Upgrades to Common Data Acquisition System Software Development for NASA's Rocket Propulsion Test Facilities and Software Reuse

Phillip W. Hebert, Sr.[1], Alex C. Elliot[2]
NASA, Stennis Space Center, MS, 39529

Kristopher A. Mobbs[3]
NASA, Stennis Space Center, MS, 39529

ABSTRACT

Approximately five years ago, the National Aeronautics and Space Administration (NASA) Stennis Space Center (SSC) resumed operation of its large rocket engine test facilities after thirty years of contractor control. During this period, contactors used their own proprietary Data Acquisition System (DAS) to record and process rocket propulsion test data. The transition from a contractor managed facility to a NASA managed facility posed a difficult challenge. In order to support the commercial space launch initiative, SSC needed to develop a software replacement for the contractor proprietary DAS. This replacement software would enable SSC to operate propulsion test facilities more cost effectively and to be more readily able to adapt software for reuse, while at the same time provide internal and external customers with reliable population test data.

Therefore, SSC developed in-house, a non-proprietary software suite of applications to replace the previously used proprietary DAS. The requirements for the DAS suite included recording and processing propulsion test data. This capability eliminates the necessity for customers to provide a DAS or rely on a competitor's DAS. An additional benefit of owning the software suite included enabling the ability to add additional features and functionality at a lower cost. The Rocket Propulsion Test (RPT) Program Office reviewed consideration for funding this project with the caveat that development of the software included availability for use with minimal modifications to all SSC test facilities and RPT centers: Marshall Space Flight Center (MSFC), White Sands Test Facility (WSTF), and Glenn Research Center (GRC) Plum Brook Station. Based upon this guideline, SSC created the NASA Data Acquisition System (NDAS) software suite.

The ability to use the software at multiple centers, even though each field center uses differing DAS hardware with different capabilities, drove a requirement that the software design be portable with minimal modifications to the software. Then, with software release requirements, evaluations, and approvals completed, the NDAS software suite could also become available to other government agencies, corporations, universities, and the general United States public.

[1] Electrical Engineer, Experimental Electrical Equipment & Techniques, Electrical Design & Software Branch, Engineering & Test Directorate, EA-31, Stennis Space Center, MS, 39529; AIAA Senior Member.

[2] Computer Engineer, Electrical Design & Software Branch, Engineering & Test Directorate, EA-31, Stennis Space Center, MS, 39529; non-AIAA member.

[3] Electrical Engineer, Experimental Electrical Equipment& Techniques, Test Operations Electrical Branch, Engineering & Test Directorate, EA-21, Stennis Space Center, MS, 39529; non-AIAA member.

I. Introduction

One of the primary drivers for developing the NDAS software suite was NASA's recent takeover of operations at SSC’s large test facilities after thirty years of contractor control. With the advent of the commercial space launch industry and the evolving potential for testing a wider variety of contractor engines, the need for a non-proprietary data acquisition system to support government and commercial engine testing became a priority. The software provides the government with unlimited rights and guarantees privacy of data to commercial entities.

Another driver for developing NDAS was to provide a measure of commonality of software for the diverse set of data acquisition systems within and across the NASA RPT centers. Each system contains differing hardware and software platforms, even within the SSC test facilities. Some of these facilities also employ differing downstream support applications, which, in some cases, require modification to function properly with NDAS. Some test facilities used in aerospace test projects/missions also require the use of specialized equipment, specific to that test facility, rather frequently. Specialized data system hardware also typically requires specialized software. This specialization leads to custom software to operate these systems. NDAS addresses the issue of providing common DAS software to these facilities, and can provide commonality for these specialized systems, if the customers so desire.

There are also data stream conversion functions that can be used to translate test facility data streams to and from downstream systems including customer engine systems. The software design is modular to accommodate test facility and test program variability. For example, one test program may require real-time displays and data recording; while, others may require more complex data stream conversion, measurement filtering, or test stand configuration management. NDAS can meet both these needs.

II. Purpose

The primary mission of SSC’s is to perform testing of chemical rocket propulsion test articles and provide test data to test article customers. To accomplish this mission, test facilities use data acquisition systems to acquire, process, and provide this data.

Data acquisition systems require software to perform their required functions, including acquiring, storing, and distributing data. NDAS is designed to operate and control many critical aspects of rocket engine testing. NDAS combines real-time data visualization, data recording to a variety formats, short-term and long-term acquisition system calibration capabilities, test stand configuration control, and a variety of data post-processing capabilities. Additionally, data stream conversion functions exist to translate test facility data streams to and from downstream systems, including engine customer systems.

Historically, data system software at SSC was re-written or extensively modified for each new test hardware configuration. The modified DAS software developed used the DAS software from a previous test article project as a starting point. However, porting the software to a different data system platform was extremely difficult because the previous software was not modular enough in design to adapt to the new hardware environment in a cost effective way. The conventional means of developing test article data acquisition software was both schedule and resource intensive. The development of NDAS directly addresses these inefficiencies. NDAS improves operational efficiency of aerospace propulsion testing which further advances propulsion technology development. The NDAS design includes the capability to operate and control many critical aspects of rocket engine testing to address the varying needs of different facilities. The key to the software being portable to other test facilities is a translation layer, which allows higher-level application functions to access hardware through an abstracted interface. By providing a standard set of hardware calls and a flexible and easily accessible database, software application layers can easily communicate with underlying hardware layers regardless of test facility location.

III. NDAS System Overview

The primary overarching design goals for NDAS were flexibility, extensibility, and modularity. Providing a common user interface for a variety of hardware platforms helps drive consistency and reduce errors during article testing. In addition, since test facilities have different requirements and setups, software design needs to be modular. The NDAS suite allows test facility users to choose the specific components needed.[4]

NDAS incorporates Hardware Abstraction Layers (HALs) which more readily enables adaptation of the software to different hardware platforms. In addition to the HALs, the architecture groups the software into modules based on function. The NDAS suite includes the main operation and control module (NOPS), the engineering unit processing

[4] AIAA-2015-2561. NASA Data Acquisition System Software Development for Rocket Propulsion Test Facilities, AIAA Aviation and Aeronautics Forum and Exposition 2015

module (NPRO), the display module (NDIS), the calibration module (NCAL), the data logging module (NLOG), the data stream gateway (NGATE), and the instrumentation and roadmap database (NIRD). The NDAS one stop shop (NOSS) serves as the front-end for all user interaction with NIRD.

The software suite consists of six major capabilities organized into distinct components:

1. **System Configuration** – NOSS (NDAS One Stop Shop) serves as the centralized acquisition system configuration interface for engineers to configure measurements and define the test stand hardware configurations.
2. **Data Acquisition & Distribution** – NOPS (NDAS Operations) acquires raw data, scales the data to engineering units, and propagates all measurement data to client applications.
3. **Data Visualization** – NDIS (NDAS Display) serves as the client application for real-time data graphing, tabular data displays, and event monitoring.
4. **Data Recording** – NLOG (NDAS Logger) serves as the client application for data logging and data file conversion for long-term storage and analysis.
5. **Data Gateways** – NGATE (NDAS Gateway) serves as the interface between NDAS and downstream systems, converting engine data streams to NDAS data streams and *vice versa*.
6. **System Calibration** – NCAL (NDAS Calibration) performs short-term and long-term electrical calibrations on the data acquisition hardware.

Fig. 1 shows the NDAS components and how they relate to one another. Note that measurement configuration data for all components is stored in a database called NIRD (NASA Instrumentation Roadmap Database). Acquired measurement data is broadcast over a network to all client components.[5]

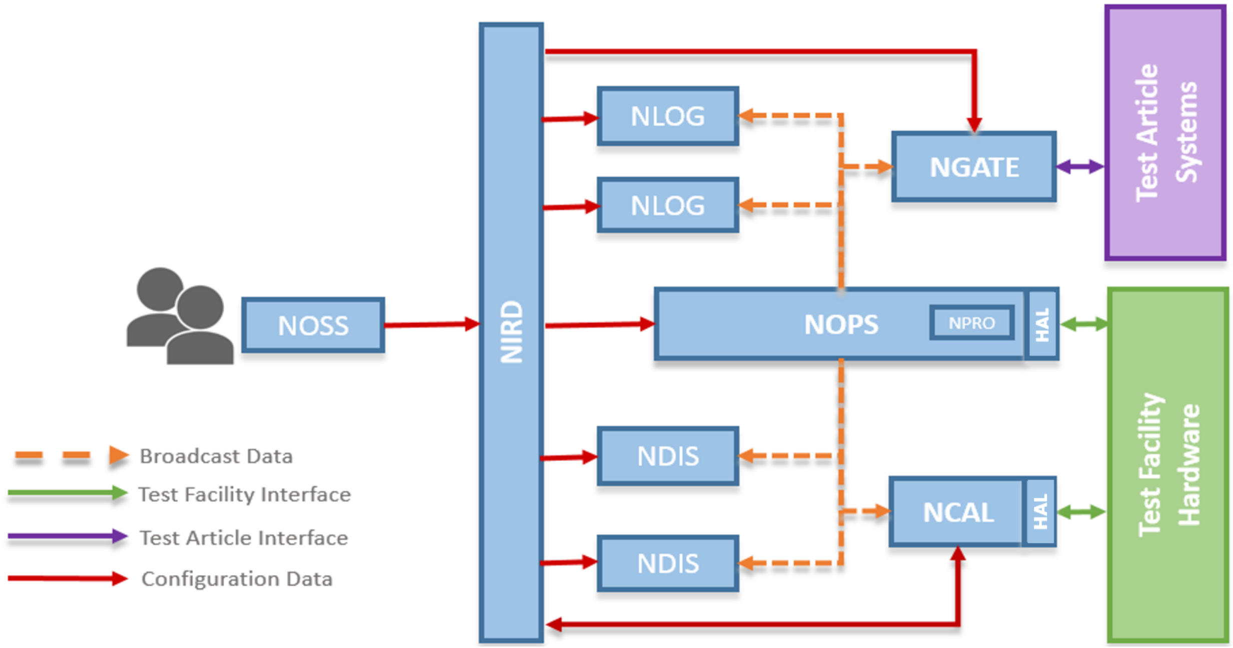


Figure 1. NDAS Block Diagram

[5] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

IV. NDAS System Architecture[6]

The NDAS system architecture can be organized into three major sections:

A. Data Management: NDAS data storage and access

NDAS utilizes two types of data stores: MySQL™[7] databases and data files. System and measurement configuration data and metadata are stored using MySQL™ databases. There are currently two instances of MySQL™ databases. The first is called the NASA Instrumentation Roadmap Database (NIRD) database. This database is utilized by all NDAS applications to get measurement configuration information. The NOSS database is utilized exclusively by the NOSS application and is used to store the current test stand or facility configuration, including how all hardware is connected.

Data files are used to store the acquired data from the NDAS system. The NDAS native data file format is Technical Data Management Solution (TDMS)[8]; however, NDAS supports other file formats such as WinPlot[9] and MATLAB®[10].

B. Communication: NDAS data flow

Individual NDAS applications communicate over an Internet Protocol (IP) network. Data packets are broadcast via User Datagram Protocol (UDP) for all applications to see. This type of communication mechanism allows the NDAS client applications (NDIS and NLOG) to be distributed across multiple locations. Additionally, it allows the number of client applications to scale up or down as needed. As an example, the NDIS application can be deployed to all operator consoles in a Test Control Center (TCC), all required machines on a test stand, and other machines offsite as needed. The NLOG application can be deployed on the test stand and in the TCC to provide redundancy.

Applications communicate with the databases over Open Database Connectivity (ODBC)[11] connections.

C. Software Applications: NDAS data creation, modification, and display

Most NDAS applications are written in the LabVIEW™[12] programming language using the Actor framework, an object-oriented, message-based architecture. LabVIEW was chosen for the following reasons:

1. There is a large library of drivers for data acquisition hardware and test instruments.
2. The source code is very similar to circuit diagrams, a familiar paradigm for electrical engineers.
3. There are many Graphical User Interface (GUI) controls built into the language for graphical displays, charts, etc. to speed up development.
4. Multitasking and multithreading are built-in and managed by the language.

