Defence Research and
Development Canada

Recherche et développement
pour la défense Canada

APPENDIX A: Q302 Trial Software Description Report

Sonar test Bed Support for DRDC Atlantic Sea Trail Q302

Chris Wheeler

General Dynamics Canada Ltd
3785 Richmond Road
Ottawa, Ontario
K2H 5B7

PWGSC Contract Number: W7707-063585/A
CSA: Jim Theriault: 902-426-3100 ext 376

Defence R&D Canada - Atlantic

Contract Report
DRDC Atlantic CR 2007-064
August 2007

APPENDIX A: Q302 Trial Software Description Report

Sonar test Bed Support for DRDC Atlantic Sea Trail Q302

Chris Wheeler

General Dynamics Canada Ltd
3785 Richmond Road
Ottawa, Ontario
K2H 5B7

PWGSC Contract Number: W7707-063585/A
CSA: Jim Theriault: 902-426-3100 ext 376

The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of the Department of National Defence of Canada.

Defence Research and Development Canada – Atlantic

Contract Report
DRDC Atlantic CR 2007-064
August 2007

IMPORTANT INFORMATIVE STATEMENTS

The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of the Department of National Defence of Canada.

© Her Majesty the Queen in Right of Canada, as represented by the Minister of National Defence, 2007

© Sa Majesté la Reine (en droit du Canada), telle que représentée par le ministre de la Défense nationale, 2007

Table of Contents

List of Figures

1. Introduction

1.1 Overview

This report summarizes the hardware configuration and describes the software functionality, the software components, the shell scripts, the configuration files, and the procedures used to provide the dry-end active and passive sonar system functionality for the DASM array, the MANTArray, and sonobuoys during the trial Q302 onboard CFAV Quest conducted in the Bahamas, during the period 20 January 2007 to 4 February 2007.

1.2 Hardware Configuration

The hardware configuration for Q302 consisted of the following subsystems:

a. Wet-end;

b. Moist-end; and

c. Dry-end.

1.3 Wet-End Configuration

The wet-end configuration consisted of the following components;

a. DASM array:
   i. 96 CORDS module; and
   ii. 2 NAS modules.
b. MANTArray:
   i. 2x48 HF channel modules;
   ii. 6x24 MF channel modules; and
   iii. 2 NAS modules.
c. Vertical Projector (VP2) – with only one free flood ring projector;
d. Neutrally Buoyant Tow cable (NBC);
e. Fiber Optic Transmitter modules (FOTX);
f. Array winch;
g. VP2 winch;
h. DASM power supply;
i. MANTA power supply; and
j. GPS and omni sonobuoys.

1.4 Moist-End Configuration

The moist-end configuration consisted of the following components:

a. MIDAS – miscellaneous data acquisition system;

b. DASM ATM array receiver;

c. MANTA ATM array receiver;

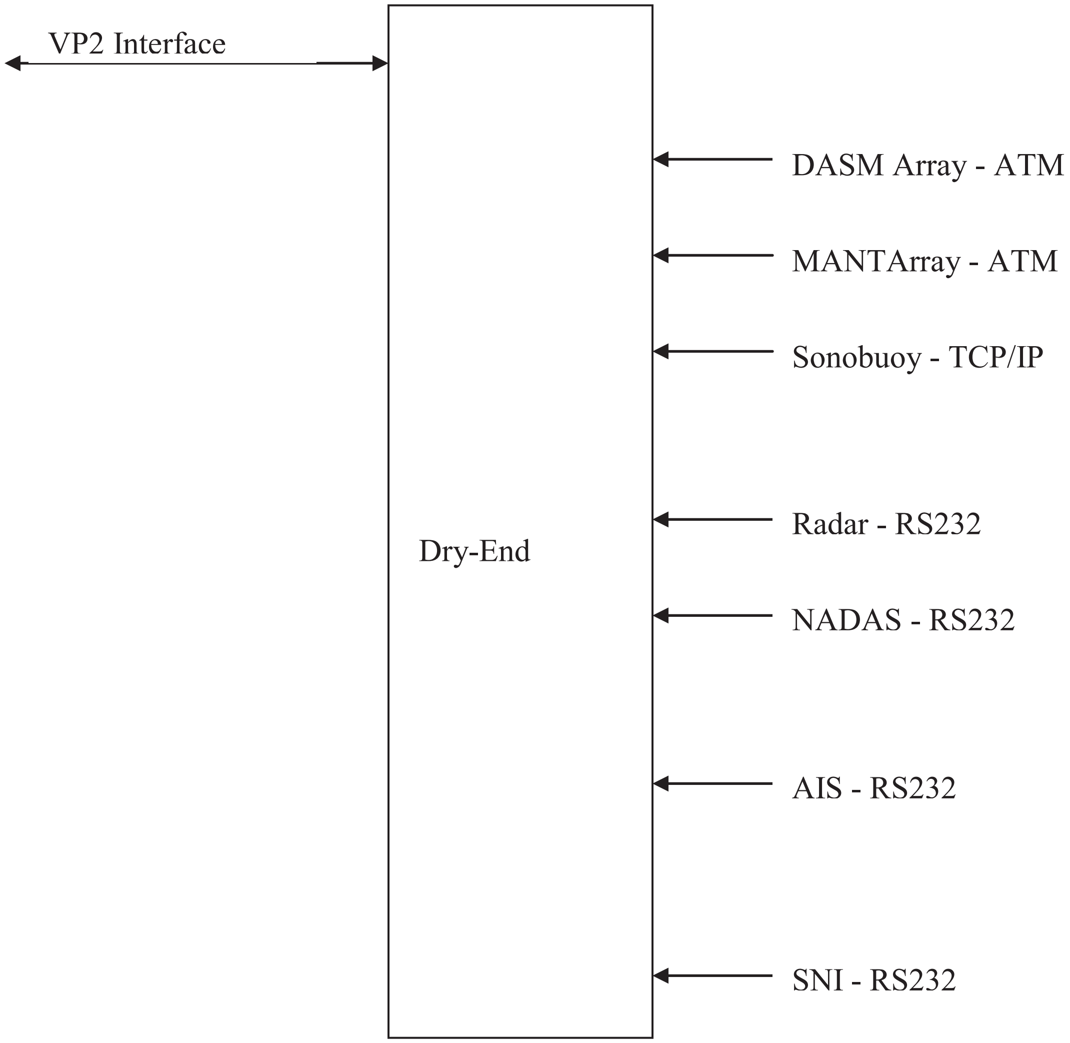
d. VP2 amplifiers; and

e. Sonobuoy receiver.

1.5 Dry-End Configuration

Figure 1 presents the external interfaces for the dry end. Figure 2 presents the hardware configuration for the dry-end.

