# Kepler Science Operations Center Architecture

Christopher Middour[a], Todd Klaus[a], Jon Jenkins[b], David Pletcher[c], Miles Cote[c], Hema Chandrasekaran[b], Bill Wohler[a], Forrest Girouard[a], Jay P. Gunter[a], Kamal Uddin[a], Christopher Allen[a], Jennifer Hall[a], Khadeejah Ibrahim[a],
Bruce Clarke[b], Jie Li[b], Sean McCauliff[a],
Elisa Quintana[b], Jeneen Sommers[a], Brett Stroozas[a], Peter Tenenbaum[b], Joseph Twicken[b], Hayley Wu[b], Doug Caldwell[b], Stephen Bryson[c], Paresh Bhavsar[c], Michael Wu[d], Brian Stamper[e], Terry Trombly[e], Christopher Page[e], Elaine Santiago[e]

[a] Orbital Sciences Corp., NASA Ames Research Center, M/S 244-30 Moffett Field, CA 94035 USA
[b] SETI Institute, NASA Ames Research Center, M/S 244-30 Moffett Field, CA 94035 USA
[c] NASA Ames Research Center, M/S 244-30 Moffett Field, CA 94035 USA
[d] Bastion Technologies, NASA Ames Research Center, M/S 244-30 Moffett Field, CA 94035 USA
[e] LOGYX, LLC, NASA Ames Research Center, M/S 244-30 Moffett Field, CA 94035 USA

## ABSTRACT

We give an overview of the operational concepts and architecture of the Kepler Science Data Pipeline. Designed, developed, operated, and maintained by the Science Operations Center (SOC) at NASA Ames Research Center, the Kepler Science Data Pipeline is central element of the Kepler Ground Data System. The SOC charter is to analyze stellar photometric data from the Kepler spacecraft and report results to the Kepler Science Office for further analysis. We describe how this is accomplished via the Kepler Science Data Pipeline, including the hardware infrastructure, scientific algorithms, and operational procedures. The SOC consists of an office at Ames Research Center, software development and operations departments, and a data center that hosts the computers required to perform data analysis. We discuss the high-performance, parallel computing software modules of the Kepler Science Data Pipeline that perform transit photometry, pixel-level calibration, systematic error-correction, attitude determination, stellar target management, and instrument characterization. We explain how data processing environments are divided to support operational processing and test needs. We explain the operational timelines for data processing and the data constructs that flow into the Kepler Science Data Pipeline.

Keywords: NASA, Ames, Kepler, CCD, transit photometry, architecture, infrastructure, software, Java, MATLAB, overview, operations, extrasolar, space telescope

## 1. INTRODUCTION

The Kepler Science Operations Center (SOC) designed, developed, and operates the Kepler Science Data Pipeline, a major analysis software component of the NASA Kepler Mission. The Science Operation Center's charter is to design, develop, operate, and maintain a science data pipeline for use in analyzing photometric data from the Kepler spacecraft in support of the Mission's search for Earth-like, extrasolar planets. During operations, the SOC analyzes stellar photometric data and reports the results to the Kepler Science Office. This is accomplished via the Kepler Science Data Pipeline. The pipeline performs four main functions: target observation planning (under direction by the Kepler Science Office), maintenance of target lists and target definition tables, photometer performance monitoring, and analysis of the stellar photometric data. The SOC software maintains list of stellar targets, which are delivered by the Science Office, and generates target definitions, which specify to the spacecraft which photometer pixels are to be used for a given quarter's observations. The software includes science analysis tools which are used by project scientists to retrieve information supporting analysis.

The Science Operations Center (SOC) also runs the science data pipeline for transit searches, manages the database of science targets, provides target data, and is responsible for monitoring and reporting the photometer's status to the project for further evaluation. The Kepler Science Data Pipeline receives raw instrument data from other ground segment

elements. The main source of data is the Data Management Center (DMC) located at the Space Telescope Science Institute (STScI) at The Johns Hopkins University in Baltimore, Maryland.