[6] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

[7] MySQL™ is a trademark of Oracle Corporation. Neither NASA Stennis Space Center, nor any software programs or other goods or services offered by NASA Stennis Space Center, are affiliated with, endorsed by, or sponsored by Oracle Corporation.

[8] http://www.ni.com/white-paper/3727/en/

[9] WinPlot is a NASA developed product used to view time-domain data.

[10] MATLAB is a registered trademark of The MathWorks, Inc.

[11] Microsoft is a registered trademark, and ODBC, Windows, Windows NT, Windows 95 and Windows 2000 are trademarks of Microsoft Corporation. Neither NASA Stennis Space Center, nor any software programs or other goods or services offered by NASA Stennis Space Center, are affiliated with, endorsed by, or sponsored by Microsoft Corporation.

[12] LabVIEW™ is a trademark of National Instruments (NI). Neither NASA Stennis Space Center, nor any software programs or other goods or services offered by NASA Stennis Space Center, are affiliated with, endorsed by, or sponsored by National Instruments.

V. NDAS Detailed Architecture

A. NOSS[13]

NOSS is the central configuration interface for NDAS, and can be used more broadly to track the entire test facilities hardware and measurement configuration. NOSS is a web application that allows concurrent multi-user access.

NOSS is organized around two main building blocks: nodes and measurements.

Nodes represent physical and logical components of a test facility. Examples of physical components are signal conditioner cards, receptacle boxes points, filters, attenuators, digitizer cards, transducers, valves, etc. Logical components are acquisition system channels, facility redline system channels, or any other software-type endpoint systems. Therefore, NOSS can be used to track and manage the current state of all components on a test facility.

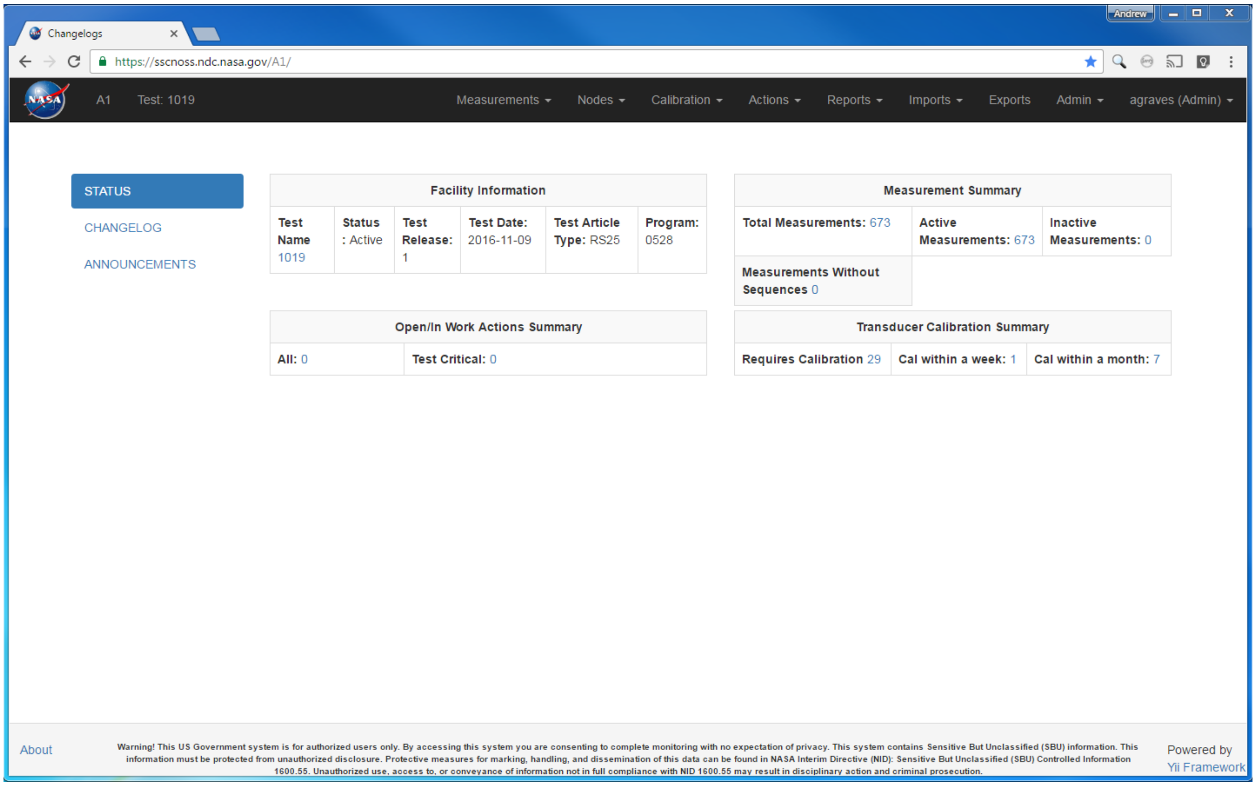
Measurements represent the full signal path from a signal source (commonly a transducer) to acquisition system endpoint, and are defined by linking together the nodes defined in the system. Additional measurement information can also be defined such as a unique measurement name, description, etc.

1. NOSS Major Functions

- **Defining measurements** – Add new measurements, delete measurements, inactivate measurements, or change the configuration of measurements in the system, including calibration parameters.
- **Defining derived measurements** – Add, delete, and update all derived measurements (i.e. calculated measurements that contain inputs from other measurements) in the system.
- **Defining acquisition hardware setup** – Configure and track the current state of all acquisition hardware and instrumentation, including patching.
- **Running reports on the system configuration** –custom reports on all data that exists in the system, and export the reports to multiple formats.
- **Archiving system configurations** – Archive and save snapshots of configurations.
- **Exporting system configurations** – Export the system configuration to run the NDAS system or other systems that are configured in NOSS (FRS or other high speed acquisition systems for example).
- **Importing configuration data** – Bulk import data in various formats.
- **Tracking work actions** – Create tickets to manage work on the acquisition systems, such as adding or removing transducers, adding or removing patching, or verifying specific measurement setups.
- **Tracking changes to system configuration** – View a changelog.
- **User Management** – Administrative users may add, remove, inactive, or block users of NOSS. Additionally, administrative users can change access levels of users.

[13] NASA Data Acquisition System (NDAS) Software User’s Guide and Developer Manual, October 2016

2. *User Interfaces*



Facility Information

Test Name	Status	Test Release:	Test Date:	Test Article Type:	Program:
1019	: Active	1	2016-11-09	RS25	0528

Measurement Summary

Total Measurements: 673	Active Measurements: 673	Inactive Measurements: 0
Measurements Without Sequences 0		

Open/In Work Actions Summary

All: 0	Test Critical: 0

Transducer Calibration Summary

Requires Calibration 29	Cal within a week: 1	Cal within a month: 7

Warning! This US Government system is for authorized users only. By accessing this system you are consenting to complete monitoring with no expectation of privacy. This system contains Sensitive But Unclassified (SBU) information. This information must be protected from unauthorized disclosure. Protective measures for marking, handling, and dissemination of this data can be found in NASA Interim Directive (NID): Sensitive But Unclassified (SBU) Controlled Information 1600.55. Unauthorized use, access to, or conveyance of information not in full compliance with NID 1600.55 may result in disciplinary action and criminal prosecution.

Figure 2. NOSS Main Dashboard

3. *NOSS Theory of Operation*

The NOSS application supports two main phases: test facility definition and test facility operation. During the test facility definition phase, users define all the nodes available on the test facility. For simple deployments, this will only be the components available to the NDAS acquisition system. For more complex deployments, this will be all the components on the entire test facility including other acquisition systems or control systems. An example of node definition is as follows:

A typical test facility may have 500 signal conditioners. These signal conditioners may be sub-divided as follows: 400 analog cards, 50 counter cards, and 50 digital cards. NOSS defines this as 500 different nodes (assuming single channel cards) total with three different types of nodes: one for analog, one for counter, and one for digitals. Each of these cards has different types of metadata to track and thus defined separately. Fields are added to each node type. For the analog card nodes, there may be a need for gain field, filter field, bridge type, and excitation source for example, to be added. Additionally, the physical location of the card (level 6, bay 32, rack 20) or the serial number of the card could be defined. The last phase of node definition is to define the constraints on the fields. Gain, for example, should be a specific set of positive numbers, and should not be negative or have fraction decimal parts.

After the node definition phase is complete and the entire universe of nodes is defined, users can utilize NOSS to define measurements, and generally configure and track all components on the test facility by linking nodes.

4. *NOSS Configuration Options*

The NOSS application allows a high degree of configuration for different types of instrumentation and acquisition hardware. As described above, the entire universe of nodes is defined in the initial setup phase of the application lifecycle. Test facilities will contains different types of hardware and systems, and NOSS allows highly customizable node definitions. These node definitions are defined as Extensible Markup Language (XML) schemas.

5. NOSS Architecture

NOSS is developed using the Yii Framework[14] in Hypertext Preprocessor (PHP$^{\text{TM}}$)[15]. There are various resources online that describe how to download, configure, and develop in the framework.

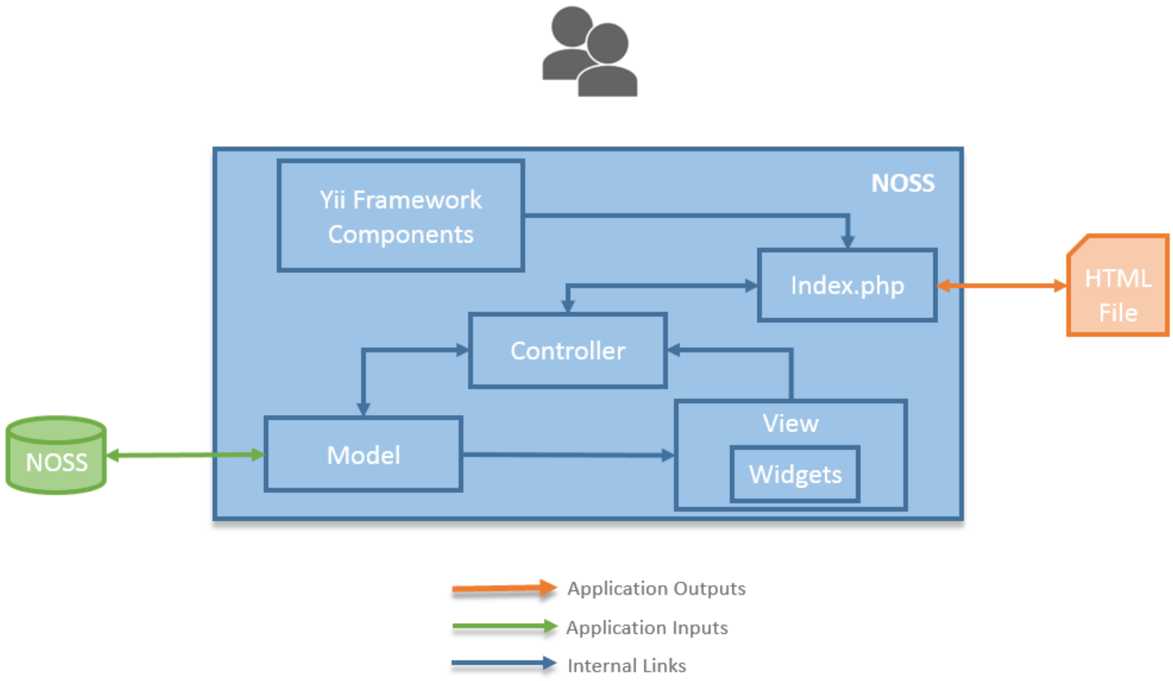


Figure 3. NOSS Block Diagram

B. NOPS[16]

The NOPS application is the main data-stream producer and control application for NDAS.