Figure 1. Dry End External Interfaces



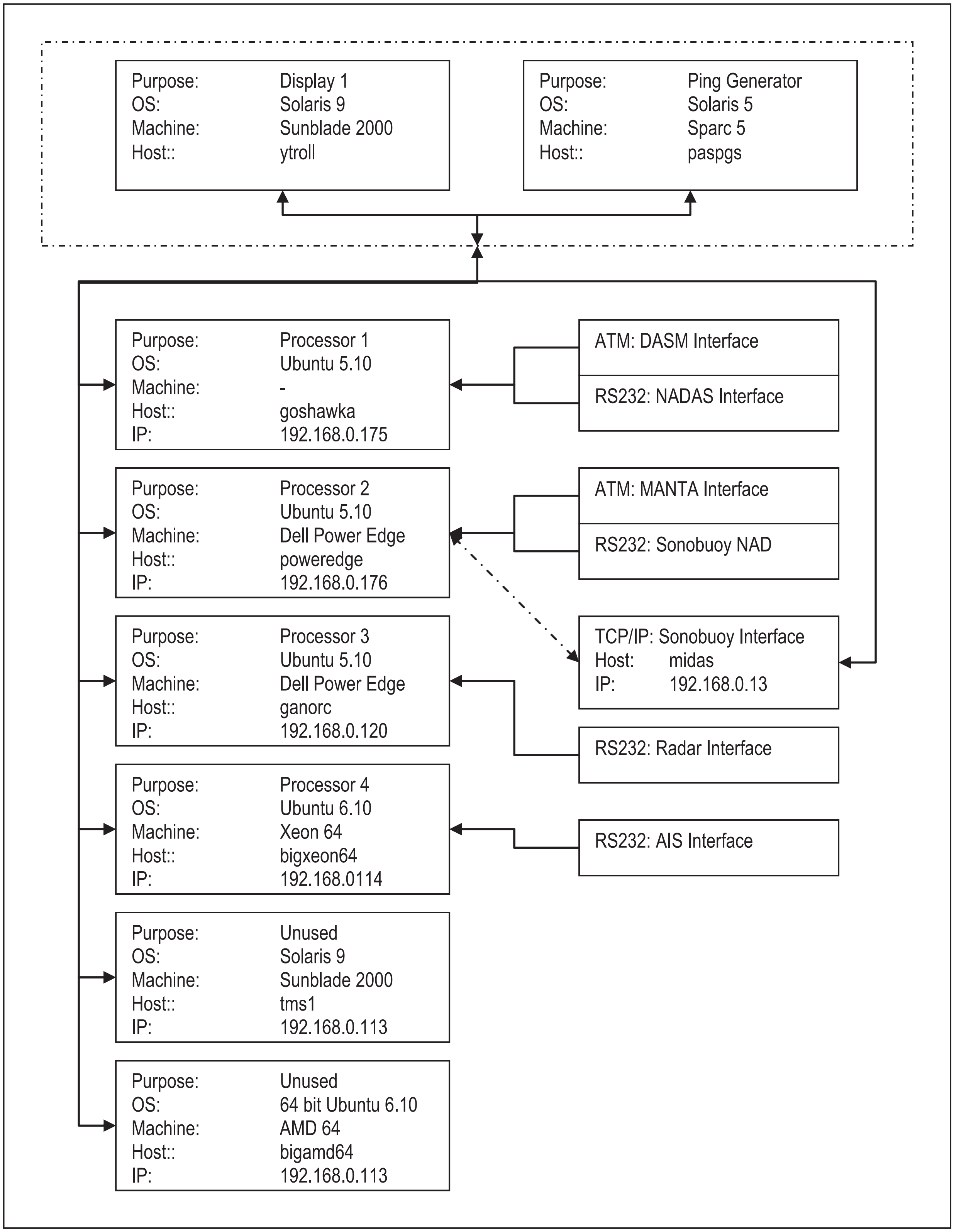


Figure 2. Dry End Processors

2. Dry-End Software Configuration

The dry-end software consists of components of the Sonar Test Bed and segments of the USA DOD Defense Information Infrastructure Common Operating Environment (DII/COE) configured to provide the active and passive sonar system functionality identified in section 2.1.

2.1 Software Functionality

The dry-end software provides the following functionality:

a. Data management and distribution;

b. Data definition;

c. External interfaces;

d. Signal processing;

e. Feature processing;

f. Contact processing;

g. Ping processing;

h. Track processing; and

i. Operator interface.

2.1.1 Data Management and Distribution

The data distribution and management functionality supports data producers, data consumers and data servers on a network of distributed processing resources. Software components are either data producers and/or data consumers, or data servers, but not both. The data producers and data consumers do not provide an intercomponent interface. The data servers provide distributed common services to data producers and data consumers to create and delete data items, discover and describe data items, read and write data item updates, subscribe for notification of data item creation and destruction, and subscribe for data item updates. Figure 3 presents a diagram that shows some of the interactions between generic data producer, data consumer, and data server components.

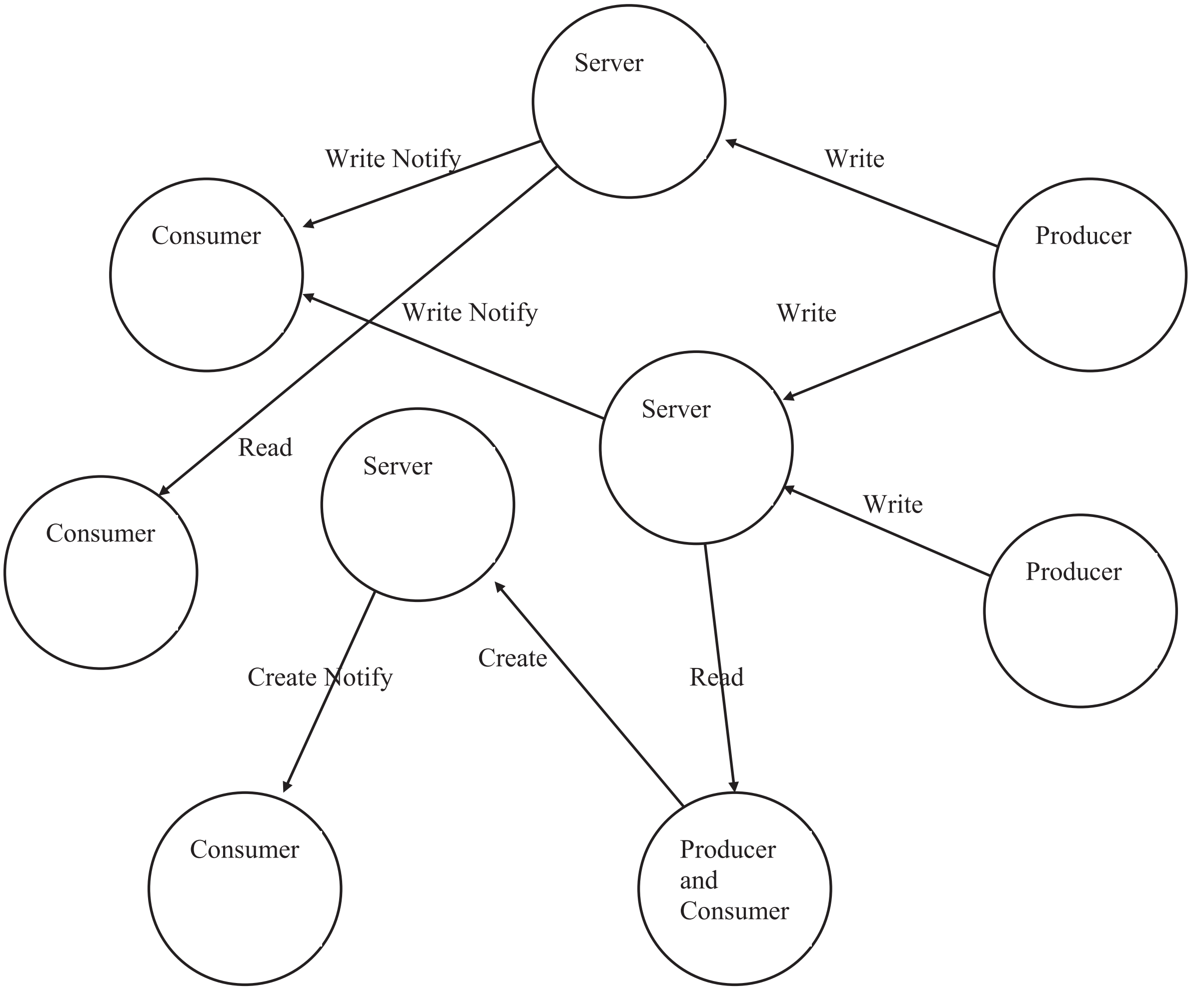


Figure 3. Producer, Consumer and Server Component Interaction

2.1.2 Data

The dry-end software components exchange data item updates to provide distributed functionality. The data server components support a general data model that is content independent and uses a set of metadata to describe data item attributes. Components that need to exchange data item updates use common data types. Where possible, specialized data types that have broad distribution between components have been generalized into common data types. The common data types are:

a. Acoustic Data Type;

b. Feature Data Type;

c. Contact Data Type;

d. Sensor Data Type;

e. Time Data Type;

f. Track Data Type;

g. Waveform Data Type; and

h. Wavetrain Data Type.