The SOC consists of an office at Ames Research Center, a software development organization, an operations department, and a data center which hosts the computers required to do the data analysis. TBD: ADD REFERENCE: INITIAL CHARACTERISTICS OF KEPLER LONG CADENCE DATA FOR DETECTING TRANSITING PLANETS http://iopscience.iop.org/2041-8205/713/2/L120/ The Kepler Mission has invested more than one hundred person-years in building a custom-designed transit photometry pipeline. The software has the capability to process 170,000 stellar targets and provide instrument calibration and performance metrics. Mission requirements demand a high degree of software parallelism, scalability, and data storage throughput. The unique Kepler instrument's signatures require software calibration and systematic corrections which are not part of standard photometry packages. Existing commercial and open source photometry packages did not scale to 5.5 million pixels sampled every half hour. Responding to the unique needs of the mission, SOC staff designed a platform of widely adopted hardware and software upon which to implement the custom algorithms. The programming languages MATLAB, Java, and C++ are the software languages used. Science algorithms are implemented in MATLAB, with data management provided by Java code. A small amount of C++ code is used to optimize performance. The Java code is responsible for retrieving the inputs for a given unit-of-work from the database servers, passing these inputs to the algorithm, and storing the resulting outputs in the data store once the algorithm completes. Standard Intel-architecture servers running Linux are used to execute the software.

## 1.1 Mission description

The spacecraft surveys a part of the Orion arm of our Milky Way galaxy. This neighborhood of the galaxy has enough potential targets and is far enough from the ecliptic so as not to be obscured by the Sun. Candidate objects for follow-up study are discovered by examining the amount of light emitted by each star, then looking for periodic dimming caused by a planet orbiting the star in an orientation that crosses between the Earth and the star.

Kepler data consist primarily of 30-minute (known as long cadence) samples for up to 170,000 stellar targets collected from 84 CCD output amplifiers (referred to as a CCD channel or a module/output), resulting in nearly 20 GB of raw, long cadence pixel data received by the SOC per month. In addition, Kepler collects 1-minute (short cadence) samples for up to 512 targets for an additional 4 GB of raw pixel data per month.

# 2. SOFTWARE ARCHITECTURE

The Kepler Science Data Pipeline software consists of approximately one million lines of code, 51% in MATLAB, 49% in Java. Functionality is spread across 25 software components, which cooperate via a flexible pipeline framework. High-level functions include:

1) Transform pixel data from the Kepler spacecraft into stellar light curves (flux time series).

2) Search each flux time series for signatures of transiting planets.

3) Fit physical parameters of planetary candidates, and calculate error estimates.

4) Perform statistical tests to reject false positives and establish accurate statistical confidence in each detection.

5) Manage target aperture and definition tables specifying which pixels in the spacecraft's CCD array are to be downlinked.

6) Manage the science data compression tables and parameters.

7) Report on the Kepler photometer's health and status semi-weekly after each low-bandwidth contact and monthly after each high-bandwidth science data downlink.

8) Monitor the pointing error and compute pointing tweaks when necessary to adjust the spacecraft pointing to ensure the validity of the uplinked science target tables.

9) Calibrate pixel data to remove instrument systematics

10) Archive calibrated pixels, raw and corrected flux time series, and centroid location time series.

The SOC Operations department configures and executes the Kepler Science Data Pipeline software. TBD: Add reference: Kepler Science Operations Processes, Procedures, and Tools Operations staff maintain software models containing thousands of parameters, update focal plane characterization (FC) models, perform data acceptance with the Kepler Science Office, perform data accounting, configure and execute processing pipelines, and generate archive products using Kepler Science Data Pipeline software.

## 3. HARDWARE ARCHITECTURE

The Kepler Science Data Pipeline hardware architecture is built from commodity server and high end storage components. The Fedora 11 or Red Hat Enterprise GNU/Linux operating system is used on all servers. The modular architecture allows the SOC to integrate additional computing power into the pipeline with a minimum of effort, as needed. The hardware is partitioned into separate clusters, which are configuration-managed and used for different purposes, such as monthly processing or quarterly transit searches. Firewalls isolate each cluster, which use separate user access-lists based on the identified minimum set of users required.