1. NOPS Major Functions

- **Turn on/off data stream** – Start or stop the NDAS data stream.
- **View current configuration** – View the current NIRD configuration.
- **Raw Counts to Volts Conversion** – For each measurement, raw count values are converted to voltages and added to the NDAS data stream when acquiring.
- **Volts to Engineering Unit Conversion** – For each measurement, the voltage values are converted to engineering units and added to the NDAS data steam when acquiring.
- **Derived measurement calculations** – Real-time derived measurements are calculations performed and added to the NDAS data stream when acquiring.
- **System-level marker monitoring** – There are two types of markers that can be defined in NDAS; digital and analog. Digital markers can be configured to trigger on low to high, high to low, or on either direction. Analog measurements can be configured to trigger greater than, less than, greater than or equal to, or less than or equal to a specific threshold.
- **Object Linking and Embedding (OLE) for Process Control[17] (OPC) Simulator**- Uses the measurements defined in an OPC server for channelization, and use the live data values from the OPC server as the data values for each measurement.
- **TDMS Simulator** – Uses the channels defined in a TDMS file for channelization, and the data values in the TDMS file.

[14] Reference http://www.yiiframework.com

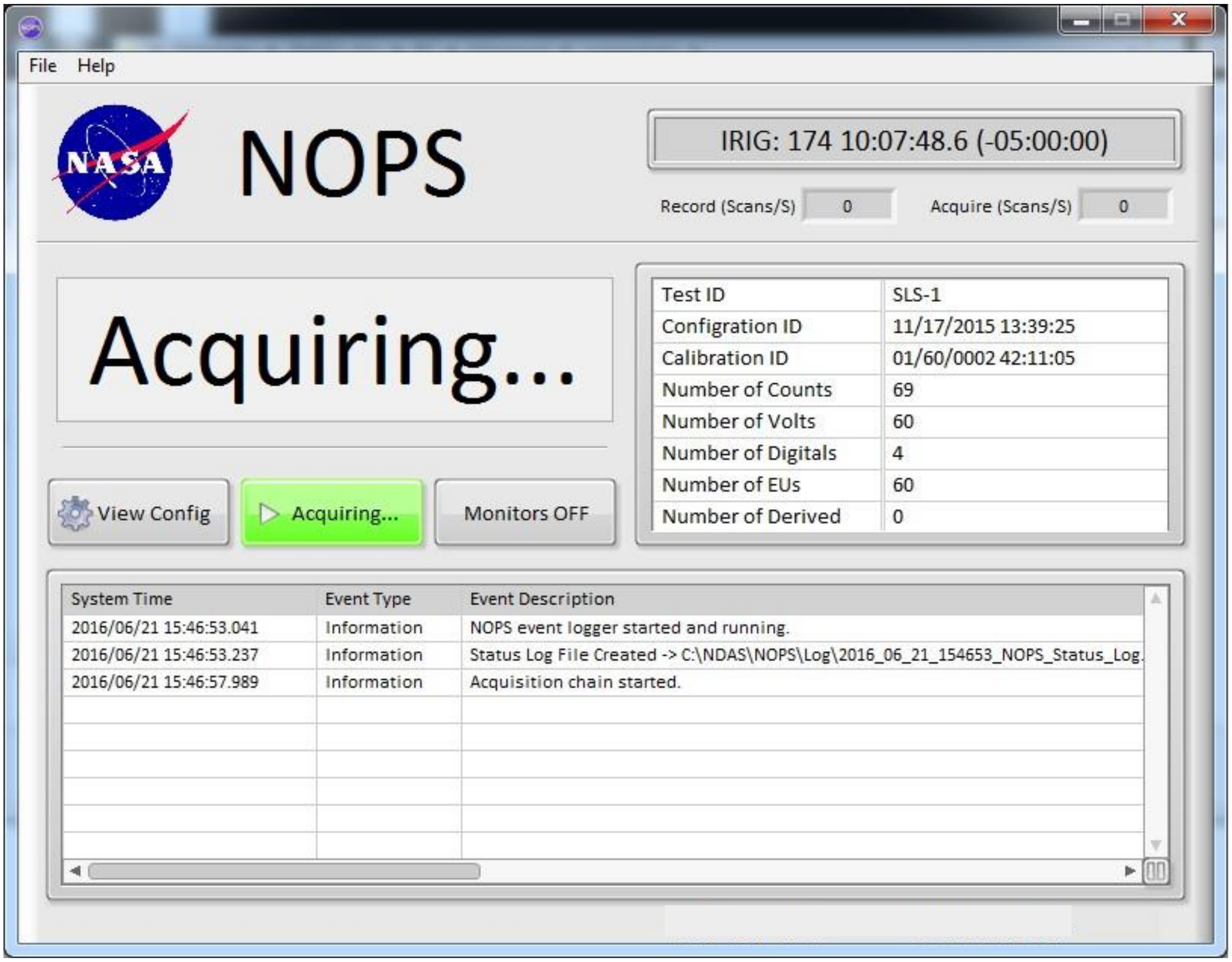
[15] PHP is a trademark held by PHP.net

[16] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

[17] Reference https://opcfoundation.org

- **Hardware Simulator** – Uses the current channels defined in NIRD for channelization, and simulate values for all defined measurements (sine wave, etc.).

2. *User Interfaces*



IRIG: 174 10:07:48.6 (-05:00:00)

Record (Scans/S) 0 Acquire (Scans/S) 0

Acquiring...

Test ID	SLS-1
Configration ID	11/17/2015 13:39:25
Calibration ID	01/60/0002 42:11:05
Number of Counts	69
Number of Volts	60
Number of Digitals	4
Number of EUs	60
Number of Derived	0

System Time	Event Type	Event Description
2016/06/21 15:46:53.041	Information	NOPS event logger started and running.
2016/06/21 15:46:53.237	Information	Status Log File Created -> C:\NDAS\NOPS\Log\2016_06_21_154653_NOPS_Status_Log.
2016/06/21 15:46:57.989	Information	Acquisition chain started.

Figure 4. NOPS Main User Interface

3. *Theory of Operation*

Operation of NOPS assumes the following preconditions have been met:

- All measurement system configuration has been defined in NOSS and imported into the NIRD database.
- All acquisition hardware has been installed correctly, powered up, and configured.
- The network is properly configured and is operating correctly.

After all the preconditions have been met, NOPS will run with little user interaction. The main operations that occur in NOPS are starting (acquiring) and stopping the data stream. NOPS provides the capability to stop the data stream to load a new NIRD configuration, troubleshoot hardware issues, or restart the acquisition hardware. Most of the time during normal operation, the NOPS application will be in the acquiring and broadcasting state. Users can view the current NIRD configuration or view the timing analysis of each stage of acquisition.

4. NOPS Architecture

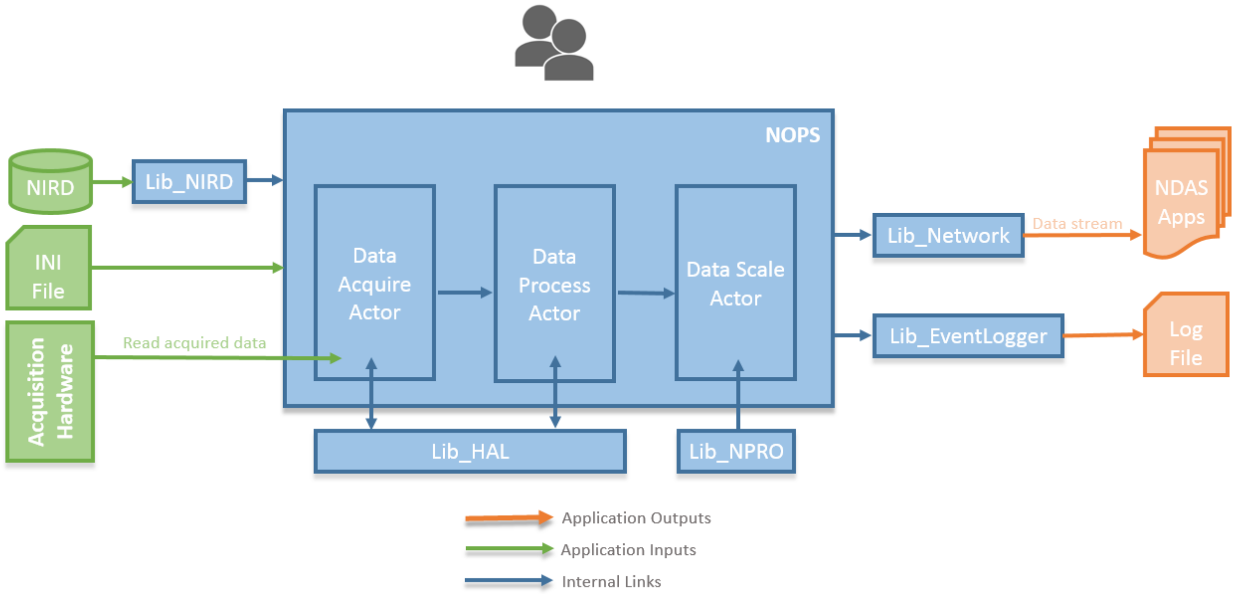


Figure 5. NOPS Architecture

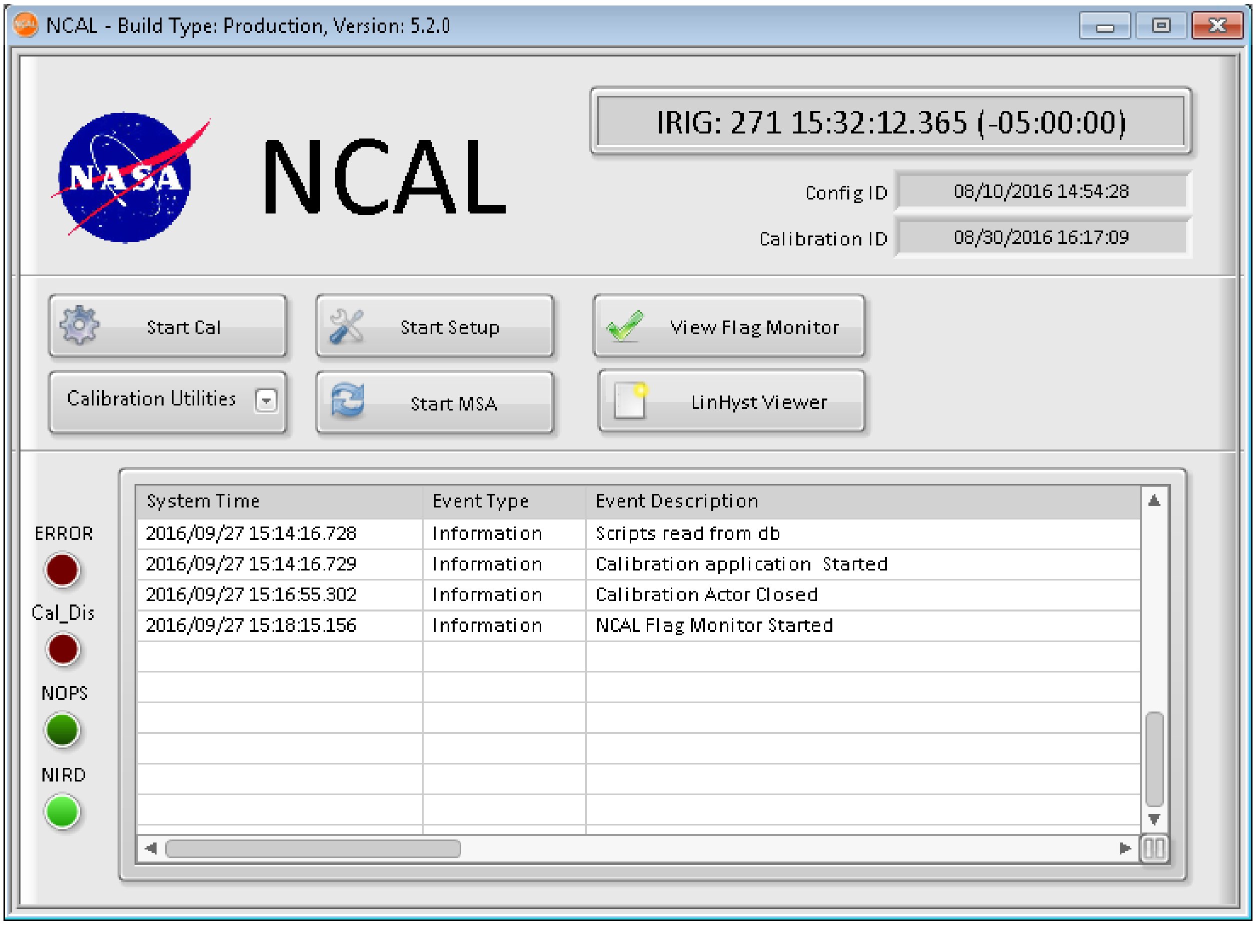
C. NCAL[18]

1. NCAL Major Functions

- **Auto Calibration:** Runs a pre-defined system electrical calibration:
  - Multiple, user selectable calibration schemes are available. Some examples are: Calibrate all, Linearity and Hysteresis, Calibrate Shunts, Calibrate TC / Barometer.
  - User defined calibration schemes can be developed via NOSS.
- **Manual Calibration:** Runs a calibration on individual channels or groups of channels.
- **Setup Panel:** Manually configures signal conditioning parameters and signal sources for advanced diagnostics, troubleshooting, and system checkout.
- **Measurement System Analysis (MSA):** Runs a series of system calibrations over a scheduled period for evaluation of DAS system stability and accuracy.
- **Calibration Editing:** View and edit calibration results. Also provides for updating NIRD database with latest coefficients.
- **Calibration Reporting:** View a report of the calibration, Linearity-Hysteresis, or the MSA calibrations results.