2.1.2.1 Acoustic Data Type

```
struct STB_acoustic_data_header_type {
   double time;
   char time_string[STB_TIME_LENGTH];

   long checkword;
   long checksum;

// ping values
   long ping_flag;
   long ping_offset;
   long ping_to_ping_offset;

   double ping_time;
   double ping_latitude;
   double ping_longitude;
   double ping_speed;
   double ping_heading;

// own values
   double latitude;
   double longitude;
   double speed;
   double heading;
// sensor values
   double sensor_latitude;
   double sensor_longitude;
   double sensor_speed;
   double sensor_heading;
   double sensor_depth;
   double sensor_gain;
   double sensor_sensitivity;

   double sound_speed;
   double array_scope;
   double temperature;
};
```

2.1.2.2 Feature Data Type

```
struct STB_feature_data_v2_type {
   int check_word;
   int check_sum;
   double time;
   char time_string[STB_TIME_LENGTH];

   int command;
   int status;

   char name[STB_NAME_LENGTH];
   char associated_name[STB_NAME_LENGTH];
   double bearing;
   double bearing_variance;
   double bearing_rate;
   double bearing_rate_variance;
   double frequency;
   double frequency_variance;
   double frequency_rate;
   double frequency_rate_variance;
   double amplitude;
   double amplitude_variance;
   double range;         // nautical miles
   double range_variance;
   double range_rate;
   double range_rate_variance;
   double doppler_speed; // knots
   double heading;
   double depth;         // metres
   double speed;         // knots
   double longitude;
   double latitude;
// uncertainty
   double major_axis_bearing;
   double major_axis_range;
   double minor_axis_range;

// sensor
   double sensor_heading;
   double sensor_depth;
   double sensor_speed;
   double sensor_longitude;
   double sensor_latitude;

//source
   double source_heading;
   double source_depth;
   double source_speed;
   double source_longitude;
   double source_latitude;
};
```

2.1.2.3 Contact Data Type

```
class STB_contact_data_class

{
   public:

   char name[40];
   char feature_name[40];
   char master_name[40];
   char classification[40];
   char time_string[32];
   long threat;
   long command;
   long status;
   long check_word;
   double time;

// contact
   double heading;
   double speed;
   double longitude;
   double latitude;
   double sound_speed;

// uncertainty
   double major_axis_bearing; // degree true
   double major_axis_range;   // nautical miles
   double minor_axis_range;   // nautical miles

// platform
   double ship_heading;
   double ship_speed;
   double ship_longitude;
   double ship_latitude;
// sensor
   double sensor_heading;
   double sensor_depth;
   double sensor_longitude;
   double sensor_latitude;
// measurement
   double amplitude;
   double amplitude_rate;
   double amplitude_variance;
   double bearing;                  // true bearing
   double bearing_rate;
   double bearing_variance;
   double bearing_rate_variance;
   double frequency;
   double frequency_rate;
   double frequency_jitter;
   double frequency_wander;
   double frequency_variance;
   double frequency_rate_variance;
   double range;           // nautical miles
   double range_rate;      // knots - + closing and - opening
};
```

2.1.2.4 Sensor Data Type

```
struct STB_sensor_data_type {
   int check_word;
   int check_sum;
   double time;
   char time_string[STB_TIME_LENGTH];
   int command;
   int status;
   double latitude;
   double longitude;
   double true_heading;
   double magnetic_heading;
   double speed;
   double depth;
   double pitch;
   double roll;
   double sound_speed;
   double magnetic_declination;
   double gain;
   double scope;
   double temperature;
   double voltage;
   double current;
   double tension;
};
```

2.1.2.5 Time Data Type

```
struct STB_time_data_type {
   double time;           // second count
   double difference;     // second difference from last record
   long cycle;            // cycle count
   long check_word;
   char time_string[STB_TIME_LENGTH];  // JJJ:HHMM:SS.TTT DD MMM YYYY
};
```

2.1.2.6 Track Data Type

```
struct STB_track_data_type {
   long status;
   long command;
   long check_word;
   long threat;
   double time;
   double speed;           // knots
   double heading;
   double longitude;
   double latitude;
// double depth;           // metres
// double range;           // metres
// double bearing;
// double cpa_time;
// double cpa_range;       // metres
// double cpa_bearing;
// double uncertainty;     // metres

   char name[STB_NAME_LENGTH];
   char time_string[STB_TIME_LENGTH];  // JJJ:HHMM:SS.TTT DD MMM YYYY
};
```

2.1.2.7 Waveform Data Type

```
struct STB_waveform_v3_type {
   long check_word;
   long status;
   double time;
   double delay;
   double amplitude;
   double frequency;   // center frequency
   double modulation_frequency;       // SFM
   double modulation_index;           // SFM
   double bandwidth;
   double duration;
   double weight;
   long waveform;
   long shading;
   char time_string[32];
   char name[32];
   char source_name[32];
};
```

2.1.2.8 Wavetrain Data Type

```
struct STB_wavetrain_v2_type {
   waveform_v3_type waveform_table[WAVETRAIN_V2_WAVEFORMS];
   double time;
   double ping_time;
   double latitude;
   double longitude;
   double heading;
   double speed;
   double depth;
   double power;        // attenuation dB
   double bearing;      // true, -1 omni
   double sound_speed;

   long waveforms;
   long check_word;
   char time_string[STB_TIME_LENGTH];
   char ping_time_string[STB_TIME_LENGTH];
   char name[STB_NAME_LENGTH];
   char source_name[STB_NAME_LENGTH];
   long command;
   long status;
};
```

2.1.3 External Interfaces

The external interface functionality supports the input and/or output of data from the external interfaces identified in Figure 1. Each interface function is a standalone component that supports the protocol of the interface to input or output data across the interface and provide functionality to process data, and send and/or receive data item updates to/from one or more data servers.

2.1.3.1 DASM Array Interface

The DASM array interface function supports an ATM interface that receives DASM array data. The DASM array data consists of 256 channels of 24-bit data at a sample frequency of 4096 Hz. There are 96 channels of omni time series data and 96 channels of dipole time series data . This function subscribes to updates of the 0.125 second time data item. This function concatenates the channel time series from the array data into 0.25 second blocks of data and generates acoustic data updates for the DASM channel time series data item. This function averages the non-acoustic sensor values from the array data over a 0.25 second period and generates sensor data updates for the forward DASM sensor data item and the after DASM sensor data item. The forward DASM sensor values are heading, depth, roll and temperature. The after DASM sensor values are heading, depth and roll. The time value from the latest time data update is used to provide a time value for the acoustic data item updates and sensor data item updates.

2.1.3.2 MANTArray Interface

The MANTArray interface function supports an ATM interface that receives MANTA array data. The MANTA array data consists of 288 channels of 24-bit data at a sample frequency of 4096 Hz. There are 192 MF channels and 48 HF channels of time series data. This function subscribes to updates of the 0.125 second time data item. This function concatenates the channel time series from the array data into 0.25 second blocks of data and generates acoustic data updates for the MANTA channel time series data item. This function averages the non-acoustic sensor values from the array data over a 0.25 second period and generates sensor data updates for the forward MANTA sensor data item and the after MANTA sensor data item. The forward MANTA sensor values are heading, depth and temperature. The after MANTA sensor values are heading, depth and temperature. The time value from the latest time data update is used to provide a time value for the acoustic data item updates and sensor data item updates.

2.1.3.3 Sonobuoy Interface

The sonobuoy interface supports a TCP/IP socket interface that receives sonobuoy time series data. This function subscribes to updates of the 0.125 second time data item. This function concatenates the time series into 0.25 second blocks of data and generates acoustic data updates for the sonobuoy time series data item. The time value from the latest time data update is used to provide a time value for the acoustic data item updates.

2.1.3.4 Sonobuoy NAD Interface (Not Availiable)

A new interface will need to be developed for the GPS sonobuoy. However, there are no additional RS232 ports available.

The sonobuoy NAD interface function supports a RS232 serial interface that receives position reports at a periodic rate for one or more sonobuoys. Each position report includes a buoy identifier, a time, a latitude and a longitude. The time, latitude and longitude data from each position report is used to generate a track update for the associated sonobuoy track data item.

2.1.3.5 Radar Interface

The radar interface function supports a RS232 serial interface that receives radar contact reports at a periodic rate for one or more radar contact. Each contact report includes a contact identifier, a range, a bearing, a course and a speed. This function subscribes for updates of the own track data item. This functions uses the time, the latitude and the longitude from the latest update of the own track data item and the range and bearing from the radar contact reports to generate a latitude and longitude for the radar contract. For each radar contact report, this function generates an update for the associated radar track data item. This function creates a new radar track data item in the appropriate data server if the radar contacxt is new.

2.1.3.6 NADAS Interface

The NADAS interface function supports a RS232 serial interface that receives ship and environmental data updates at a periodic rate from the CFAV Quest Non-Acoustic Data Acquisition System. The ship and environmental data updates include time, ship speed, ship heading, ship latitude, ship longitude, bottom depth, wind speed, and wind direction.

2.1.3.7 VP2 Interface

The VP2 interface function supports two analogue 10 VPP output wavetrain channels, four digital TTL input fault channels and two digital TTL output enable channels. This function subscribes for updates of the ping notification data item for ping notifications. When a ping notification is received for the VP2, this function generates a 16-bit 2's complement wavetrain time series as specified in the ping notification update at a sample frequency of 20 kHz and outputs the time series on the enabled wavetrain output channels at the ping time specified in the ping notification update. Prior to the specified ping time, this function sets the specified output enable channels to the enabled state. When any output enable channels are enabled, this function periodically monitors the corresponding input fault channel for a fault state. When this function detects a fault on an input fault channel, this function terminates the generation of the wavetrain time series on the enabled wavetrain output channels. This function may be configured to ignore detected faults and continue generating the wavetrain time series.

