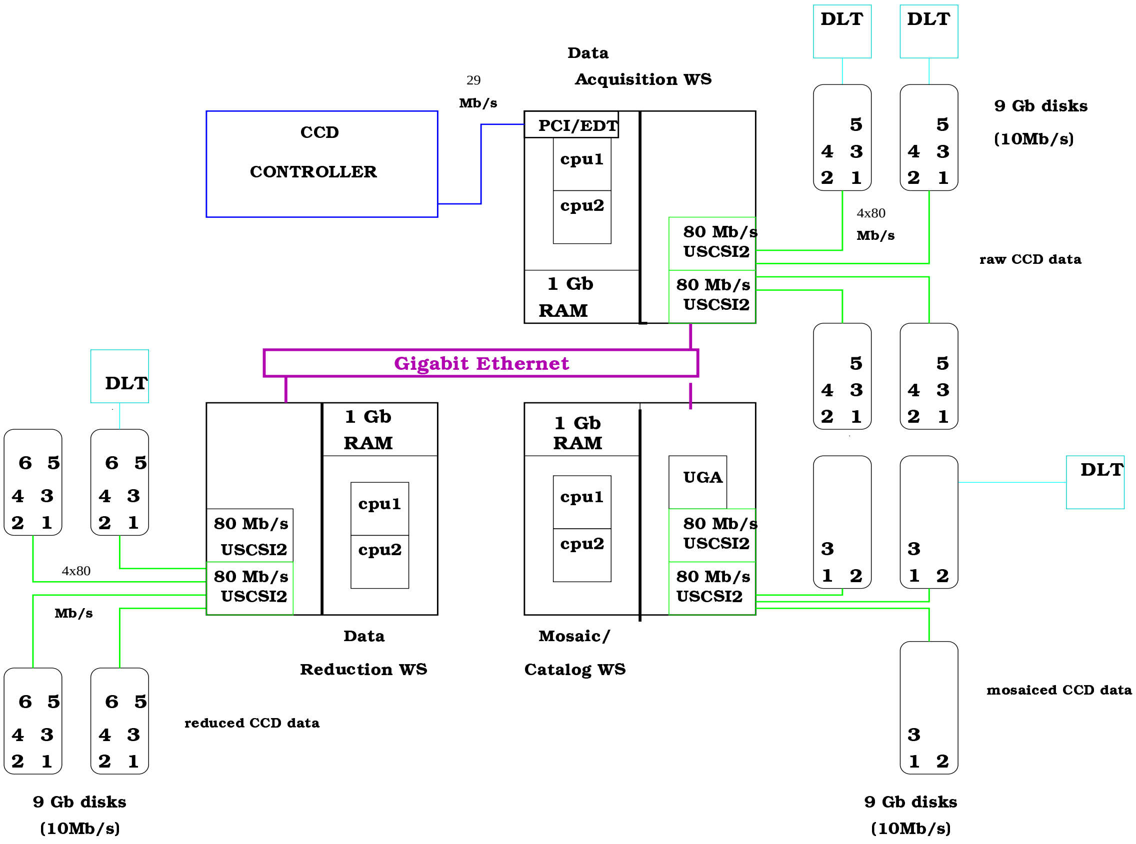

Figure 3. Example of a distributed system for doing the Megacam data acquisition and pipeline reductions. This is an architecture that can be built up in increments as more components of the system come on-line. It shows 3 Ultra 60/2 workstations each serving the data from one stage of the acquisition or processing scenario.

be connected with at least a 200-400 Mbit ethernet switch or Gigabit ethernet to the data acquisition work station, where the raw data is stored. This will allow efficient access of all the raw data from the data acquisition computer and sufficient processing power to perform both CCD reductions mosaicing and catalog generation.

3.3. Software Architecture

Software development, maintenance and complexity are another area where optimization can yield savings. Our software goals are to manage complexity while minimizing development effort and providing a modular, flexible software system. We have designed a system that minimizes platform dependencies and maximizes options for software re-use by requiring these criteria for the software components:

- Parameter-driven Toolkit
- *POSIX* shells for scripting tools together
  e.g. Korn shell, Tcl
- *ASCII* or FITS formats for all data files
- *Tcl/Tk* architecture independent widget set
- *POSIX* components for interprocess communication, *e.g. pipes, sockets, mapped files*
- *ANSI* C/C++ for programming language

3.3.1. UnixIRAF

NOAO's *IRAF* system is a rich source of existing astronomical analysis and reduction routines. We rely heavily on existing software for a large number of processing algorithms. In particular, we are following the CCD FITS header keyword conventions proposed by NOAO and currently implemented in their Mosaic Data Handling System presented elsewhere in this volume.[9] By following these keyword conventions we then have access to all the data reduction and analysis tools available in IRAF for CCD reductions and calibrations, source extractions, astrometry solutions and image combination. In particular, the MSCRED package supports multi-extension FITS files and mosaiced CCD data reductions. We have successfully used the MSCRED package to reduce data from both the Big Throughput Camera (BTC)[10] and the Mt. Hopkins 4-Shooter[11] multi-chip CCD instruments.