[18] NASA Data Acquisition System (NDAS) Software User’s Guide and Developer Manual, October 2016

2. User Interfaces



NCAL - Build Type: Production, Version: 5.2.0

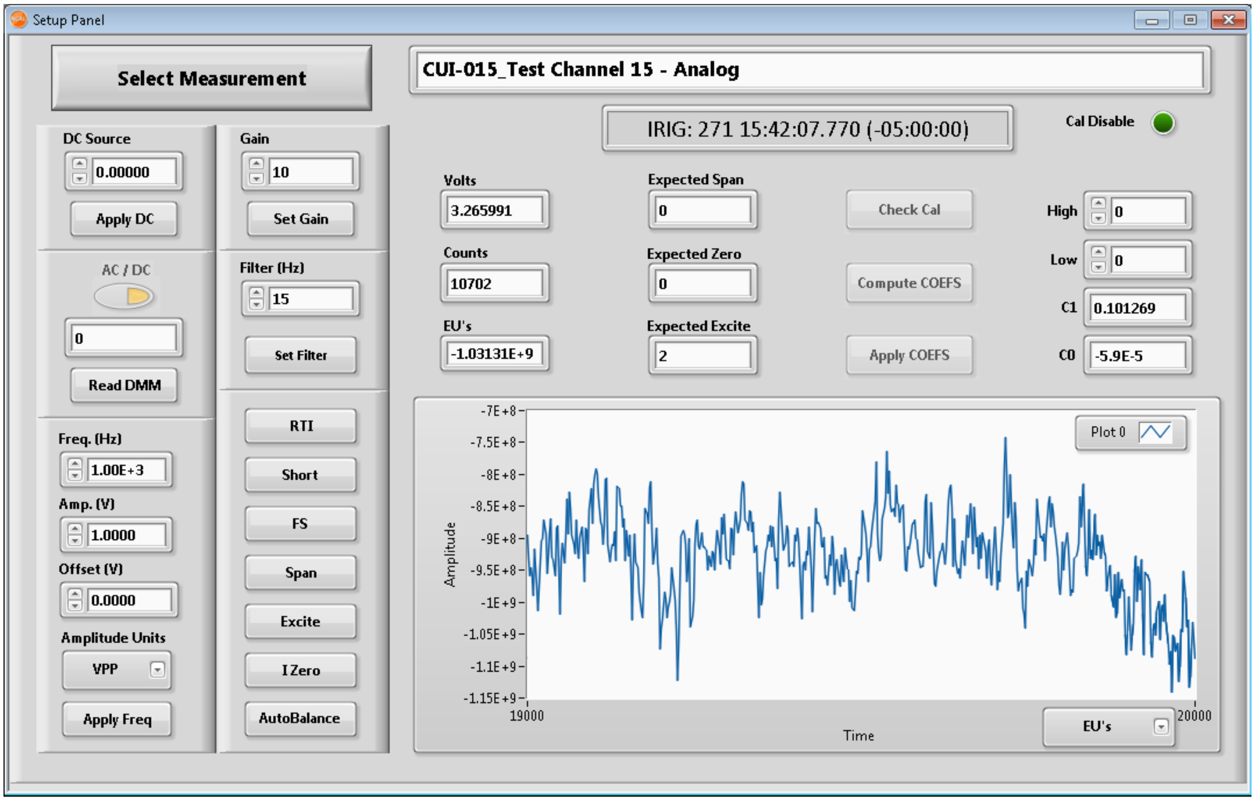
IRIG: 271 15:32:12.365 (-05:00:00)

Config ID 08/10/2016 14:54:28

Calibration ID 08/30/2016 16:17:09

System Time	Event Type	Event Description
2016/09/27 15:14:16.728	Information	Scripts read from db
2016/09/27 15:14:16.729	Information	Calibration application Started
2016/09/27 15:16:55.302	Information	Calibration Actor Closed
2016/09/27 15:18:15.156	Information	NCAL Flag Monitor Started

Figure 6. NCAL Main User Interface



Setup Panel

CUI-015_Test Channel 15 - Analog

IRIG: 271 15:42:07.770 (-05:00:00)

Cal Disable ●

DC Source: 0.00000

Gain: 10

AC / DC

0

Filter (Hz): 15

Freq. (Hz): 1.00E+3

Amp. (V): 1.0000

Offset (V): 0.0000

Amplitude Units: VPP

Volts: 3.265991

Counts: 10702

EU's: -1.03131E+9

Expected Span: 0

Expected Zero: 0

Expected Excite: 2

High: 0

Low: 0

C1: 0.101269

C0: -5.9E-5

Figure 7. Calibration Setup Panel

3. Theory of Operation

The NCAL application is used at regular intervals for electrical / bias check calibration of the measurement system.

NCAL is based on a user programmable state machine, which allows for National Institute of Standards and Technology (NIST) traceable calibration sources to be connected to the NDAS Data Acquisition hardware for system calibration.

Operation of NCAL assumes the following preconditions have been met:

- All measurement system configuration has been defined in NOSS and imported into the NIRD database.
- All acquisition and calibration hardware has been installed correctly and powered up.
- The network and GPIB bus are properly configured and is operating correctly.

After all the preconditions have been met, NCAL will run. Once started, NCAL configures all signal conditioning based on information in the NIRD database, set active input source to transducer, and perform any preconfigured auto-balance / auto-zero of enabled channels. After this task is complete, NCAL is ready for user interaction.

4. NCAL Architecture

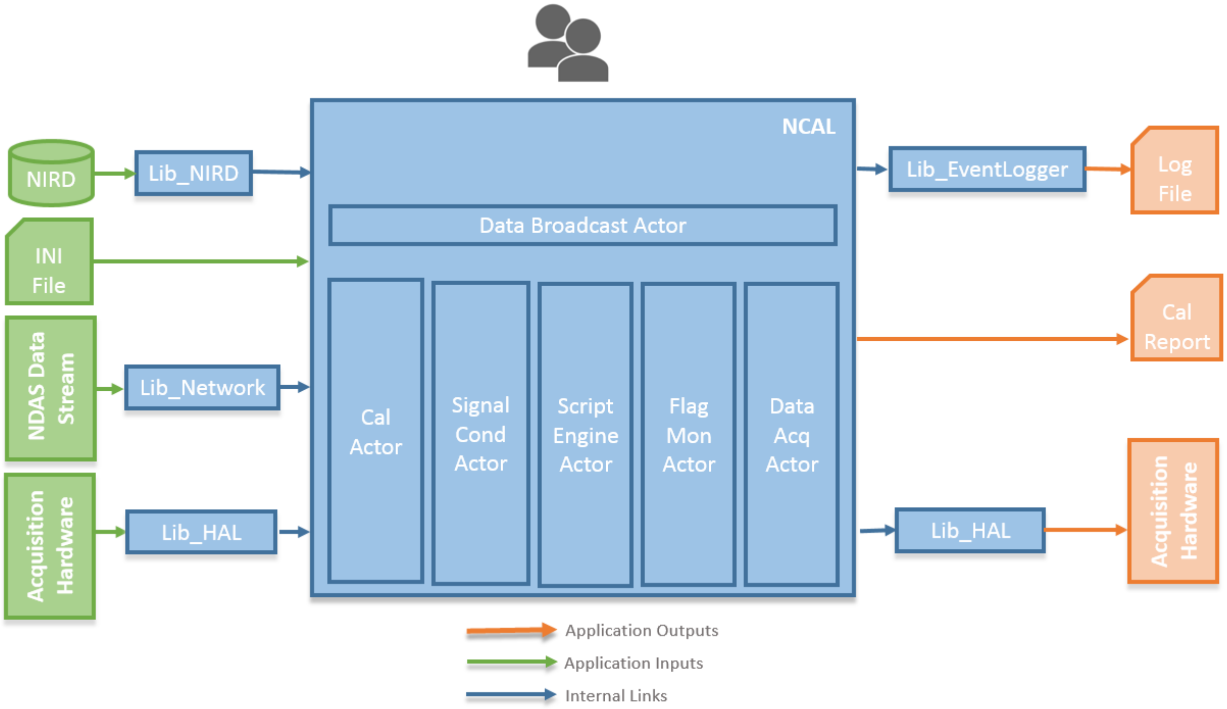


Figure 8. NCAL Architecture

D. NDIS[19]

NDIS is the real-time data visualization and display application for NDAS. It allows users to monitor measurements and system health during all phases of operations. NDIS can display multiple measurements from multiple data streams in the same interface.

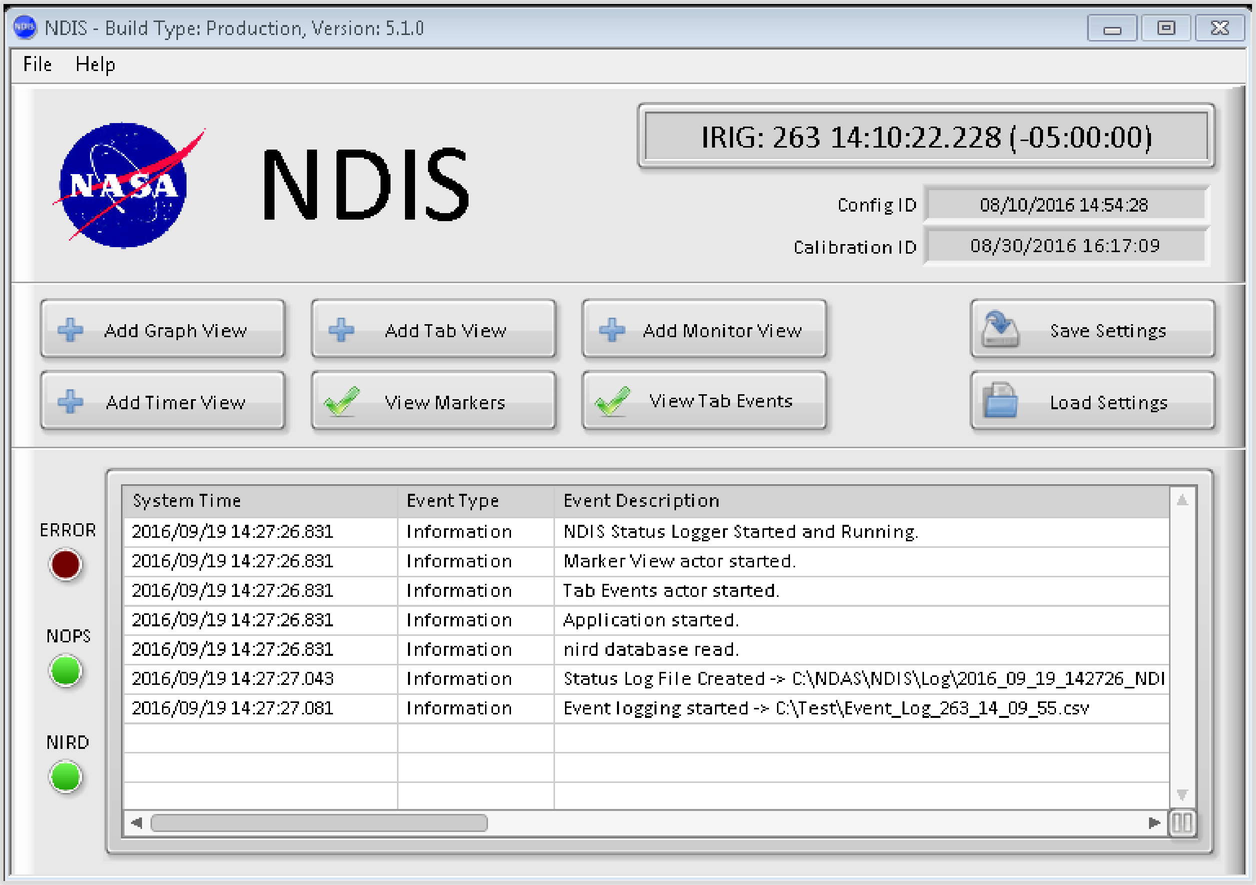
1. NDIS Major Functions

- **Graph data displays** – Plot measurements on a real-time graph, configurable by the user.
- **Tabular data displays** – Display measurements on tabular displays, configurable by the user.
- **Timer data displays** – Set timers based on events in the system.
- **Marker data displays** – View triggered markers on measurements real-time.
- **Events data displays** – View and record events as they occur in the system.