2.1.4 Signal Processing

The signal processing functionality is divided into DASM, MANTA and sonobuoy signal processing functionality as described in the following paragraphs. There is no beamforming on this trail. There will however be ascans and lofargrams to determine array health.

2.1.4.1 DASM Signal Processing

Figure 4 presents the DASM signal processing functionality.

The channel spectral analysis performs a 2048 point real FFT every 0.75 seconds on 192 channels (96 omni and 96 dipole) of real time series with a 25 percent time series overlap to generate a 1 Hz complex spectra from 0 – 2047 Hz.

The channel time series analysis performs a 2048 point complex IFFT on 192 channels of complex spectra to generate a complex time series with a sample frequency and bandwidth of 2048 Hz and a heterodyne frequency of 1024 Hz. The first 25 percent of the complex time series points are discarded.

The channel narrowband analysis performs 1024 and 2048 point complex FFTs on 192 channels of 2048 Hz complex time series with a 50 percent overlap to generate normalized and quantized 3-bit pixel data at 1 Hz and 2 Hz resolution

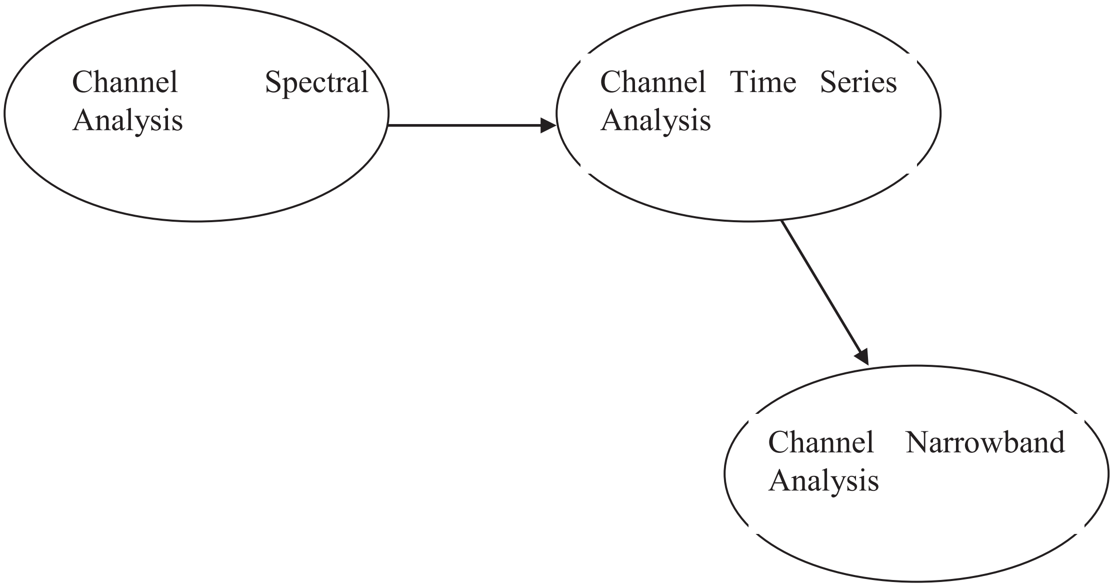

Channel Spectral Analysis
Channel Time Series Analysis
Channel Narrowband Analysis

Figure 4. DASM and MANTA Signal Processing Functionality

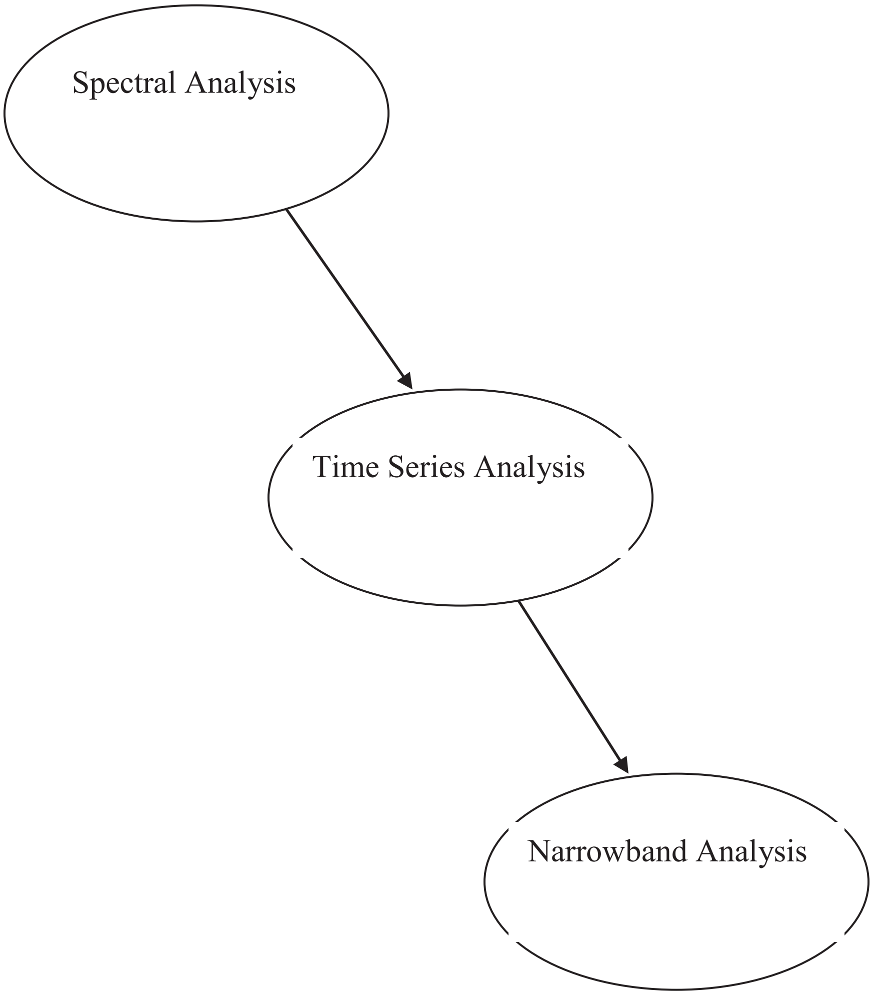


Figure 5. Sonobuoy Signal Processing Functionality

The beam narrowband analysis performs complex FFTs on complex beam time series to generate complex beam spectra at various resolutions with a 50 percent overlap of the time series. The complex beam spectra is converted to beam power spectra and integrated over a several time periods. The integrated beam power spectra is normalized and qunatized to 3-bit pixel data.

2.1.4.2 MANTA Signal Processing

Figure 4 presents the MANTA signal processing functionality.

The channel spectral analysis performs a 4096 point real FFT every 0.75 seconds on 192 channels (144 MF and 48 HF) of real time series with a 25 percent time series overlap to generate a 1 Hz complex spectra from 0 – 2047 Hz.

The channel time series analysis performs a 2048 point complex IFFT on 192 channels of complex spectra to generate a complex time series with a sample frequency and bandwidth of 2048 Hz and a heterodyne frequency of 1024 Hz. The first 25 percent of the complex time series points are discarded.

The channel narrowband analysis performs 1024 and 2048 point complex FFTs on 192 channels of 2048 Hz complex time series with a 50 percent overlap to generate normalized and quantized 3-bit pixel data at 1 Hz and 2 Hz resolution.

2.1.4.3 Sonobuoy Signal Processing

Figure 5 presents the sonobuoy signal processing functional interaction.

The spectral analysis performs a 4096 point real FFT every 0.75 seconds on 16 channels of real time series with a 25 percent time series overlap to generate a 1 Hz complex spectra from 0-2047 Hz.

The time series analysis performs a 2048 point complex IFFT on 16 channels of complex spectra to generate a complex time series with a sample frequency and bandwidth of 2048 Hz and a heterodyne frequency of 1024 Hz. The first 25 percent of the complex time series points are discarded.

The narrowband analysis performs 1024 and 2048 point complex FFTs on 16 channels of 2048 Hz complex time series with a 50 percent overlap to generate normalized and quantized 3-bit pixel data at 1 Hz and 2 Hz resolution.