These tools are designed to run within the *IRAF Command Line* (CL) environment, but this environment does not provide a *POSIX* scripting language, nor does it support *InterProcess Communication (IPC)* between separate tasks. This means it is not possible to combine these *IRAF* analysis routines with other non-IRAF components and support *IPC* between them. To use these tools within the Megacam *POSIX* environment it is necessary to make them available from command-line shells rather than the *IRAF* CL. *UnixIRAF* is an SAO project which prototypes many of the planned OpenIRAF features. The motivation for *UnixIRAF* is to allow use of the large number of existing IRAF tasks within a standard Unix environment, e.g. C-shell, K-shell or Tcl windows or scripts, thus allowing them to communicate with external imaging or plotting packages. A prototype pipeline has been developed to do BTC, NOAO Mosaic and 4-Shooter CCD reductions.

We are using the SAO IRAF-compatible parameter interface library to emulate the *IRAF* parameter interface when running from Unix.[12] The *IRAF* CL parameter interface system is very familiar to the astronomy community and very powerful. In addition to all the standard *IRAF* features, we've provided a few extended features to facilitate both pipeline and interactive processing. Pipeline processing requires the ability to archive a complete, as-run, set of parameter files for the entire pipeline. Our extensions allow each invocation of a tool to have its own version of the parameter file, so that no parameter values are overwritten during the course of a pipeline. We also support features that allow pipeline parameters to be configured from database entries based on current the current environment definition of detector, instrument and binning parameters. The dynamic parameters allow existing tools to be built into *GUI* driven applications where both the data and the parameter settings are queried dynamically from the image display, overlay and cursor position.

3.3.2. Applications

The current pipeline prototypes have been implemented in Korn shell, invoking UnixIRAF tools. They declare unique parameter files for each tool invocation and maintain a database of the optimized processing parameters for each instrument, filter and binning factor. For full mosaic processing the co-shell features of Korn shell will enable sub-pipelines for each amplifier to be executed concurrently and then re-synchronized before starting up the next stage of processing.

We are using *Tcl/Tk* as the system for building all graphical user interface systems. There are several reasons for this choice. *Tcl/Tk* is a platform independent package that is not dependent on a Unix platform or even an X-windows server, even though this is what we use as a development platform. This architectural choice should facilitate the migration of analysis components to new architectures in the future. The use of the Tcl scripting language allows us to bind together existing non-interactive tool components, such as *IRAF* tasks, Tk-widgets and imaging applications such as SAOtng (discussed in section 2.3) to produce a layered *GUI*-driven application that uses exactly the same underlying algorithms and tools available in the pipeline.

3.4. Benchmarks

We have used both Big Throughput Camera and the Mt. Hopkins 4-shooter mosaiced CCD detector data to benchmark the system performance. The processing times for the current prototype computer system and projections for future systems are summarized in Tables 3 and 4. Both BTC and 4-shooter CCD reductions have been processed using the Ultra 2/2170 machine and achieve an overall performance of 1 minute per integration has been achieved. This encompasses the steps of calibration field generation and CCD reductions, where the calibrations include generation of a SuperFlat from all the object targets for a given night.

The BTC data is a factor of 20 smaller than Megacam, so we extrapolate that Megacam would require 20 minutes per field on our current machine. However projecting the scenario forward to the future target machine with an

Table 3. Benchmarks and projections for pipeline components of single exposure at full resolution

operation tested	current BTC Ultra 2 time	derived Megacam Ultra 2 time	projected Megacam Ultra 60 time	projected Megacam Ultra 450 time
ccd reduction	1 min	20 min	5 min	1.25 min
mosaic	1 min	20 min	5 min	1.25 min
source extraction	1.5 min	30 min	7.5 min	2 min