[19] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

2. User Interfaces

The figures below show samples of displays available in NDIS component of NDAS. A Main NDIS User Interface, graphical and tabular displays of measurements, as well as other types of displays is available:



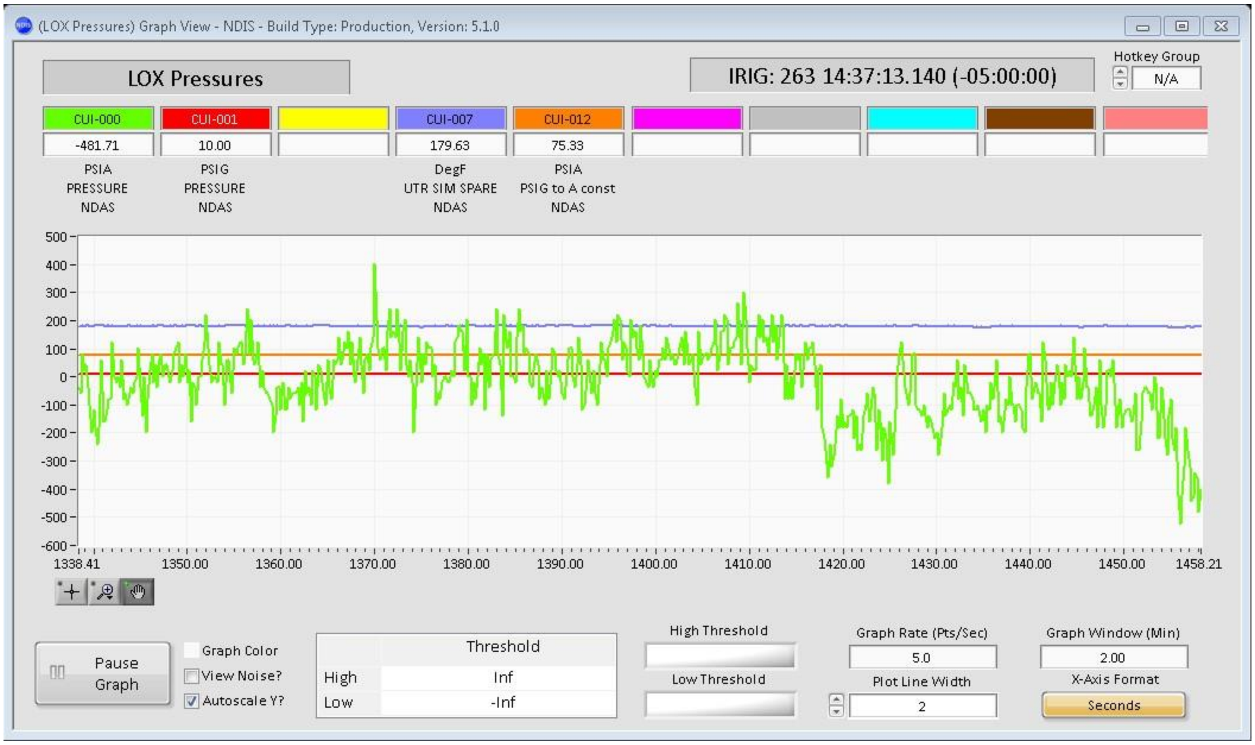
IRIG: 263 14:10:22.228 (-05:00:00)

Config ID 08/10/2016 14:54:28

Calibration ID 08/30/2016 16:17:09

System Time	Event Type	Event Description
2016/09/19 14:27:26.831	Information	NDIS Status Logger Started and Running.
2016/09/19 14:27:26.831	Information	Marker View actor started.
2016/09/19 14:27:26.831	Information	Tab Events actor started.
2016/09/19 14:27:26.831	Information	Application started.
2016/09/19 14:27:26.831	Information	nird database read.
2016/09/19 14:27:27.043	Information	Status Log File Created -> C:\NDAS\NDIS\Log\2016_09_19_142726_NDI
2016/09/19 14:27:27.081	Information	Event logging started -> C:\Test\Event_Log_263_14_09_55.csv

Figure 9. NDIS Main User Interface



(LOX Pressures) Graph View - NDIS - Build Type: Production, Version: 5.1.0

LOX Pressures

IRIG: 263 14:37:13.140 (-05:00:00)

Hotkey Group: N/A

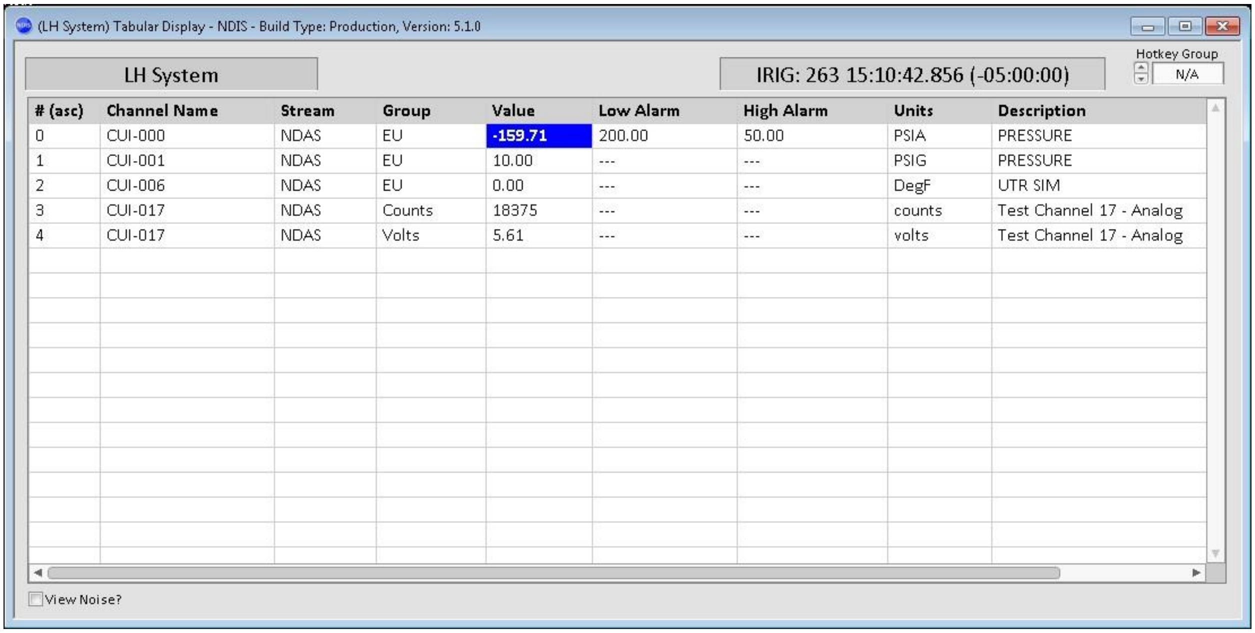
CUI-000	CUI-001	CUI-007	CUI-012
-481.71	10.00	179.63	75.33
PSIA	PSIG	DegF	PSIA
PRESSURE	PRESSURE	UTR SIM SPARE	PSIG to A const
NDAS	NDAS	NDAS	NDAS

□ Graph Color
□ View Noise?
☑ Autoscale Y?

Threshold	
High	Inf
Low	-Inf

High Threshold:
Low Threshold:
Graph Rate (Pts/Sec): 5.0
Plot Line Width: 2
Graph Window (Min): 2.00
X-Axis Format: Seconds

Figure 10. NDIS Graphical Display



(LH System) Tabular Display - NDIS - Build Type: Production, Version: 5.1.0

LH System

IRIG: 263 15:10:42.856 (-05:00:00)

Hotkey Group: N/A

# (asc)	Channel Name	Stream	Group	Value	Low Alarm	High Alarm	Units	Description
0	CUI-000	NDAS	EU	-159.71	200.00	50.00	PSIA	PRESSURE
1	CUI-001	NDAS	EU	10.00	---	---	PSIG	PRESSURE
2	CUI-006	NDAS	EU	0.00	---	---	DegF	UTR SIM
3	CUI-017	NDAS	Counts	18375	---	---	counts	Test Channel 17 - Analog
4	CUI-017	NDAS	Volts	5.61	---	---	volts	Test Channel 17 - Analog

□ View Noise?

Figure 11. NDIS Tabular Display

3. *Theory of Operation*

The NDIS application allows users to determine what types of data view(s) are best for the task. Related measurements can be grouped on the same graph or tabular view. The display configuration can be saved, allowing users to return to the same configuration later or allow other users to share the configuration.

Views can be added or removed at any time, and new measurements can be added to the views as needed.

4. NDIS Architecture

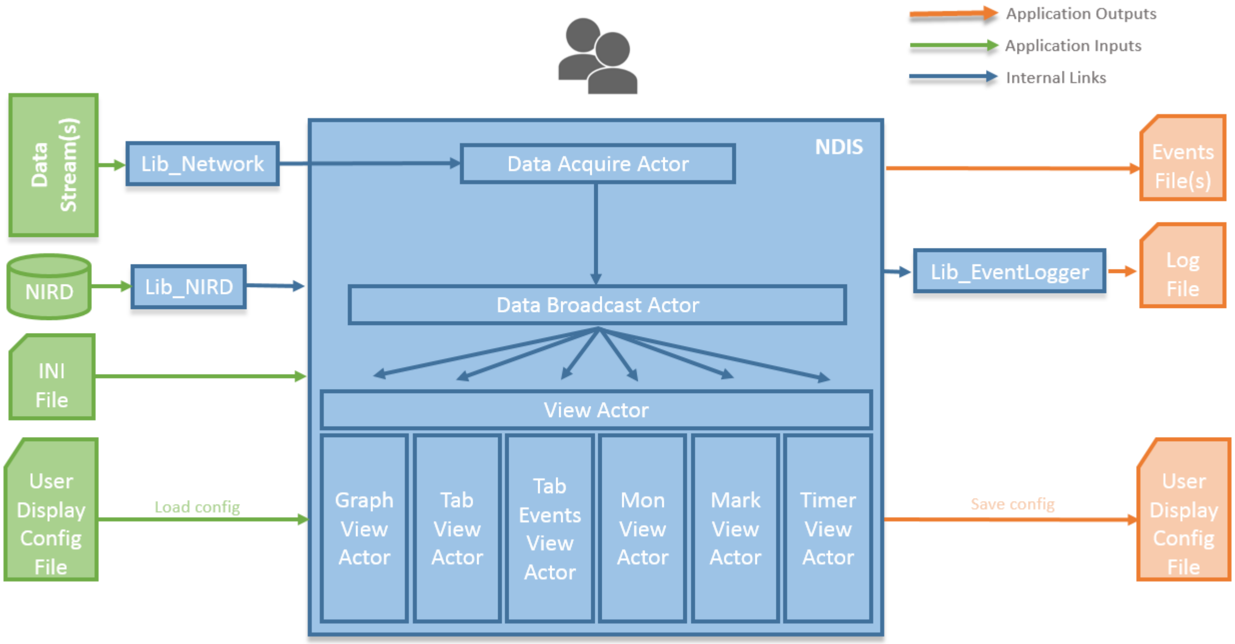


Figure 12. NDIS Architecture

E. NLOG[20]

The NLOG application serves as the data recorder for the NDAS system. The data files are used post-test for data analysis purposes. NLOG can record multiple data streams concurrently.