2.1.5 Ping Processing

The ping processing functionality provides ping management functions to create waveform, wavetrain and ping sequence data items, and to schedule ping sequences. A ping sequence consists of a series of wavetrains trainsmitted at a periodic rate. A wavetrain consists of 1 or more waveforms. A waveform is a basic signal including continuous wave (CW), linear frequency moduled (LFM), hyperbolic frequency moduled (HFM) and sinusoidal frequency modulated (SFM). Complex signals such as a Costas signal are specified as a wavetrain of CW waveforms.

This function creates waveform, wavetrain and ping sequence data items in specifed waveform, wavetrain and ping sequence groups on specified data servers. This function subscribes for updates of a specified ping control data item. A ping control update contains a command attribute, a ping sequence name, a source name, a gain level and a time. The ping control commands include start, stop and pause.

When a ping control update is received, this function executes the specified command. If the start command is specified this function schedules the specifed ping sequence for the specifed source starting at the specified time with the specifed gain applied to the gain already specifed in the ping sequence. This function generates a pre-ping notification update for a specified pre-ping notification data item a specified number of seconds prior to the scheduled ping time. This function generates a ping notification update for a specified ping notification data item at the scheduled ping time. The pre-ping notification and ping notification updates specify the source name, source latitude, source longitude, source course, source speed, wavetrain name, and the name, type, delay, gain, frequency, bandwidth, modulation index, modulation frequency for each waveform in the wavetrain.

2.1.6 Track Processing

The track processing functionality provides track management, track extrapolation and track interpolation functions as described in the following paragraphs.

2.1.6.1 Track Management

The track management functions supports the creation, destruction and update of track data items in specified track groups on specified data servers

2.1.6.2 Track Extrapolation

The track extrapolation function generates track position updates for track data items in specified track groups on specified data servers that have an extrapolate status.

This function subscribes for track data item creation and destruction notifications. When a track data item creation notification is received this function subscribes for updates of the track data item. When a track update is received, this function checks the status value to determine if the track is an extrapolated track.

This function subscribes for updates of a specifed time data item. When time updates are received, this function generates a new position for each extroplated track using the previous position, course, speed and the time difference. This function generates a track update for the track data item.

2.1.6.3 Track Interpolation

The track interpolation function generates track position updates for track data items in specified track groups on specified data servers that have an interpolated status.

This function subscribes for track data item creation and destruction notifications. When a track data item creation notification is received this function subscribes for updates of the track data item. When a track update is received, this function checks the status value to determine if the track is an interpolated track. If the track is an interpolated track, this function reads the track records of the specifed reference track data item that are before and after the time in the time update.

This function subscribes for updates of a specifed time data item. When time updates are received, this function generates a new position for each interpolated track using the current and next specifed reference track positions and the time difference. This function generates a track update for the track data item.

2.1.7 Operator Interface

The operator inteface functionality provides input and output functions to control and monitor other functionality, and display and analyse the data generated by the other functionality. The input functionality is provided by keyboard, trackball, button and touch screen hardware. The output functionality is provided by graphics displayed on video monitors and audio streamed through headphones.

This function provides a set of displays with associated tools to display and analyse data.

a. Displays:

i. lscan – lofargram (manta, dasm, sonobuoy);

ii. ascan – (manta, dasm, sonobuoy);

iii. Ping control;

iv. Track manager; and

v. Button manager.

2.2 Component Identification

The sonar test bed consists of a set of software components that provide data independent, data dependent and application specific functionality. Each component is a standalone executable and is configured using a configuration file. An application can run multiple instances of a component to distribute processing across multiple processors.

The list of STB components includes:

a. Acoustic Display (AD) – JAVA GUI component that provides ASCAN(ALI), BSCAN(BTI) and LSCAN(LOFAR) displays with associated tools.

b. Alert Summary (AS) – JAVA GUI component generates a tabular display of alert reports.

c. Ampliier Interface (AI) –generates analogue wavetrain signal on trigger from IRIG time signal and monitors input fault lines.

d. Audio manager (AM) – generates a time series stream from historical spectra at specified direction and time.

e. Audio Player (AP) – JAVA GUI component that provides functionality to control and play time series through local audio device.

f. Sonobuoy interface (BI) – TCP/IP socket interface application supporting the MIDAS protocol.

g. Bearing Plotter (BP) – Chart GUI component that provides functionality to plot feature bearing lines, ambiguous bearing lines and range ellipses.

h. CANTASS array interface (CI) – RS422 16-bit parallel interface application that receives data from CANTASS Gould array receiver and generate acoustic time series updates and non-acoustic sensor updates.

i. Contact manager (CM) – generates and correlates contact data reports from passive feature data reports.

j. Contact Plotter (CP) – JAVA Chart GUI component plots contact reports on concentric rings on the chart display centered on a track.

k. Contact Summary (CS) – JAVA GUI component that provides tabular display of contact reports.

l. Contact Tool (CT) – JAVA chart GUI component that provides functionality to create contacts.

m. Cluster Tool (XT) – generates a weighted average position for a group of features bounded by an ellipse.

n. Data display (DD) – X application that provides ASCAN(ALI), BSCAN(BTI) and LSCAN(LOFAR) displays with associated tools, tactical displays, and sensor displays.

o. Polar Display (PD) – JAVA GUI component that provides a polar display for active data with associated tools.

p. Data Browser(DB) – JAVA GUI component.

q. Data Editor(DE) – command line GUI component.

r. Data logger (DL) – writes to text formated disk files specified data updates.

s. Data Recorder (DR) – writes data updates to disk or tape files.

t. Data Player (DP) – replays data updates from disk or tape files.

u. DAT32 Data Recorder (DR.DAT32) – writes single data item updates to disk or tape files in DRDC Atlantic DAT32 format.

v. DAT32 Data Player (DP.DAT32) – replays single data item updates from disk or tape files in DRDC Atlantic DAT32 format.

w. Data Server (DS) – provides data management and distribution services to all non-data server STB components.

x. DASM Array interface (DI) – ATM interface application that receives DASM array data and generates acoustic time series updates and non-acoustic sensor updates.

y. Feature Manager (FM) – provides broadband and narrowband passive signal detection, signal estimation and signal following functionality.

z. Feature Plotter (FP) – Chart GUI component plots active feature reports on chart.

aa. GPS interface (GI) – RS232 serial line interface application that receives latitude, longitude, course and speed reports and generates track updates.

bb. HPA NAD interface (HI) - RS232 serial line application that receives HPA heading, depth, temperature and pressure reports and generates sensor updates.

cc. Localization Manager (LM) – generates a localization estimate from a series of contact reports using DRDC Atlantic grid search algorithm.

dd. Localization Plotter (LP) – plots the localization results on the chart.

ee. Localization Tool (LT) -

ff. MANTArray interface (MI) – ATM interface application that receives MANTArray data and generates acoustic time series updates and non-acoustic sensor updates.

gg. NADAS interface (NI) - RS232 serial line application that receives own ship latitude, own ship longitude,own ship course, own ship speed, wind heading and wind speed, and water depth reports and generates own ship track updates.

hh. Menu Display (MD) – JAVA GUI application that provides menu display and selection, and message areas.

ii. Ping Controller (PC) – JAVA GUI application that controls ping selection, ping source and ping sequence.

jj. Ping Manager (PM) – manages synchronous and asynchronous ping scheduling.

kk. Ping Plotter (PP) – Chart GUI component that generates enlarging ellipses on chart indicating outgoing ping, received sound and processing latency.

ll. Rader Interface (RI) – RS232 serial line application that receives radar contact reports, creates tracks and generates track updates.

mm.System Manager (SM)

nn. Sensor manager (XM) – generates extended sensor data including position estimates for towed sensors, sound speed estimates, true heading of sensor and water depth under sensor.

oo. Signal Processor (SP) – provides macro level sonar signal processing functionality.

pp. Time Generator (TG) – generates periodic time updates.

qq. Track Manager (TM) – provides track management functionality including waypoint interpolation and extrapolation.

rr. Track Plotter (TP) – Chart GUI compoent that plots track reports on the chart display.

ss. Track Summary (TS) – JAVA GUI component that provides tabular display of track reports.

tt. Track Tool (TT) – JAVA chart GUI component that provides functionality to create tracks.

uu. UAT interface (UI) – TCP/IP socket interface application that receives UAT latitude and longitude reports and generates UAT track reports.

vv. VP2 NAD interface (VI) – RS232 serial interface application that receives VP2 depth and acceleration reports and generates VP2 sensor reports.

ww.Winch interface (WI) - RS232 serial interface application that receives winch scope reports and generates winch sensor reports.

xx. Waveform manager (WM) – manages waveform, wavetrain and ping sequence definitions.