A cluster nominally consists of a relational database server, a time series database server, and fourteen compute servers, which execute the SOC's science data processing algorithms and pipeline software. The servers are dual quad-core Xeon X54702 series processors (eight cores per server), containing 32-gigabytes of RAM at 2.6 to 3.0 GHZ speed. The photo in Fig. 1 shows the as-built system located within the SOC at NASA Ames Research Center:



Figure 1. Kepler Science Data Pipeline at NASA Ames Research Center.

## 4. SOFTWARE COMPONENTS

The Kepler Science Data Pipeline is configured as multiple pipeline segments. Each pipeline segment is based on the particular type of dataset it processes and how frequently it runs. Each pipeline segment consists of a configurable sequence of modules. In this context, a pipeline module is a coarse grained task that implements a set of algorithms. The set of pipeline segments, the pipeline modules, and the module library are completely configurable. This architecture enables modules to be easily updated, added, or removed and minimizes code changes. Fig. 2 graphically represents the Kepler Science Data Pipeline software components and illustrates the data flow between the software components.

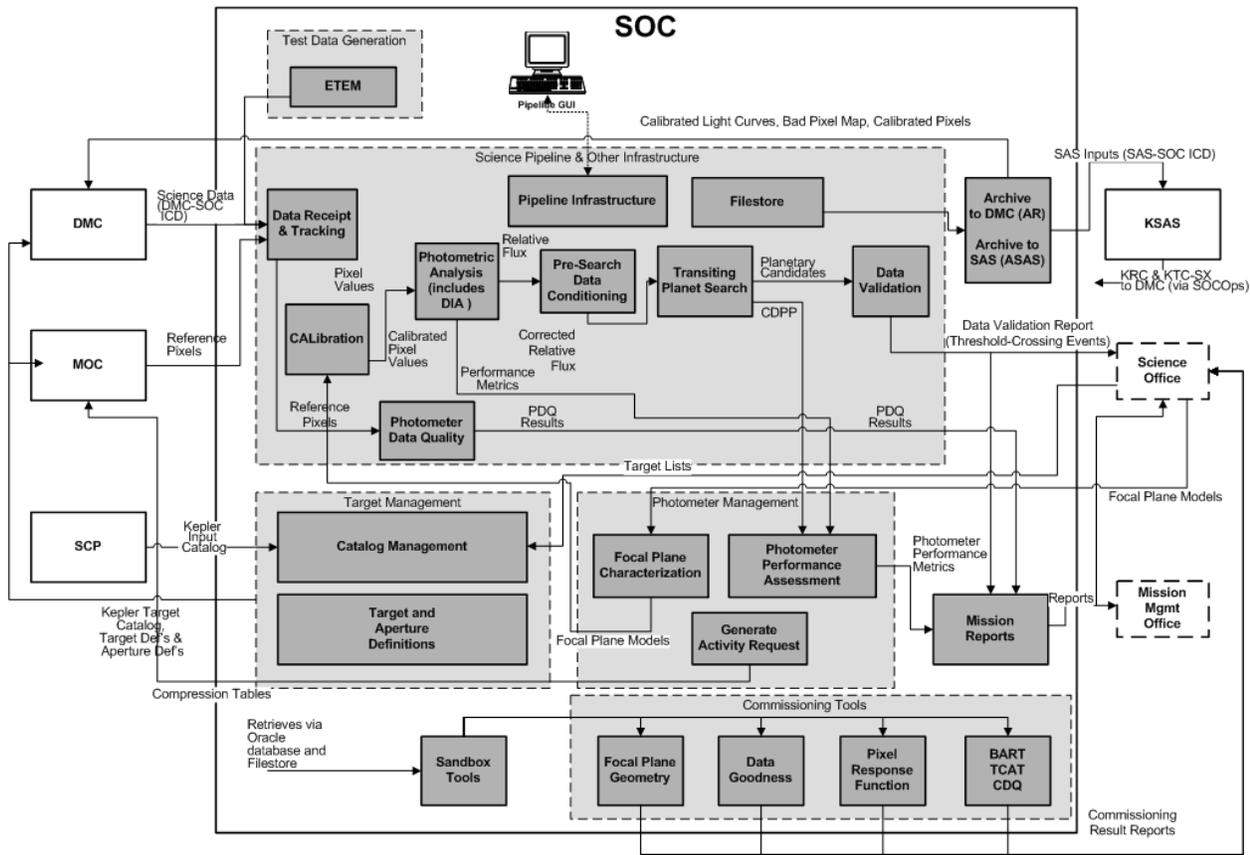TBD: Streamline the diagram and remove the outside elements.