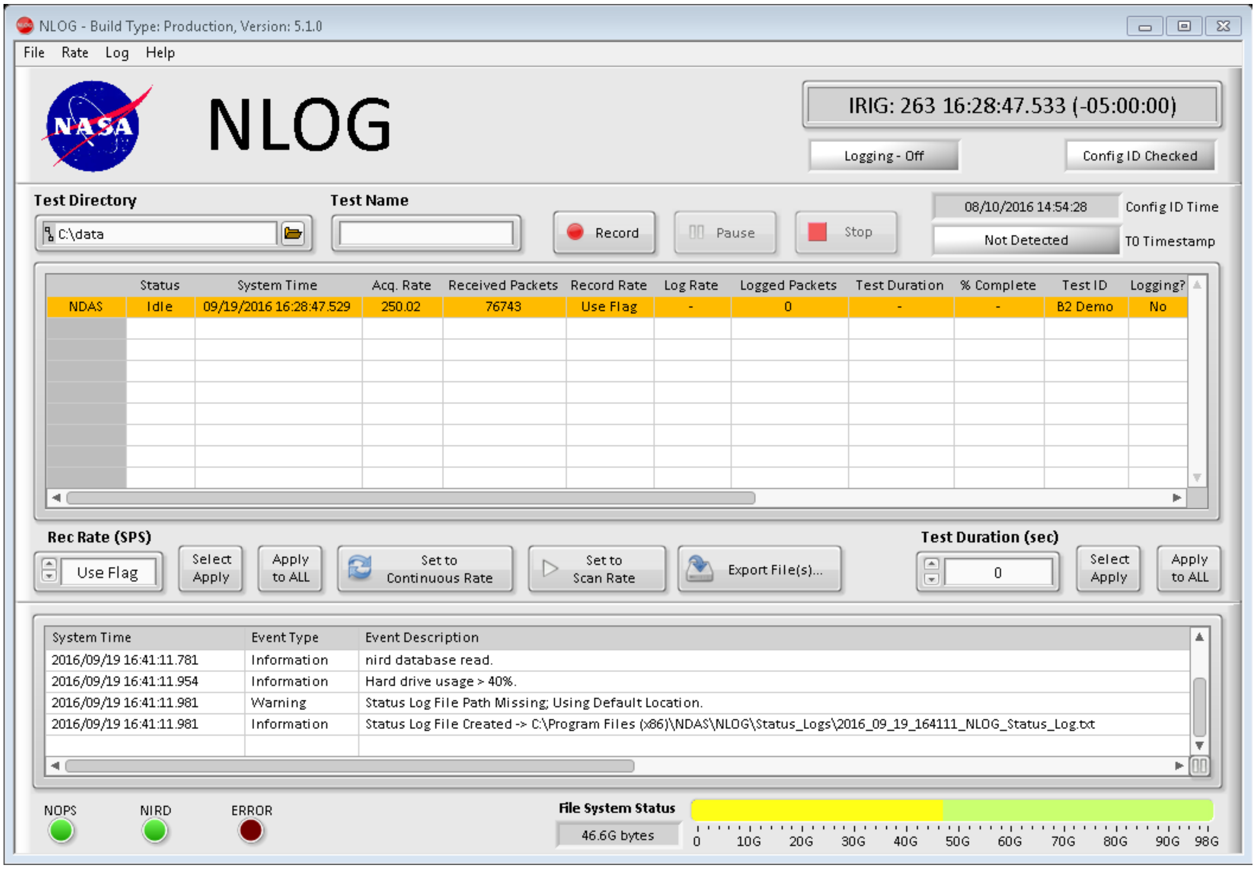
1. NDIS Major Functions

- **Logging data to file** – Records data to a file.
- **Changing recording rates** – Dynamically changes record rates throughout the recording cycle.
- **Surveillance logging** – Set up recorder that runs on a 24-hour basis with a configurable circular buffer.
- **Fixed Duration Logging** – Set a fixed-length of time to record the data streams.
- **Data file format conversion** – Convert the native TDMS file format to other file formats such as WinPlot, Comma Separated Values (CSV), Hierarchical Data Format-5 (HDF5)[21], or MatLAB®.

[20] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

[21] Reference https://support.hdfgroup.org/HDF5

2. *User Interfaces*



IRIG: 263 16:28:47.533 (-05:00:00)

Test Directory: C:\data

Test Name:

Config ID Time: 08/10/2016 14:54:28

T0 Timestamp: Not Detected

	Status	System Time	Acq. Rate	Received Packets	Record Rate	Log Rate	Logged Packets	Test Duration	% Complete	Test ID	Logging?
NDAS	Idle	09/19/2016 16:28:47.529	250.02	76743	Use Flag	-	0	-	-	B2 Demo	No

Rec Rate (SPS): Use Flag

Test Duration (sec): 0

System Time	Event Type	Event Description
2016/09/19 16:41:11.781	Information	nird database read.
2016/09/19 16:41:11.954	Information	Hard drive usage > 40%.
2016/09/19 16:41:11.981	Warning	Status Log File Path Missing; Using Default Location.
2016/09/19 16:41:11.981	Information	Status Log File Created -> C:\Program Files (x86)\NDAS\NLOG\Status_Logs\2016_09_19_164111_NLOG_Status_Log.txt

File System Status: 46.6G bytes

Figure 13. NLOG Main User Interface

3. *Theory of Operation*

The NLOG application can be utilized in all phases of system development. Generally, there multiple instances of the logger set up across the network. One instance is configured for surveillance mode at a low sample rate. This logger is useful for troubleshooting and it can be used to investigate anomalies that occur during off hours when no one is actively monitoring the system.

During active testing, another logger is used actively to record various pre-test activities and used at full sample rate during a test to capture all data. Post-test data files are converted to other formats for analysis, most commonly WinPlot.

4. NLOG Architecture

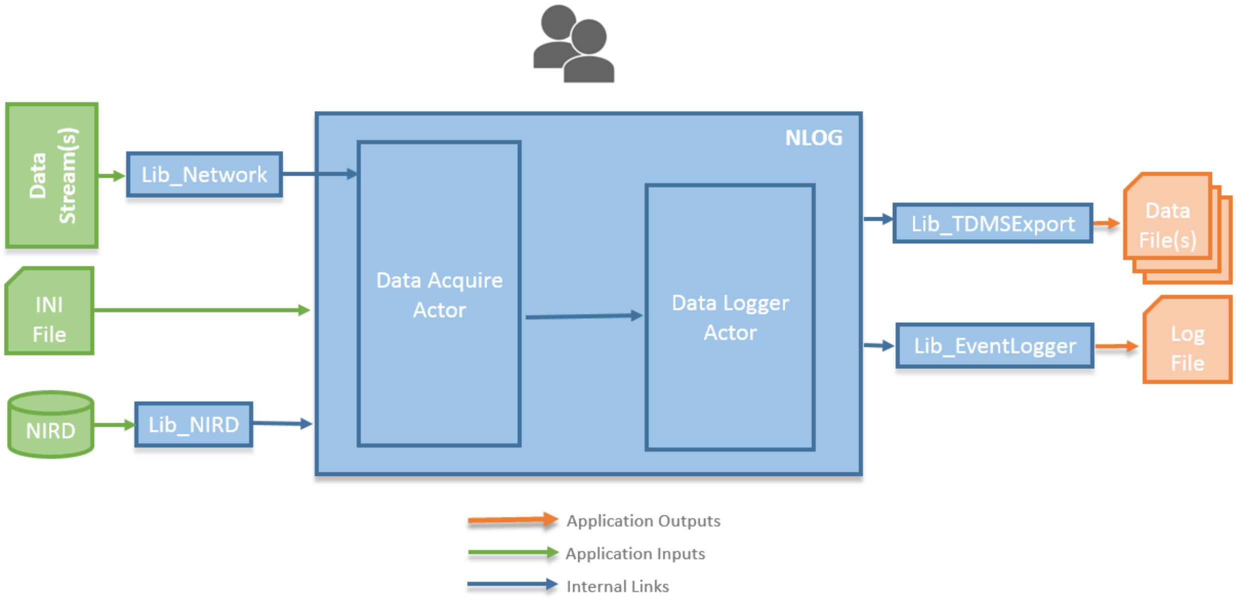


Figure 14. NLOG Architecture

F. NGATE[22]

NGATE is a set of applications that serve as the NDAS system gateway to/from other data producing or data consuming systems. This application is tailored to specific system interfaces and there are currently three versions: NGATE OLE for Process Control (OPC), NGATE Data Exchange Message (DEM), and NGATE FastPac.

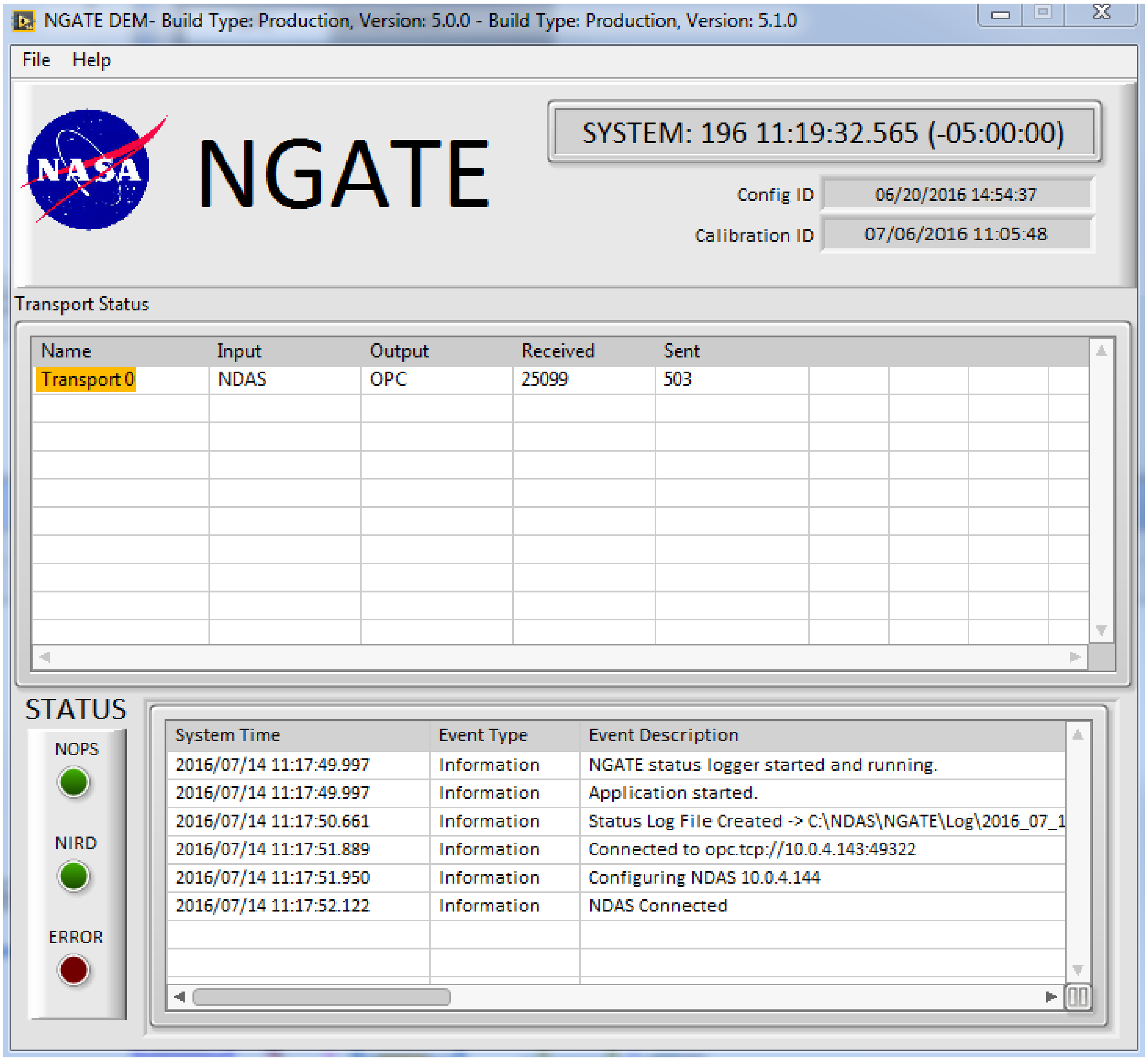
1. NDIS Major Functions

The NGATE application performs the following functions:

- **Data Stream Conversion** – Converts the NDAS data stream format to/from other formats including OPC, FastPac, and DEM.
- **Data stream Measurement Filtering** – Filter measurements in the data stream that are not required.
- **Data Stream Rate Limiting** – Adjusts the data rate for outgoing data streams.
- **Create Legacy Stennis Data Acquisition System (SDAS) Setup and Configuration Files** – This only applies to the NGATE FastPac application.