Figure 6 graphically groups the above software components in to one of six functional groups.

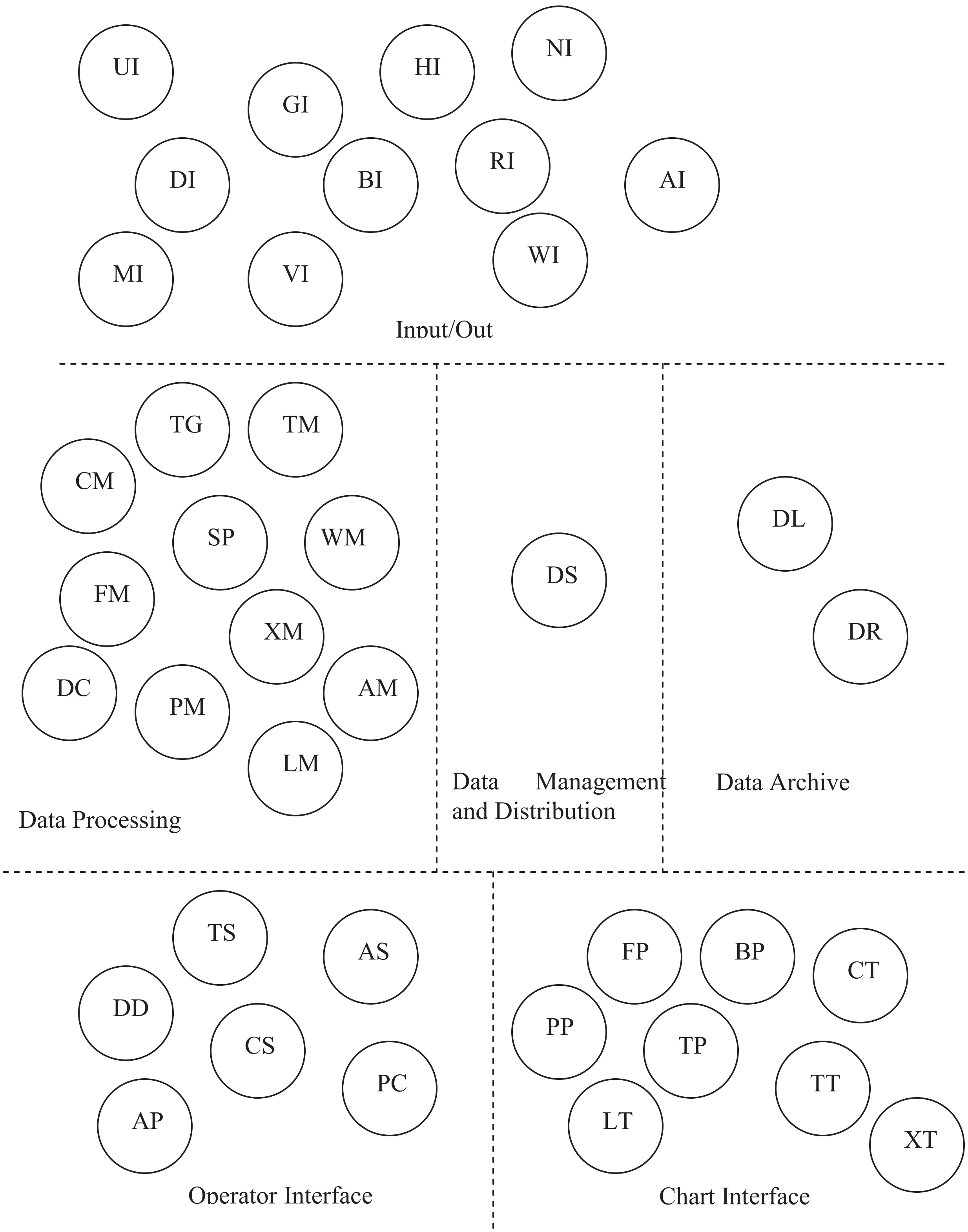


Figure 6. Software Component Groupings

2.3 Component Allocation

The STB components and configuration files that were allocated to the processors to support the dry-end applications were:

Note that all script source stb.cfg. There are two stb.cfg configuration used. One is stb.cfg.q302, which is the live configuration. Another configuration is stb.cfg.sim, which is used for offline testing.

a. Any host:

   i. DE.Q302; and
   ii. JDE.

b. goshawka – processor 1: 192.168.0.175:

   i. DI.Q302 di.cfg, <atm interface>;
   ii. NI.Q302 ni.cfg, <serial interface>; and
   iii. DS.Q302.DA.IN ds.cfg.dasm.in.

c. poweredge – processor 2: 192.168.0.176:

   i. MI.Q302 mi.cfg, <atm interface>;
   ii. DS.Q302.MA.IN ds.cfg.manta.in;
   iii. BI.Q302 bi.cfg <over existing eithernet>;
   iv. DS.Q302.SB.IN ds.cfg.sb.in; and
   v. SNI.Q302 sni.cfg, <serial interface>.

d. ganorc – processor 3: 192.168.0.120:

   i. NS.Q302;
   ii. DS.Q302.DA.CH ds.cfg.dasm.ch;
   iii. DS.Q302.MA.CH ds.cfg.manta.ch;
   iv. DS.Q302.SB.CH ds.cfg.sb.ch;
   v. DS.Q302.NAD ds.cfg.nad;
   vi. SP.Q302.DA.TS sp.cfg.dasm.ts;
   vii. SP.Q302.DA.CH sp.cfg.dasm.ch;
   viii. SP.Q302.MA.TS sp.cfg.manta.ts;
   ix. SP.Q302.MA.CH sp.cfg.manta.ch;
   x. SP.Q302.SB.CH sp.cfg.sb.ch;
   xi. SP.Q302.SB.TS sp.cfg.sb.ch;
   xii. XM.Q302.MANTA xm.cfg.manta;
   xiii. XM.Q302.DASM xm.cfg.dasm;
   xiv. XM.Q302.VP2 xm.cfg.vp2;

xv. TM.Q302 tm.cfg;
xvi. GM.Q302 gm.cfg;
xvii. PM.Q302 pm.cfg;
xviii. WM.Q302 wm.cfg;
xix. TG.Q302 tm.cfg; and
xx. RI.Q302 ri.cfg, <serial interface>.

e. bigzeon64 – processor 4: 192.168.0.114:

i. AI.Q302 ai.cfg <serial interface>;
ii. DP.Q302 dp.cfg;
iii. DAT.Q302.SB dat.cfg.sb;
iv. DAT.Q302.MANTA dat.cfg.manta;
v. DAT.Q302.DASM dat.cfg.dasm; and
vi. DR.Q302 dr.cfg.

f. tms – processor 5.

g. ytroll - processor 6:

i. TP tp.cfg.

h. paspgs – processor 7:

i. PG.Q302 pg.cfg.

i. Display – processor 9:

i. DD.Q302.DA.AS dd.cfg.da.ascan;
ii. DD.Q302.DA.LS dd.cfg.da.lscan;
iii. DD.Q302.MA.AS dd.cfg.ma.ascan;
iv. DD.Q302.MA.LS dd.cfg.ma.lscan;
v. DD.Q302.SB.LS dd.cfg.sb.ascan;
vi. DD.Q302.SB.AS dd.cfg.sb.lscan;
vii. DD.Q302.KP dd.cfg.kp;
viii. DD.Q302.TS dd.cfg.ts; and
ix. DD.Q302.CHAT dd.cfg.chat.

3. Shell Script Identification

The dry-end software applications were started and stopped through a sequence of nested UNIX tcsh scripts that were either locally executed in a terminal session or remotely executed through the ssh utility. Data server scripts were seperated from application scripts because most data items were statically defined in data server configuration files and all the data servers had to be running prior to the applications running.