Figure 2. Kepler Science Operations Center Architecture

## 4.1 Software infrastructure functionality

The framework of the Kepler Science Data Pipeline software provides the basic functionality for data receipt , storage, and processing. This functionality is provided by the following components: Pipeline Framework (PF), Data Receipt (DR), Kepler DB , Mission Reports (MR), and Archive to DMC (AR). While developed for Kepler, none of the framework software [TBD: Add reference: Title: "Kepler Science Operations Center pipeline framework."]   Add reference:  "The Kepler Science Operations Center pipeline framework extensions" is Kepler-specific and could be used for other applications where these services are needed.

The Pipeline Framework component TBD: Add reference: Title: "Kepler Science Operations Center pipeline framework." provides basic functionality for data communications, storage, retrieval, and manipulation. To support parallel processing requirements, the Pipeline Framework software partitions tasks into small pieces, which can execute on a single CPU core. Jobs are distributed to the workers via a Java Message Service (JMS) queue. Jobs are not pre-assigned to specific workers; rather the jobs are placed on the queue where worker threads can claim them when they are ready to accept a new job. In this way, jobs are dynamically load-balanced across the cluster. There is no centralized controller for pipeline execution. Instead, the pipeline transition logic is performed by the workers in a distributed fashion. As each job completes on a particular worker machine, the worker is responsible for executing the transition logic that generates the jobs for the next module. The ways the data can be broken up depend on the nature of the pipeline module algorithms.  For example, the Pre-search Data Conditioning (PDC) module needs data from all stars for a given CCD channel, but each month of data can be processed independently.  In contrast, the Transiting Planet Search (TPS) module can operate on a single target at a time, but needs as many time samples as possible. The Pipeline Framework manages the distribution and synchronization of these tasks across more than 100 computer cores and

provides a framework for each worker machine to call other software modules; it also provides common services like user authentication, logging, and error handling.

The Pipeline Framework's graphical console is a Java/Swing application that displays the attributes of the other pipelines that are running, including how much data has been processed, what data are currently being processed, and average throughput. The interface TBD: Add Reference Title: The Kepler SOC Pipeline Configuration and Execution provides control over starting, stopping, and configuring pipelines.

To support development and normal operation, the Pipeline Framework software can scale to run on a single PC to a full cluster of servers. A software developer can run the pipeline on a laptop during development, and then the same software can be deployed to the operational system without modification. The Pipeline Framework manages data integrity by coordinating storage transactions between the relational database and the Kepler DB software. To support management of the system, alerts to operators are displayed, and a performance metrics subsystem provides reports. The metrics subsystem provides a simple API that can be used to instrument strategic locations in framework or application code for the purpose of collecting statistics on pipeline performance.

The Data Receipt (DR) component handles the receipt of science data that is sent from the DMC to the SOC. Data are parsed using XML, FITS, and binary parsers, depending on the type of data received. The parsed data are persisted into a popular relational database or Kepler DB, depending on the type of data received. DR also provides APIs for other pipeline components to access the persisted data.

The Kepler DB component is a custom-designed high capacity, high-performance transactional database management system. It is used to store time series data and generic byte arrays. Kepler BD is used to store the vast majority of the data collected from the spacecraft as well as final and intermediate data produced by the various pipeline modules. TBD: ADD REFERENCE TO FILESTORE Title: The Kepler DB, a database management system for arrays, sparse arrays and binary data. Authors: Sean McCauliff

The Mission Reports (MR) component is a web-based report viewing system. Data from other software modules display their reports via MR, which generates mission reports for use by the SOC, Science Office, and Mission Management Office. MR generates some reports based on data persisted by other components and simply presents some reports (generic mission reports) generated by other components. MR includes photometer performance assessment reports, science pipeline performance reports, and mission status reports.

The Archive to DMC (AR) component compiles mission data (usually in the FITS format) for export to the permanent archive at the STScI.