[22] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

2. User Interfaces


NGATE DEM- Build Type: Production, Version: 5.0.0 - Build Type: Production, Version: 5.1.0
File
Help
NASA
NGATE
STATUS
NOPS
NIRD
ERROR

SYSTEM: 196 11:19:32.565 (-05:00:00)

Config ID 06/20/2016 14:54:37

Calibration ID 07/06/2016 11:05:48

Transport Status

Name	Input	Output	Received	Sent
Transport 0	NDAS	OPC	25099	503

System Time	Event Type	Event Description
2016/07/14 11:17:49.997	Information	NGATE status logger started and running.
2016/07/14 11:17:49.997	Information	Application started.
2016/07/14 11:17:50.661	Information	Status Log File Created -> C:\NDAS\NGATE\Log\2016_07_1
2016/07/14 11:17:51.889	Information	Connected to opc.tcp://10.0.4.143:49322
2016/07/14 11:17:51.950	Information	Configuring NDAS 10.0.4.144
2016/07/14 11:17:52.122	Information	NDAS Connected

Figure 15. NGATE Main User Interface

The NGATE interface is minimalistic because of its intent to run in the background with minimal user interaction.

3. Theory of Operation

NOPS module provides the NDAS broadcast data stream as it acquires data from the hardware. This data stream is in the NDAS native format as used by all NDAS modules. Frequently NASA customers/partners have software programs that can also read data from a broadcast stream; however, the format of the stream is not the same as NDAS. The NGATE module provides the infrastructure to help with those situations. Given the new stream format, the new stream converter can be built and installed.

4. NGATE Architecture

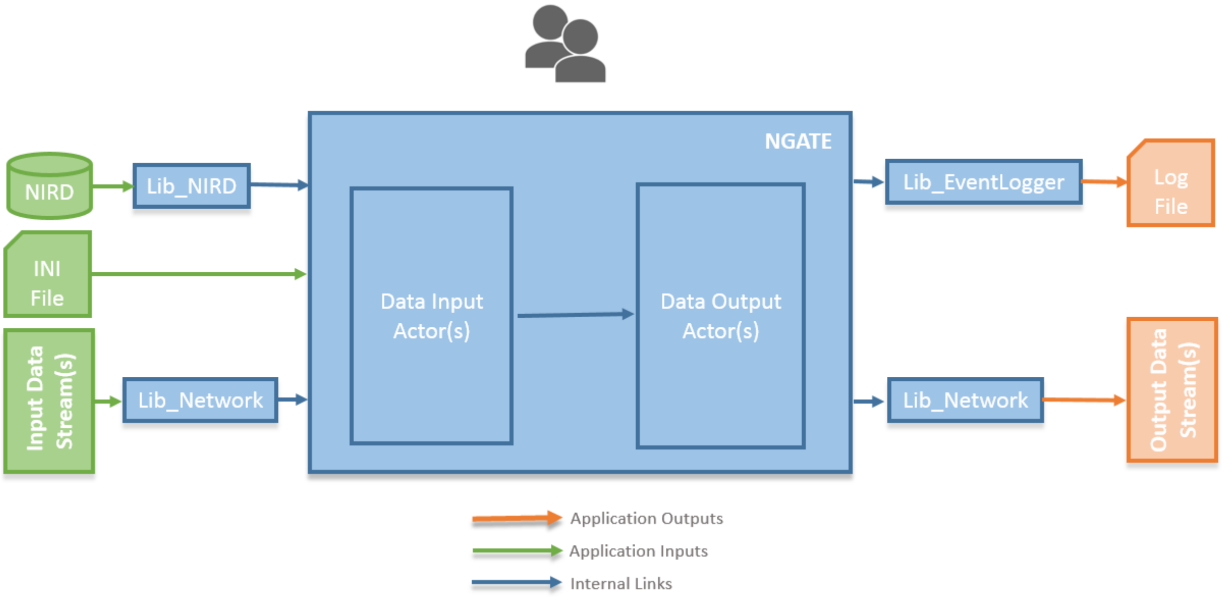


Figure 16. NGATE Architecture

G. TDMS Editor[23]

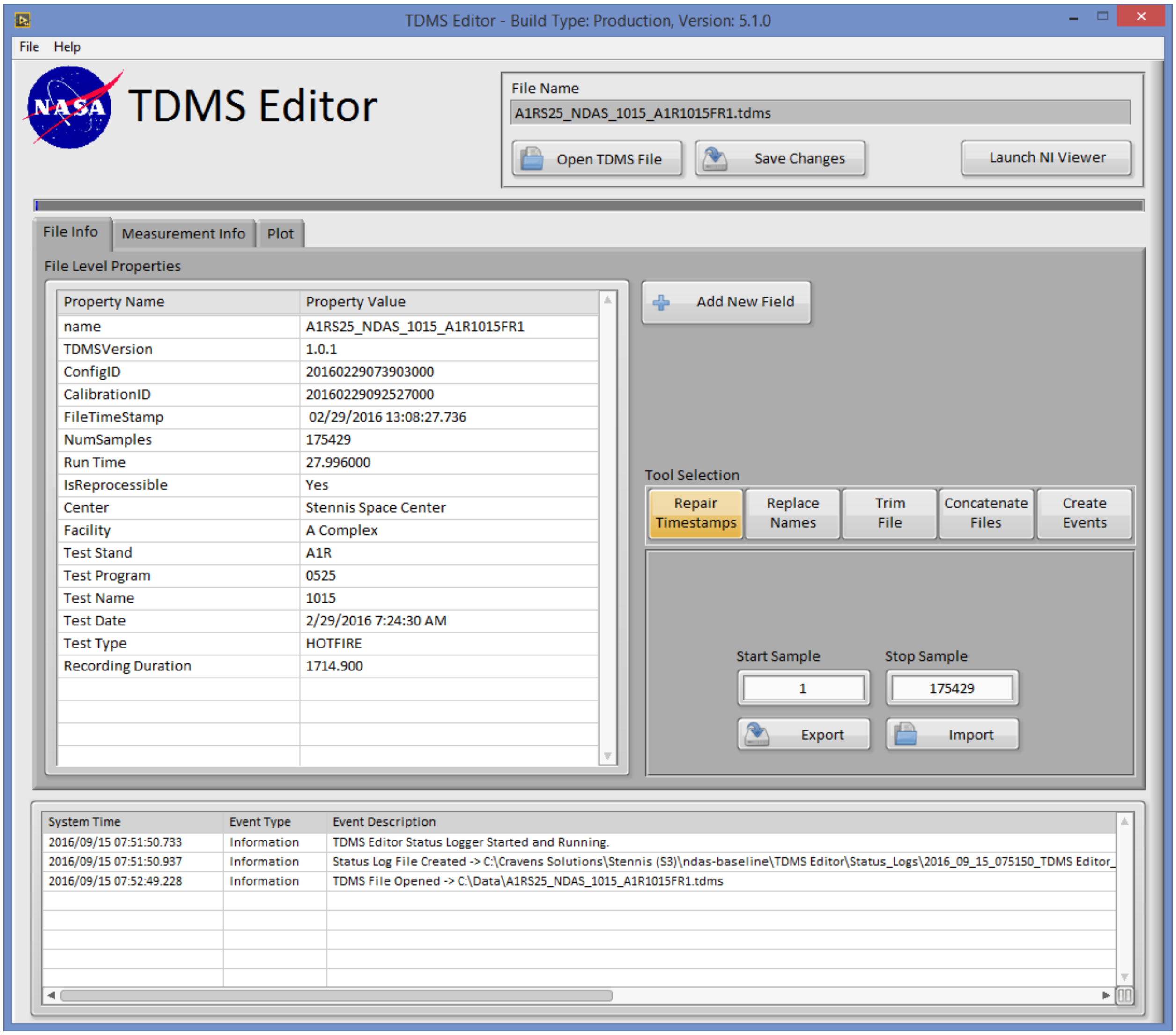
The TDMS Editor application is a post-processing tool used to review and edit TDMS files.

1. TDMS Editor Major Functions

- **View measurements in the TDMS File** – View measurement plots, measurement data in tabular form, and all metadata associated with all measurements.
- **Add and update metadata** – File-level (or test-level) metadata and measurement-level metadata can be edited and additional fields can be added.
- **Reprocess Measurements** – Update scaling coefficients for measurements and reprocess the data for that measurement with the new coefficients.
- **Trim TDMS files** – Trim sections of the original TDMS file.
- **Concatenate TDMS file** – Combine two TDMS files.

[23] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

2. *User Interfaces*



File Name: A1RS25_NDAS_1015_A1R1015FR1.tdms

Property Name	Property Value
name	A1RS25_NDAS_1015_A1R1015FR1
TDMSVersion	1.0.1
ConfigID	20160229073903000
CalibrationID	20160229092527000
FileTimeStamp	02/29/2016 13:08:27.736
NumSamples	175429
Run Time	27.996000
IsReprocessible	Yes
Center	Stennis Space Center
Facility	A Complex
Test Stand	A1R
Test Program	0525
Test Name	1015
Test Date	2/29/2016 7:24:30 AM
Test Type	HOTFIRE
Recording Duration	1714.900

Start Sample: 1

Stop Sample: 175429

```
System Time              Event Type    Event Description
2016/09/15 07:51:50.733  Information   TDMS Editor Status Logger Started and Running.
2016/09/15 07:51:50.937  Information   Status Log File Created -> C:\Cravens Solutions\Stennis (S3)\ndas-baseline\TDMS Editor\Status_Logs\2016_09_15_075150_TDMS Editor_
2016/09/15 07:52:49.228  Information   TDMS File Opened -> C:\Data\A1RS25_NDAS_1015_A1R1015FR1.tdms
```

Figure 17. TDMS Editor Main User Interface

3. *Theory of Operation*

The TDMS Editor is generally used post-test to correct or update measurements and generate special data files for analysts needs.

VI. NDAS Technology Stack

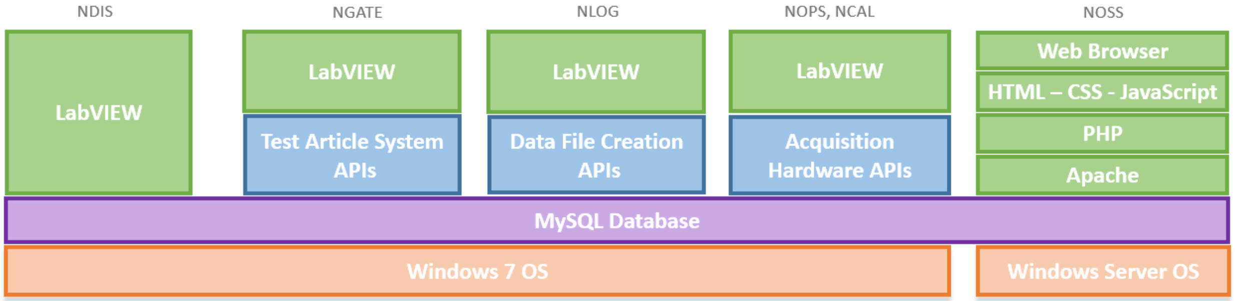


Figure 18. NDAS Technology Stack[24]

While primarily written in LabVIEW™, a graphical, data-flow driven language where program execution follows ‘data flow’ along wires, NDAS, as depicted in Fig. 18, incorporates numerous technologies. NDAS runs on the Windows 7®[25] operating system. NOPS, NDIS, NCAL, NGATE, and NLOG are written in LabVIEW™. The underlying NIRD database is defined using MySQL™. NOSS combines a multitude of technologies, including a web browser, Hyper-Text Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript®[26], Apache™[27], PHP™.