3.1 Processor Start Scripts

3.1.1 All Processors: start_stb.q302

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg.q302
start_mode.q302
#
# master host
# ganorc
# runs name service, majority of data servers, signal processing, data processing
echo "spinning up ganorc"
ssh $PROC3 "cd ./"${DIR}";start_stb.proc.3; exit" & # RI_IF, BI_IF
#
sleep 60
#
# child hosts
# note: these host have dependicies to the master host

echo "spinning up bigxeon64 processes"
ssh $PROC4 "cd ./"${DIR}";start_stb.proc.4; exit" &# AIS_IF

# ytroll
#ssh $PROC7 "cd ./"${DIR}";start_stb.proc.7; exit" &# VP2
#
# paspgs
# sleep 10
# echo "spinning up paspgs processes"
# ssh "pascam@paspgs" "cd /export/home/pascam/src/atass/IF/pgs_interface.061016/bin; start_pg; exit" &
#
sleep 10
echo "spinning up stbdisplay processes"
#start_stb.proc.9 &
ssh –X $PROC9 "cd ./"${DIR}";start_stb.proc.9" & # DD

# remote chart
```

```
sleep 10
start_chart.remote &
startJDE &
#
sleep 60
#
echo "spinning up goshawka processes"
ssh $PROC1 "cd ./"${DIR}";start_stb.proc.1; exit" &# DI_IF, NI_IF

echo "spinning up poweredge processes"
ssh $PROC2 "cd ./"${DIR}";start_stb.proc.2; exit" &# MI_IF

# end
```

3.1.2 Processor 1 (goshawka): start_stb.proc.1

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
# start_ns
#
# start_ds.da.in "–t"
#
sleep 5
#
# interface components
#
start_di
start_ni
#
# end
```

3.1.3 Processor 2 (poweredge): start_stb.proc.2

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
#start_ns
#
# data server component
#
```

```
#start_ds.ma.in
#
sleep 5
#
# interface components
#
start_mi  # manta interface
start_sni  # sonobuoy nad interface
start_bi  # sonobuoy interface
# start_wi
#
# end
```

3.1.4 Processor 3 (ganorc): start_stb.proc.3

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
echo "starting name service"
start_ns
#
sleep 5
#
# data server components
#
echo "starting data server dasm input"
ssh ${PROC1} "cd ./"${DIR}"; ./start_ds.da.in; exit" &

echo "starting data server dasm channel"
start_ds.da.ch #"–t"

echo "starting data server manata input"
ssh ${PROC2} "cd ./"${DIR}"; ./start_ds.ma.in; exit" &

echo "starting data server manta channel"
start_ds.ma.ch # "–t"

echo "starting data server NAD"
start_ds.nad # "–t"

echo "starting data server sonobuoy input"
#start_ds.sb.in # "–t"
ssh ${PROC2} "cd ./"${DIR}"; ./start_ds.sb.in; exit" &

echo "starting data server sonobuoy channel"
start_ds.sb.ch # "–t"
#
```

```
echo "waiting for data servers to spin up"
sleep 30
echo "data servers running"
#
echo "copying ior file to paspgs"
#
scp ./nad.ior pascam@paspgs:/export/home/pascam/src/atass/IF/pgs_interface.061016/bin
#
echo "starting the time generator"
start_tg "–x"

# recording components
#
# start_dr
# start_dl
#
# interface components
#
echo "starting sonobuoy interface"
#start_bi

echo "starting radar interface"
start_ri
#
# data processing components
#
echo "starting waveform manager"
start_wm

# echo "starting ping manager"
# start_pm

echo "starting track manager"
start_tm

echo "starting geomagnetic manager"
start_gm

echo "starting sensor manager dasm"
start_xm.dasm

echo "starting sensor manager manta"
start_xm.manta

echo "starting sensor manager vp2"
start_xm.vp2

#
# signal processsing components
#
echo "starting signal processor dasm time series"
start_sp.da.ts
```

```
echo "starting signal processor dasm channel"
start_sp.da.ch
echo "starting signal processor manta time series"
start_sp.ma.ts
echo "starting signal processor manta channel"
start_sp.ma.ch
echo "starting signal processor sonobuoy time series"
start_sp.sb.ts
echo "starting signal processor sonobuoy channel"
start_sp.sb.ch
exit
# end
```

3.1.5 Processor 4 (bigxeon64): start_stb.proc.4

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
# start_ns
#
start_ai
start_dl
#
# end
```

3.1.6 Processor 5 (bigamd64)

NOT USED

3.1.7 Processor 6 (tms1)

NOT USED

3.1.8 Processor 7 (ytroll)

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
start_dd.chat
start_dd.ps
#
# end
```

3.1.9 Processor 8 (paspgs)

See section 3.1.1.

3.1.10 Processor 9 (stbdisplay)

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
# display components
#
start_dd.kp
start_dd.ps
start_dd.ts
start_dd.chat

start_dd.da.lscan
start_dd.ma.lscan
start_dd.sb.lscan

start_dd.da.ascan
start_dd.ma.ascan
start_dd.sb.ascan
#
# #############################################################################
# utility components
# #############################################################################
#start_chart.remote
#startJDE

# end
```

3.2 Processor Stop Scripts

3.2.1 All Processors

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg
#
# master host
# runs name service, majority of data servers, signal processing, data processing
ssh $PROC1 "cd ./"${DIR}";stop_stb &" & # DI_IF, NI_IF
ssh $PROC2 "cd ./"${DIR}";stop_stb &" & # MI_IF, WI_IF
ssh $PROC4 "cd ./"${DIR}";stop_stb &" & # AIS_IF
ssh ytroll "cd ./"${DIR}";stop_chart & " &
ssh $PROC7 "cd ./"${DIR}";stop_stb &" & # AIS_IF
ssh $PROC9 "cd ./"${DIR}";stop_stb &" & # AIS_IF
ssh pascam@paspgs "pkill PG" & # Ping Generator
ssh $PROC3 "cd ./"${DIR}";stop_stb &" & # RI_IF, BI_IF
#
#
# end
```

3.2.2 Local Processor

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/stop_stb,v $
# $Revision: 1.8 $
# $Date: 2005/06/22 15:46:01 $
# $Author: inglis $
#
source ./stb.cfg
#
ps –ae | grep $APP
#
pkill –9 $APP
pkill –9 java
pkill –9 JDE
#
sleep 2
#
# end
```

3.3 Data Server Start Scripts

3.3.1 Name Service: start_ns

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_ns,v $
# $Revision: 1.5 $
# $Date: 2005/02/22 17:26:25 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill NS."$APP"
#
sleep 2
#
./NS."$APP" $1 $NS_FLAG &
#
# end
```

3.3.2 DASM Channel: start_ds.da.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.7,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:51 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".DA.CH
#
sleep 2
#
cp ./data_server DS."$APP".DA.CH
./DS."$APP".DA.CH –c ds.cfg.dasm.ch –n DS_DASM_CH $DS_FLAG $1 &
#
# end
```

3.3.3 DASM Input: start_ds.da.in

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.2,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:50 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".DA.IN
#
sleep 2
#
./DS."$APP".DA.IN –c ds.cfg.dasm.in –n DS_DASM_IN $DS_FLAG $1 &
#
# end
```

3.3.4 MANTA Channel: start_ds.ma.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.7,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:51 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".MA.CH
#
sleep 2
#
./DS."$APP".MA.CH –c ds.cfg.manta.ch –n DS_MANTA_CH $DS_FLAG $1 &
#
# end
```

3.3.5 MANTA Input: start_ds.ma.in

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.1,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:50 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".MA.IN
#
sleep 2
#
./DS."$APP".MA.IN –c ds.cfg.manta.in –n DS_MANTA_IN $DS_FLAG $1 &
#
# end
```

3.3.6 Sonobuoy Channel: start_ds.sb.in

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.11,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:50 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".SB.IN
#
sleep 2
#
./DS."$APP".SB.IN –c ds.cfg.sb.in –n DS_SB_IN $DS_FLAG $1 &
#
# end
```

3.3.7 Sonobuoy Input: start_ds.sb.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.11,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:50 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".SB.IN
#
sleep 2
#
./DS."$APP".SB.IN –c ds.cfg.sb.in –n DS_SB_IN $DS_FLAG $1 &
#
# end
```

3.3.8 Non Acoustic Data (NAD): start_ds.nad

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ds.proc.1,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:50 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DS."$APP".NAD
#
sleep 2
#
./DS."$APP".NAD –c ds.cfg.nad –i nad.ior –n $DS $DS_FLAG $1 &
#
# end
```

3.4 Signal Processing Start Scripts

3.4.1 DASM Channel: start_sp.dasm.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".DA.CH
#
sleep 2
#
./SP."$APP".DA.CH –c sp.cfg.dasm.ch –n DS_DASM_CH $STB_FLAG $1 &
#
# end
```