## 4.2 Target management functionality

The target management portion of the SOC science pipeline provides basic functionality for managing target observations. Target management is one of the core SOC activities, consisting of target selection, generation of target and aperture definitions, uplink of target/aperture definition tables to the spacecraft, and performance verification for those newly uplinked tables. Once per quarter (3 months), the spacecraft is rolled 90 degrees to re-orient the solar panels to the Sun. This also relocates the apparent position of stars on the photometer, so the SOC collaborates with the Science Office to produce new target tables, which are rotated to match the new stellar positions. TBD: Add Reference: SELECTION, PRIORITIZATION, AND CHARACTERISTICS OF *KEPLER* TARGET STARS http://iopscience.iop.org/2041-8205/713/2/L109/ The target selection process provides lists of prioritized targets of various categories that are balanced to fill the final tables. Although the planetary target list accounts for the vast majority of the science targets, a number of other lists are included to provide photometer performance information (e.g., stellar, and image artifact targets), while a host of others perform different scientific duties (e.g., comparison, Guest Observer, and eclipsing binary targets). The Science Office has developed a suite of MATLAB software tools to perform target selection, and to ensure that estimated target/pixel allocations remain within the operations constraints of the spacecraft and bandwidth available.

SOC target management software includes the following components: Catalog Management (CM) is a user interface for selecting or importing target lists and providing them to target management software. CM provides a graphical user interface to perform three distinct functions:

(1) validate the Kepler Input Catalog for correct format and missing data, load it into the SOC database, and generate KIC statistics and metrics;

(2) assemble target lists specified by the Science Office into the Kepler Target Catalog and add observation start and stop time, target source, ID, and cadence; and

3) display or export data from the characteristics table or any of the Kepler catalogs based on science criteria used to select entries from that catalog.

Target and Aperture Definitions (TAD) creates target definition tables for long cadence targets and short cadence targets by selecting pixel locations on the focal plane and aperture patterns for each target. TBD: Add Reference Title: "Selecting Pixels for Kepler Downlink" Authors: Bryson TAD also creates a target definition table for selected background and reference pixels as well as aperture pattern definition tables for regular targets and background pixels.

## 4.3 Photometer management functionality

The photometer management portion of the SOC Science Pipeline provides the basic functionality for assessing, monitoring, and improving photometer performance in order to maximize the scientific quality of the resulting data. It includes the following: The Photometer Data Quality (PDQ) component operates on a pixel set from the spacecraft to produce metrics relating to spacecraft attitude, photometer brightness, and focus. Twice per week, the spacecraft sends a small number of pixels (called Reference Pixels) to Earth. PDQ is executed, and the resulting report is provided to the project for analysis. The Photometer Performance Assessment (PPA) component TBD: Add Reference: Photometer Performance Assessment in Kepler Science Data Processing reports metrics on photometer performance, including spacecraft attitude, brightness, and focus. The PPA software is designed to have access to all downlinked pixels, which results in higher accuracy than the PDQ report, which is constrained to function with a smaller subset of pixels, but at a higher frequency than PPA.

The Focal Plane Characterization (FC) component TBD: ADD REFERENCE TO FC paper*Title*: Kepler Mission's Focal Plane Characterization Models Implementation calculates physical attributes of the focal plane as measured during commissioning and computes point-spread functions based on Full-Field Image (FFI) data sets. These are used to support target selection and focus analysis. Focal plane characteristics include: CCD arrangement, alignment, and angles; gaps between CCDs; focus; and target point-spread functions. Models are imported into FC when required. A history of models imported is maintained, supporting data accountability needs, and to support reprocessing older data. FC is capable of interpolating between points. For example, the 2-D black model is occasionally updated. FC is capable of interpolating in the time interval between updates. FC models are retrieved directly by the pipeline at runtime for processing, but there are additional query tools, known as "Science User Tools," which are used by mission scientists for offline analysis and reporting.

The Generate Activity Request (GAR) component generates compression tables for upload to the spacecraft via Kepler's Mission Operations Center at the Laboratory for Atmospheric and Space Physics at the University of Colorado and the DMC.