Table 4. Time profile for complete processing of typical observing night at full resolution

elapsed type	exposure exposure time	Ultra 60 reduction time	Ultra 60 mosaic time	Ultra 60 source time	Ultra 450 reduction time	Ultra 450 mosaic time	Ultra 450 source time
50 calibration fields	175 min	250 min	250 min	-	62 min	62 min	-
90 object fields	495 min	450 min	450 min	-	112 min	112 min	-
30 mosaiced fields	-	-	-	225 min	-	-	56 min
Total	670	700 min	700 min	225 min	174 min	174 min	56 min

increase of 4 in CPU speed and perhaps an increase of 2 in number of CPUs, this number reduces to 1.25 minutes per field. Again much of this speed is achieved by maximizing the file I/O across independent disks. Thus each processing thread is reading from only one disk and writing to only one disk. Similarly, each of the observation will be regridded when the astrometric corrections are applied. These operations now take 1 minute on the current system, so we will apply the same projection to the future systems and allocate 1.25 minutes for this operation. After the regridding, the images are combined using a simple stacking operation that is almost completely I/O bound. This requires another 10 seconds. but produces only 1/3 - 1/5 of the number of output files. The source extractions have been run using the SExtractor software[13] and also require just over 1 minute per BTC field. Using the same extrapolations, this will require 3 minutes per field on Megacam.

Thinking ahead to the full Megacam processing load leads to two possible scenarios. Current projections indicate that the processing load could be handled by perhaps 2 high-end workstations connected to an existing data acquisition workstation, or the entire data acquisition and processing system could be replaced by a single very high-end processor, an Ultra 450 server-class system with at least 4 500 MHz CPUs, multiple internal buses and large amounts of RAM memory. However, these numbers have been calculated for the heaviest possible load, that of an entire observing night of full resolution Megacam images. In the case of binning by 3, then most of these numbers reduce immediately by almost a factor of 10. In this scenario, an existing Ultra 60/2300 would easily handle the processing load in less than 3 hours.

4. INTERACTIVE APPLICATIONS

The complexity of the mosaiced data will require additional visualization tools to enable users to navigate the data easily. The data will often be stored in individual amplifier FITS files distributed among disks. It will be important to have a suite of simple tools that hide this storage mechanism from the user and allows the data files to be navigated as if a single entity. At the telescope the *ICE* directory which would normally contain the CCD data will instead contain directories with names of the form <*imagename*>.mos. This directory file serves as a new *mosaic* data type which refers to the entire multi-amplifier image. Each of these directories will contain links to all the components files for this observation, located on several disks. Our tools accept filename templates as well as multi-extension

FITS filenames as input specifications. For imaging purposes, NOAO has defined header keywords that provide the information on how the individual image components piece together in the field of view.

4.1. Navigation Tool

The optimal data organization for storage and processing efficiency is not consistent with the most natural user access modes. Therefore, we provide a simple set of viewing utilities that mask the details of the physical storage plan and allow easy user user access to observation units. The *XDIR* facility, which is distributed with SAOtng, has the ability to display virtual directories and allows mouse clicks to be linked to tasks, such as display tools. We intend to customize this utility to allow browsing of an observation data set, where the observation data set may be stored in two or three different forms. For raw data storage we will write single FITS data files for each integration from each CCD component, e.g. single FITS file for each CCD amplifier.

Utilities have been developed that collect all the components for a single observation into a multi-extension FITS file for archiving to tape. Similarly, downloading the tape goes through a utility to re-split the data across multiple disks. The configuration of disks and disk controllers will be optimized for the CCD processing, so that each disk contains all the data for a particular CCD amplifier. This scheme organizes the raw data completely by amplifiers. Another process would collect and write the 72 FITS files to a multiextension FITS file and store on a FITS tape (or provide to the specified archiver). There are many instances where the observation must be physically moved from one location (and/or medium) to another. The gluer/splitter tools will support this.

4.2. Visualization Tools

We intend to use the SAOtng/DS9 program (as described in section 2.3) for image display. This facility provides full support for multiple coordinate system read-out, world (sky) coordinate readouts (WCS), configuration of frame buffers, multiple scaling algorithms and a large number of public access points that enable user-configuration of data fed into the tool. SAOtng allows user-defined file types that can be accessed and formatted with user-provided drivers. We have written drivers to support our new *mosaic* data type. Thus with a single command, the user can load all of the sub-images into a single display buffer. SAOtng recognises the boundaries between the individual amplifiers and accesses the amplifier specific WCS for each component. As the user moves the cursor across the image display, SAOtng switches WCS contexts at each of the amplifier boundaries. This enables a very accurate cursor position readout from the real-time display even without the expensive mosaicing operations to build a single image from the individual amplifier components. With the DS9 system, support for the mosaic file type will be built-in rather than a user-extension.

It is a difficult problem to view the entire CCD field and still achieve sufficient accuracy for meaningful statistics. To allow fast zooming to regions of interest, the Megacam data acquisition system will pre-compute a handful of binned images, especilly those zoom factors that span several amplifier files. When a pre-computed zoom factor is requested the appropriate file can be loaded either from shared memory or quickly from an existing disk file. For small zoom factors with only a few actual pixels per display pixel, it should be possible to generate these images on-the-fly.