VII. Interaction with Downstream Facility Systems

As shown in Fig. 19, NDAS interfaces with downstream facility systems such as the Facility Control and Redline Systems and the High Speed DAS. At SSC’s A-1 Test Facility, these systems typically used a proprietary file format from the SDAS called RDCARD, which was generated by the Instrumentation Roadmap System (IRS) database. This database was used with the legacy SDAS software Suite. Because NOSS is now implemented at this facility, IRS no longer creates this file. NOSS now generates an XML file for use by the downstream applications. The fact that NOSS does not generate this file necessitated modification of these downstream applications. This change was implemented and used on an RS-25 Test during the summer of 2016. In addition, NGATE was modified to generate a Measurement System Identifier (MSI) file in order to transmit data to SSC’s Integrated Advisory System (IAS), which transmits live test data to off-site customers. For implementation at SSC’s B-2 Test Facility, these applications have yet to be

[24] NASA Data Acquisition System (NDAS) Software User’s Guide and Developer Manual, October 2016

[25] Windows 7® is a trademark of the Microsoft Corporation. Neither NASA Stennis Space Center, nor any software programs or other goods or services offered by NASA Stennis Space Center, are affiliated with, endorsed by, or sponsored by the Microsoft Corporation.

[26] JavaScript® is a trademark of Oracle Corporation. Neither NASA Stennis Space Center, nor any software programs or other goods or services offered by NASA Stennis Space Center, are affiliated with, endorsed by, or sponsored by Oracle Corporation.

[27] Apache, Apache Foo, and Foo are trademarks of The Apache Software Foundation. Used with permission. No endorsement by The Apache Software Foundation is implied by the use of these marks.

developed; therefore, these applications can directly use the XML file generated by NOSS. Functionality to transmit live test-data to off-site customers is still being evaluated for the B-2 Test Facility.

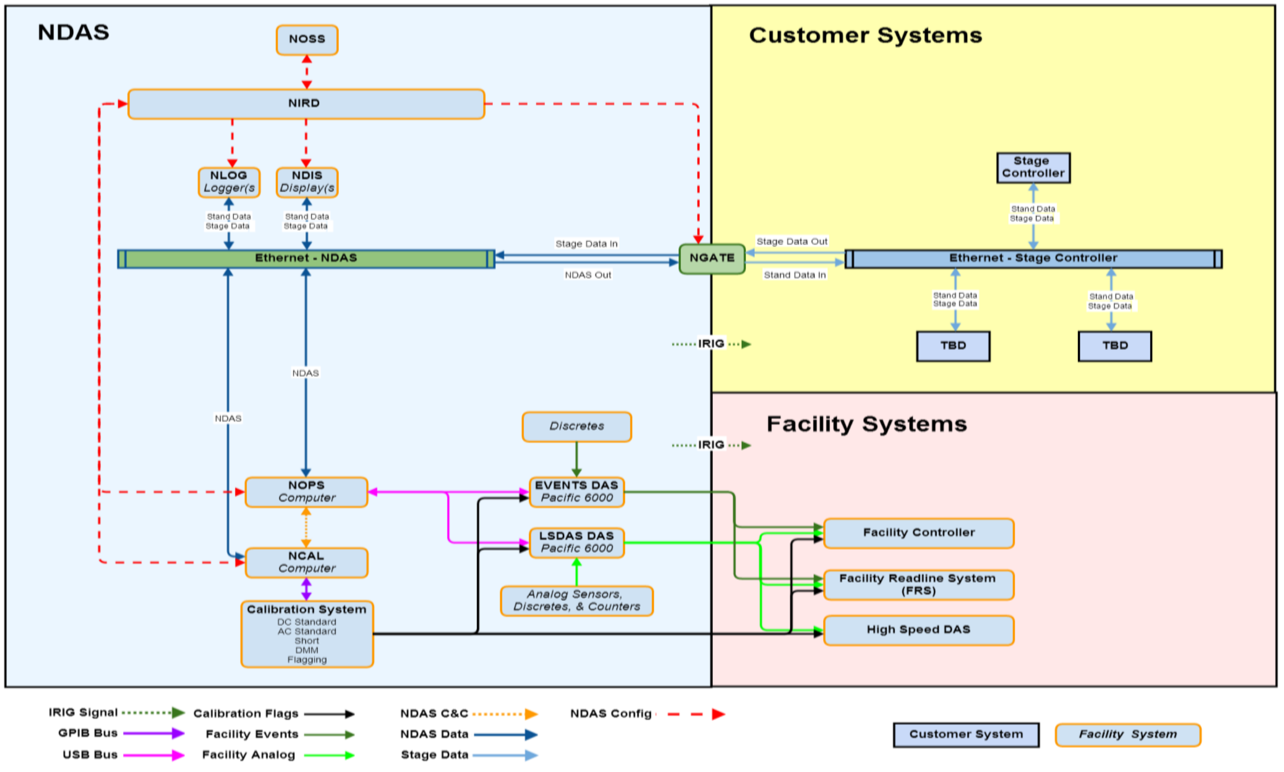


Figure 19. NDAS Interfaces

VIII. Implementation at the B-2 Test Facility

Extensive modifications to SSC's B-2 Test Facility are being made to support the SLS Core Stage Green Run, the first time the engines are assembled into a single configuration with the core stage and fired at nearly full power. These modifications include development and implementation of a completely new DAS. This new DAS project selected the Pacific Instruments 6000 family of data acquisition hardware, including both low and high-speed DAS equipment. This selection necessitated the modification of NDAS to support this new equipment. The NDAS team has extended the support / functionality of the HAL to support the full set of new DAS hardware. The NDAS team is currently testing these changes in preparation for use at the B-2 Test Facility to support activation of the facility mechanical systems.

IX. Use at SSC and other NASA Centers

Several rocket engine hot-fire tests at two different test facilities at SSC, from March of 2014 through present day, have successfully used NDAS as the primary DAS software suite. The NDAS data acquired from these tests is currently being used to certify rocket engines for space flight, supporting NASA's Exploration Mission-1 (EM-1)[28]. In February of 2017, modifications for use at SSC's E-Complex are scheduled to begin, allowing other facilities to potentially accept new features. SSC has also begun discussions with other NASA Centers and facilities to address implementation at these locations. Specifically, SSC has begun discussions with GRC and WSTF.

X. Application Configuration Management

Because of its applicability on multiple test facilities, including potential use at other RPT centers, the NDAS software suite exists currently in both development and maintenance states. A released set of software is in use on A-

[28] Reference https://www.nasaspaceflight.com/tag/em-1/

1 while concurrent development and testing is being performed for B-2. Development work for the next release is carried out in a repository provided by Atlassian®, called Bitbucket®. New features and bug fixes are its own branch within the repository. All updates and bug-fix tickets are regression tested to ensure compatibility across all supported areas. Once a new version is released, system owners can choose to install the latest update with the new features, which may or may not originate from other facilities (Fig. 20). Selected components will be unique to areas, for example, NGATE FastPac is used only in the A-Complex. Such applications will be deployed only to necessary systems, but will all reside in the unified master branch. This strategy will also be used when NDAS is deployed at other SSC test facilities as well as other RPT centers.

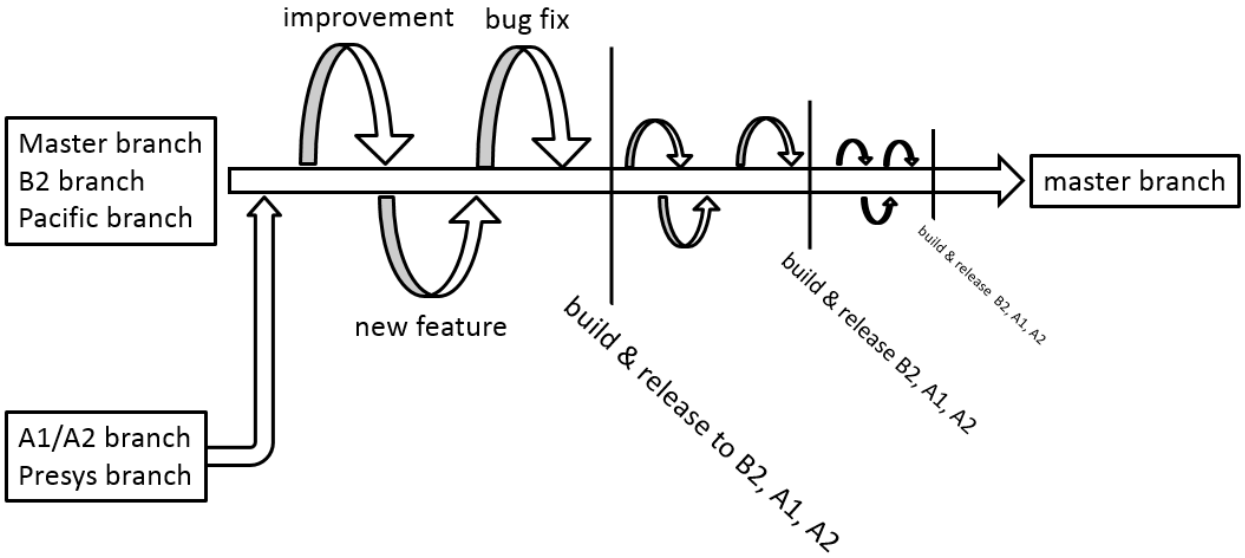


Figure 20. NDAS CM Common Branch for Maintenance[29]

XI. Summary

Over the past five years, NASA SSC has been developing the NDAS software suite to support rocket propulsion testing. The NDAS design provides portability to multiple test facilities not only across SSC, but at other NASA centers as well. NDAS has potential uses at MSFC, Plum Brook Station, WSTF, GRC, and possibly KSC.

One of the original goals for the software was to develop a common data acquisition software suite for NASA propulsion test centers. In addition, the software has the potential to be transferred to commercial organizations and other government agencies involved in any data acquisition endeavor. Once the software release requirements, evaluations, and approvals are completed, the NDAS software suite could become available to other government agencies, corporations, universities, and the general United States public.

Another goal for NDAS is to provide NASA with non-proprietary DAS software for use at all its test facilities. With the deployment of NDAS at the B-2 Test Facility, there will be an instance of NDAS that does not use proprietary software, making this goal close to being met.

The NDAS software suite is optimal for large scale, multiuser data acquisition systems with lower-speed (250 samples per second or below) acquisition requirements. NDAS uses a translation layer in order to accomplish this portability to multiple facilities. The NDAS team also uses widely available software development tools such as Atlassian JIRA®, Stash™, and Confluence® to manage software development, the LabVIEW™ Integrated Development Environment (IDE) for code development, and MySQL™ databases to store data and configurations.

Rocket propulsion tests at SSC's A-1 and A-2 test facilities have successfully used NDAS. These facilities now have updated downstream applications that no longer rely on the use of proprietary data files. SSC is currently deploying this software at the B-2 Test Facility and are in discussions with other NASA centers to implement the NDAS software suite. Successful deployment at these other NASA field centers will further NASA's goal of

[29] John C. Stennis Space Center Test Complex Software Management Plan, SMP-000002, Revision C, September 2016, Paragraph 5.14.6.

increasing affordability for spaceflight, since a majority of the software will be re-used for the purpose of acquisition and distribution of data. This promotes opportunities for inter-agency collaboration.

References

Publications

[1] AIAA-2015-2561. NASA Data Acquisition System Software Development for Rocket Propulsion Test Facilities, AIAA Aviation and Aeronautics Forum and Exposition 2015

Manuals

[2] NASA Data Acquisition System (NDAS) Software User's Guide and Developer Manual, October 2016

[3] John C. Stennis Space Center Test Complex Software Management Plan, SMP-000002, Revision C, September 2016.