## 4.4 Science Data Processing Functionality

The data analysis portion of the SOC science pipeline includes pixel-level calibration, light curve generation, systematic error correction, outlier identification, gap filling, identification of threshold crossing events, and validation. This functionality is provided by the following components:

- Calibration (CAL)

- Photometric Analysis

- Pre-search Data Conditioning

- Transiting Planet Search

- Data Validation

TBD: Add Reference: OVERVIEW OF THE KEPLER SCIENCE PROCESSING PIPELINE Jon M. Jenkins et al 2010 ApJ 713 L87

Calibration (CAL) TBD: ADD REFERENCE: Title: Pixel Level Calibration in the Kepler Science Operations Center Pipeline is the first processing step applied to pixel data in the pipeline. CAL ingests uncalibrated data from DMC and

calibrates it for a fixed-offset value, a mean black value (per CCD pair), a bias value, non-linearity, gain, undershoot (an artifact introduced by a clamp circuit in the photometer electronics), smear (due to the lack of a shutter in the telescope), dark (a thermally induced signal) and a standard flat field correction. The pipeline's CAL software removes cosmic rays from collateral data. Detecting cosmic rays in these types of pixels is done via a simple threshold algorithm. This is possible with these pixel types because they do not receive light input, and therefore cosmic rays stand out clearly over the otherwise low signal levels. To allow parallel processing and memory management, CAL processing is divided into units of work based on the source module/output, and the type of pixel. Collateral data are processed in the first invocation, in order to have black and smear values available at the start of photometric pixel processing. Processing uncertainties or "errors" are calculated by CAL's Propagation of Uncertainties (POU) software. Given the large data volumes pertaining to measuring errors in data processing, a novel approach was taken to provide uncertainty information within the computer resources available. (TBD: ADD REFERENCE  A Framework for Propagation of Uncertainties in the Kepler Data Analysis Pipeline AUTHORS: Bruce Clarke(1) Ideally, a software system would calculate errors in lockstep with data processing, but given the Kepler data volume, a solution like this would be impossible due to computer memory constraints.  The solution includes storing the relatively small set of primitive data and meta-data needed to reproduce pixel covariance information rather than storing the full covariance matrix. A singular value decomposition (SVD) is performed, keeping only the highest power components to further reduce the size of the stored information with the added benefit of acting as a low-pass filter. Elements of the pixel covariance matrix are reconstructed to within 1 part in 1000 of the original value which is within requirements for the propagated errors. This method significantly reduces the memory needed to execute.

The photometric Analysis TBD: Add reference: Photometric Analysis in the Kepler Science Operations Center (PA) component converts pixels values to target light curves.  PA uses simple aperture photometry along with background-corrected, calibrated target pixel values to generate a flux time series per target.  Cosmic rays are identified and removed in both the background and target pixels. Detecting cosmic rays in photometric data (pixels which receive stellar flux) requires information over time, so cosmic ray removal from photometric pixels is performed in PA, which is capable of processing multiple time samples. Centroids are calculated for each target star on each frame.  Argabrightening events are identified and reported.  A number of metrics are calculated and persisted.

The Pre-Search Data Conditioning (PDC) component TBD: Add reference: Presearch Data Conditioning in the Kepler Science Operations Center Pipeline is used to remove systematic errors prior to performing transiting planet searches. Error corrected light curves are also exported to the MAST archive. PDC operates on raw flux light curves from PA and performs systematic error correction, outlier identification and data gap filling. Artifacts in light curves caused by thermal changes are removed by cotrending with engineering data from the spacecraft, such as temperature sensors. Spacecraft motion effects on light curves are addressed with the use of motion models produced by PA. Random discontinuities due, for example, to cosmic ray induced sensitivity changes are identified and corrected. Excess flux due to crowding in stellar apertures is removed. When systematic errors are corrected, an attempt is made to ensure that large transits (and other astrophysical events such as binary eclipses and flares) are left intact. Gaps in data caused by scheduled downlinks, or safe modes onboard the spacecraft, are filled.

TBD: can the safe-mode correction diagram be used?