SAOtng also has a powerful mechanism for defining regions-of-interest. It's possible to overlay circles from existing catalogs onto the image, or for the user to manually draw a region-of-interest. Using SAOtng's *XPA* messaging system, analysis tools can then be invoked on the designated region. *IRAF* is in the process of adding region (plmask) support for many of the *IRAF* and NOAO image tasks which will allow these tasks to be driven via the image display selections.

4.3. GUI Application

Tcl/Tk allows the imaging tool (*SAOtng* or *DS9*) and the processing algorithms to be combined into *GUI* driven applications. We are currently working on providing a *GUI*-driven front end to the CCD data acquisition system, that drives the *ICE* clients from the GUI and combines these functions with the image display and quick-look analysis routines from DS9. A second application in development is a catalog preview tool that allows display of the CCD image, overlay of a source catalog and editing and refining of the catalog based on cursor clicks on the image display.

5. CONCLUSIONS

We have shown that existing technology is sufficient to achieve the data throughput required of Megacam. However, the organization both of the data and of the computer hardware can make dramatic differences in the achieved performance. It will continue to be important to exploit the parallel, distributed properties of the data to ensure users the ability to process data efficiently with existing computer components. Although many users may not have access to the high-end components provided at the MMT, the same data design concepts can be applied to any local configuration to produce substantial throughput improvements. Furthermore, these throughput optimizations are achieved without requiring any platform or operating system dependent customizations. Rather the software for reduction and analysis is constructed with standard *POSIX* components, standard data formats and high-level *GUI* components that have very few platform dependencies. This is a first step in preparing for a future with different platform options.

ACKNOWLEDGMENTS

We wish to think the SAO Research and Development Group (SAORD)[14] for their support in implementing mosaic imaging functions in the current SAOtng release and the under-development DS9 application.

REFERENCES

1. D. G. Fabricant, E. H. Hertz, A. H. Szentgyorgyi, R. G. Fata, J. B. Roll, and J. Zajac, "Construction of the Hectospec: a 300 optical fiber-fed spectrograph for the converted MMT," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
2. A. H. Szentgyorgyi, L. W. Hartmann, P. N. Cheimets, D. G. Fabricant, M. R. Pieri, and Y. Zhou, "Hectochelle: a multi-object echelle spectrograph for the converted MMT," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
3. D. G. Fabricant, R. G. Fata, and H. W. Epps, "Binospec: a dual-beam, wide-field optical spectrograph for the converted MMT," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
4. B. A. McLeod, J. C. Geary, T. M. Gauron, M. P. Ordway, and J. B. Roll, "Megacam: paving the focal plane of the MMT with silicon," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
5. S. Schaller, "IRAF data acquisition software," *Astronomical Data Analysis Software and Systems I* **25**, pp. 482–483, 1992.
6. J. C. Geary and S. M. Amato, "Camera electronics for the 72-channel SAO megacam," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
7. E. G. Mandel, "Applying public access programming techniques to astronomical image display," in *Visual Data Exploration and Analysis III*, *SPIE Proceedings Series* **2656**, pp. 214–225, 1996.
8. J. B. Roll, "Starbase: A user centered database for astronomy," *Astronomical Data Analysis and Software Systems V* **101**, pp. 536–539, 1996.
9. F. Valdes and D. Tody, "NOAO mosaic data handling system," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
10. D. Wittman, J. A. Tyson, G. M. Bernstein, D. R. Smith, R. W. Lee, M. M. Blouke, I. P. Dell'Antonio, and P. Fischer, "Big throughput camera: The first year," in *Optical Astronomical Instrumentation*, S. D'Odorico, ed., *Proc. SPIE* **3355**, pp. ?–?, 1998.
11. A. Szentgyorgyi, S. Amato, D. Fabricant, R. Fata, J. Geary, C. Hughes, B. McLeod, M. Ordway, W. Wyatt, H. Epps, and M. Lesser, "Two blue-sensitive CCD cameras for the Whipple Observatory 1.2 meter telescope," *in preparation* , 1998.
12. M. A. Conroy, J. B. Roll, and E. G. Mandel, "Building software from heterogeneous environments," *Astronomical Data Analysis and Software Systems VII* **in press**, pp. ?–?, 1998.
13. E. Bertin and S. Arnouts, "SExtractor: software for source extraction," *Astronomy and Astrophysics Supplements* **124**, pp. 393–404, 1996.
14. E. G. Mandel and W. Joye, "http://hea-www.harvard.edu/RD/," *Web Home Page* , 1998.