The Transiting Planet Search (TPS) component TBD: Add Reference: Transiting Planet Search in the Kepler Pipeline applies a wavelet-based, adaptive, matched filter to identify transit-like features with durations of 1-to-16 hours. TPS makes use of the transit photometry method, examining the amount of light emitted by each star, then looking for periodic dimming caused by a planet orbiting the star in an orientation that crosses between the Earth and the star. At the moment the planet is directly between the telescope and the star, it dims the light the telescope sees be a small fraction. The TPS component measures the amount of dimming, and the length of time over which it occurred. Light curves with transit-like features and a combined transit detection statistic exceeding $7.1\sigma$ for some trial period and epoch are designated as "threshold crossing events," which are then subject to analysis by the Data Validation module of the pipeline. This threshold ensures that no more than one false positive deriving from random fluctuations will occur over the life of the mission (assuming non-white, non-stationary Gaussian observation noise).

The Data Validation (DV) component TBD: Add reference: Data Validation in the Kepler Science Operations Center Pipeline performs a suite of statistical tests to evaluate the confidence of a threshold crossing event, identify and reject false positives caused by background eclipsing binaries, and extract the physical parameters of each system for each candidate planet TBD: add reference: An Algorithm for Fitting of Planet Models to Kepler Light Curves (together with uncertainties and covariance matrices). Because it's possible that other phenomena besides actual transits can cause a

light curve signature to falsely appear to be a planet, the SOC compares the transit signatures with spacecraft engineering data (such as onboard temperatures and reaction wheel speeds) to eliminate false positives. The results are also compared to Kepler's laws of planetary motion, which is a further way to eradicate false positives. The results of this search are provided to the Follow-Up Observation Program (FOP), which schedules time on Earth-based telescopes to confirm the planetary candidates. After DV fits the planetary signature, DV removes it from the light curve and subjects the residual to a search for additional threshold crossing events. DV repeats the process until it identifies all threshold crossing events. The Kepler Science Operations Center also provides the processed data to the Threshold Crossing Event Review Team, who evaluates and prioritizes the threshold crossing events for ground-based follow-up observations.

## 4.5 Spacecraft Commissioning Software

In Spring 2009, the mission conducted a thorough checkout of spacecraft, ground station, ground software, and instruments. The SOC participated in this exercise using custom-built analysis tools. Once the spacecraft was in orbit, the commissioning phase began TBD: Add Reference Kepler Science Operations http://stacks.iop.org/2041-8205/713/L115 (Astrophysical Journal Letters 713 (2010) L115-L119), which turned on the photometer, and compared on-orbit science measurements to expected values based on ground-based tests. The spacecraft then ejected the dust cover, based on the reports from the SOC. During commissioning, many full-frame images of all photometer pixels were sent to the SOC for analysis by the commission tool suite. These tools were specifically designed for Kepler by the SOC:

The Focal Plane Geometry (FPG) TBD: Add Reference: Focal Plane Geometry Characterization of the Kepler Mission Authors: PT (corresponding author) software fits the actual position of each CCD on the focal plane using star data. FPG measures the as-built positions and orientations of the CCDs on the focal plane of the Kepler Mission's flight segment, using the night sky as a high-precision metrology tool. It is necessary to maximize the number of stellar targets which can be monitored for signs of planetary transits. Due to bandwidth and data processing limitations, this translates to a requirement to minimize the number of pixels devoted to each stellar target. To do this, the sky-to-pixel geometry must be as accurate as possible.

The Data Goodness (DG) software checks for data completeness or data corruption, possibly indicating a transmission or data handling problem. DG checks data content to see whether the right data were taken, and then checks data values to see if they indicate a potential problem with the hardware. A report is generated with all plots, histogram and statistics of each individual detector array. This software tool continues its usefulness in normal operations, and is routinely executed on FFIs from the spacecraft.

The 2D Black and Artifact Removal Tool (BART) component detects and models temperature-dependent image artifacts in pixel data. TBD: Add reference: "Flagging and Correction of Pattern Noise in the Kepler Focal Plane Array." J. Kolodziejczak The main purpose of BART is to support the decision to eject the spacecraft dust cover. BART provides insight into whether the photometer data are consistent with pre-launch expectations regarding temperature variations. The spacecraft's data gathering process is subject to a number of instrument artifacts which can damage the quality of the mission's data if not properly managed. Of particular interest is the crosstalk from the spacecraft's fine guidance sensor (FGS) readout into the science CCD readout, which has a temperature-dependent intensity. In order to properly correct for this effect, it is necessary to determine the temperature dependence of the crosstalk intensity and to produce a 2-D black model which incorporates the correct crosstalk intensity (i.e., to estimate the crosstalk intensity at the spacecraft's nominal operating temperature). The temperature dependence of the 2-D black will be studied via a series of Full Field Images (FFIs) which are acquired at different temperatures. Once the BART processing is complete, the CDQ and TCAT tools report further statistical analyses on BART's resulting data files.

The Check Data Quality (CDQ) component checks and analyzes the RMS of data fitting residuals and thermal coefficients produced by BART for the pixels in the collateral regions of the photometer. The collateral pixel regions are areas that do not receive light, but are used for photometer diagnostics. CDQ reports the statistics on the RMS of data fitting residuals and thermal coefficients, and provides different types of plots.

The Temperature Coefficient Analysis Tool (TCAT) component is a tool to study the thermal variations of fine guidance sensor cross-talk pixels. TCAT reports statistics on the thermal variations of Fine Guidance Sensor (FGS) cross talk pixels (pixels which are affected by crosstalk). FGS Crosstalk is a major source of Kepler instrument noise. In order to calibrate pixels to remove FGS crosstalk, it is necessary to understand its thermal variation.

The Pixel Response Function (PRF) component TBD: Add Reference: THE *KEPLER* PIXEL RESPONSE FUNCTION http://iopscience.iop.org/2041-8205/713/2/L97 fits the shape of the pixel response function for each module/output using star data. A PRF is defined as the optical point-spread function (PSF) convolved with pixel structure and motion. The Pixel Response Function (PRF) tool produces a continuous model of each photometer detector array's responsivity to light. It also provides quality metrics for each PRF produced. This tool aids the SOC and Science Office in choosing pixels for optimal photometry, determining which pixels are downlinked, and in determining the focus center of each star (centroid). PRF works with the Focal Plane Geometry (FPG) tool in an iterative fashion, eventually coming up with an accurate geometry model. The default PRF pipeline is configured to loop back to FPG and then again to PRF over a configurable number of iterations. The iteration stops when the centroid change is below a configurable threshold. The computed PRF(s) are stored each iteration so there is an opportunity for examination by Science Office personnel. If the PRF(s) are approved it is delivered to the SOC. If the PRF(s) are unacceptable, corrective action is determined and the process flow is restarted from an appropriate stage.

The Science User Tools component provides access to target tables, light curves, and other data products for use by mission scientists. It provides read-only querying capability for use in data analysis.

### 4.6 Flight Data Simulation

The Kepler End-to-End Model (ETEM) component is a Monte Carlo approach to produce flight-like data for the Kepler photometer so that the impacts of noise sources and systematic effects that are not amenable to direct analysis can be studied. The software produces simulated pixel data in the exact same format which exists on-board the spacecraft, (CCSDS and VCDU). After launch, ETEM is still used as a data source when ground truth is required for testing and debugging algorithms.

## 5. SUMMARY

We have highlighted the software and hardware architecture of the SOC, discussed high-level functionality implemented, and described features of each software component. Interested readers will find a wealth of information about the SOC and the Kepler mission in the referenced papers, which are authored by the software developers and scientists who designed and implemented the software discussed here.

## ACKNOWLEDGMENTS

## REFERENCES

**TBD: ADD REFERENCES**

[1] Booth, N. and Smith, A. S., [Infrared Detectors], Goodwin House Publishers, New York & Boston, 241-248 (1997).
[2] Davis, A., R., Bush, C., Harvey, J. C. and Foley, M. F., "Fresnel lenses in rear projection displays," SID Int. Symp. Digest Tech. Papers 32(1), 934-937 (2001).
[3] Van Derlofske, J. F., "Computer modeling of LED light pipe systems for uniform display illumination," Proc. SPIE 4445, 119-129 (2001).
[4] C. Jones, Director, Miscellaneous Optics Corporation, interview, Sept. 23, 2008.
[5] www.optics4yurresearch.com/7752.html