3.4.2 DASM Input: start_sp.dasm.ts

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".DA.TS
#
sleep 2
#
./SP."$APP".DA.TS –c sp.cfg.dasm.ts –n DS_DASM_CH $STB_FLAG $1 &
#
# end
```

3.4.3 MANTA Channel: start_sp.manta.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".MA.CH
#
sleep 2
#
./SP."$APP".MA.CH –c sp.cfg.manta.ch –n DS_MANTA_CH $STB_FLAG $1 &
#
# end
```

3.4.4 MANTA Input: start_sp.manta.ts

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".MA.TS
#
sleep 2
#
./SP."$APP".MA.TS –c sp.cfg.manta.ts –n DS_MANTA_CH $STB_FLAG $1 &
#
# end
```

3.4.5 Sonobuoy Channel: start_sp.sb.ch

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".SB.CH
#
sleep 2
#
./SP."$APP".SB.CH –c sp.cfg.sb.ch –n DS_SB_CH $STB_FLAG $1 &
#
# end
```

3.4.6 Sonobuoy Input: start_sp.sb.in

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_sp.proc.3,v $
# $Revision: 1.7 $
# $Date: 2005/06/22 15:47:46 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SP."$APP".SB.TS
#
sleep 2
#
./SP."$APP".SB.TS –c sp.cfg.sb.ts –n DS_SB_CH $STB_FLAG $1 &
#
# end
```

3.5 Interface Start Scripts

3.5.1 DASM: start_di

```
#!/bin/tcsh -f
#
# File: start_sys.q302
#
# Author: Chris Wheeler
# Company: General Dynamics Canada
# Created on: December 14, 2006
#
# Revision Id:          %A%
# Last Delta (YY-MM-DD): %E% %U%
# Delta Key:            %K%
#
#
# note
#
# may need to install atm driver
# "modprobe he"
# use "lsmod | grep he" to check if driver is loaded
source ./stb.cfg
#
pkill DI."$APP"
#
sleep 2
#
# menu options
#./DI.Q302 -c di.cfg -n DS_DASM_IN $STB_FLAG -e 4 -m
# no menu options
./DI."$APP" -c di.cfg -n DS_DASM_IN $STB_FLAG $1 &
#
# end
```

3.5.2 MANTA: start_mi

```
#!/bin/tcsh –f
#
# File: start_mi
#
# Author: Chris Wheeler
# Company: General Dynamics Canada
# Created on: December 14, 2006
#
# Revision Id:           %A%
# Last Delta (YY–MM–DD): %E% %U%
# Delta Key:             %K%
#
#
# note
#
# may need to install atm driver
# "modprobe he"
# use "lsmod | grep he" to check if driver is loaded
source ./stb.cfg
#
pkill MI."$APP"
#
sleep 2
#
# menu options
#./MI.Q302 –c mi.cfg –n DS_MANTA_IN $STB_FLAG $1 –m
# no menu options
./MI."$APP" –c mi.cfg –n DS_MANTA_IN $STB_FLAG $1 &
#
# end
```

3.5.3 Sonobuoy Time Series: start_bi

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_bi,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:49 $
# $Author: inglis $
#
source ./stb.cfg
#
#cp ./sonobuoy_interface BI."$APP"
pkill BI."$APP"
#
sleep 2
#
./BI."$APP" –c bi.cfg –n DS_SB_IN $STB_FLAG $1 &
#
# end
```

3.5.4 NADAS: start_ni

```
#!/bin/tcsh –f
#
# File: start_ni
#
# Author: Chris Wheeler
# Company: General Dynamics Canada
# Created on: December 14, 2006
#
# Revision Id:          %A%
# Last Delta (YY–MM–DD): %E% %U%
# Delta Key:            %K%
#
#
source ./stb.cfg
#
pkill NI."$APP"
#
sleep 2
#
./NI."$APP" –c ni.cfg –n $DS $STB_FLAG $1 &
#
# end
```

3.5.5 Winch: start_wi

```
#!/bin/tcsh -f
#
# File: start_wi
#
# Author: Chris Wheeler
# Company: General Dynamics Canada
# Created on: December 14, 2006
#
# Revision Id:          %A%
# Last Delta (YY-MM-DD): %E% %U%
# Delta Key:            %K%
#
#
source ./stb.cfg
#
pkill WI."$APP"
#
sleep 2
#
./WI."$APP" -c wi.cfg -n $DS $STB_FLAG $1 &
#
# end
```

3.5.6 VP2

See 3.1.1.

3.5.7 AIS: start_ai

```
#!/bin/tcsh -f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ri,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:56 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill AI."$APP"
#
sleep 2
#
cp ./ais_interface AI."$APP"
./AI."$APP" -c ai.cfg -n $DS $STB_FLAG $1 &
#
# end
```

3.5.8 Radar: start_ri

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ri,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:56 $
# $Author: inglis $
#
source ./stb.cfg
#
#
pkill RI."$APP"
#
sleep 2
#
cp ./radar_interface RI."$APP"
./RI."$APP" –c ri.cfg –n $DS $STB_FLAG $1 &
#
# end
```

3.5.9 Sonobuoy NAD: start_sni

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_ri,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:56 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill SNI."$APP"
#
sleep 2
#
cp ./sonobuoy_nad_interface SNI."$APP"
./SNI."$APP" –c sni.cfg –n $DS $STB_FLAG $1 &
#
# end
```

3.6 Data Processing Start Scripts

3.6.1 DASM Sensor Manager: start_xm.dasm

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_xm.sqr19,v $
# $Author: inglis $
# $Date: 2005/02/22 17:25:17 $
#
source ./stb.cfg
#
pkill XM."$APP".DASM
#
sleep 2
#
./XM."$APP".DASM –c xm.cfg.dasm –n $DS $STB_FLAG $1 &
#
# end
```

3.6.2 MANTA Sensor Manager: start_xm.manta

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_xm.sqr19,v $
# $Author: inglis $
# $Date: 2005/02/22 17:25:17 $
#
source ./stb.cfg
#
pkill XM."$APP".MANTA
#
sleep 2
#
./XM."$APP".MANTA –c xm.cfg.manta –n $DS $STB_FLAG $1 &
#
# end
```

3.6.3 VP2 Sensor Manager: start_xm.vp2

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_xm.sqr19,v $
# $Author: inglis $
# $Date: 2005/02/22 17:25:17 $
#
source ./stb.cfg
#
pkill XM."$APP".VP2
#
sleep 2
#
./XM."$APP".VP2 –c xm.cfg.vp2 –n $DS $STB_FLAG $1 &
#
# end
```

3.6.4 Geomagnetic Manager: start_gm

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_gm,v $
# $Revision: 1.4 $
# $Date: 2005/02/22 17:26:34 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill GM."$APP"
#
sleep 2
#
cp ./geomagnetic_manager GM."$APP"
./GM."$APP" –c gm.cfg –n $DS $STB_FLAG $1 &
#
# end
```

3.6.5 Ping Manager

3.6.5.1 Default

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.2 Whale Warning

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.ww –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.3 Run 1

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
```

```
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.run1 –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.4 Run 2

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.run2 –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.5 Run 3

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.run3 –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.6 System Integration Test

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
```

```
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.sit –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.7 Don’s Personel Pinger

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.don –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.5.8 Standby

```
#!/bin/tcsh –f
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_pm.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:29 $
#
source ./stb.cfg
#
pkill PM."$APP"
#
sleep 2
#
cp ./ping_manager PM."$APP"
./PM."$APP" –c pm.cfg.don –n $DS $STB_FLAG –e 1 –x $1
#
# end
```

3.6.6 Ping Generator

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/start_stb.local,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:59 $
# $Author: inglis $
#
source ./stb.cfg.q302

# paspgs
ssh "pascam@paspgs" "cd /export/home/pascam/src/atass/IF/pgs_interface.061016/bin; start_pg "–e 2"&" &
# end
```

3.7 Operator Interface Scripts

3.7.1 Button Display: start_dd.kp

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".KP
#
sleep 2
#
cp ./data_display DD."$APP".KP
./DD."$APP".KP –c dd.cfg.kp –n $DS $STB_FLAG $1 &
#
# end
```

3.7.2 Track Display: start_dd.ts

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".TS
#
sleep 2
#
cp ./data_display DD."$APP".TS
./DD."$APP".TS –c dd.cfg.ts –n $DS $STB_FLAG $1 &
#
# end
```

3.7.3 Ping Display: start_dd.ps

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".PS
#
sleep 2
#
cp ./data_display DD."$APP".PS
./DD."$APP".PS –c dd.cfg.ps –n $DS $STB_FLAG $1 &
#
# end
```

3.7.4 DASM Lofargram: start_dd.da.lscan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".DA.LS
#
sleep 2
#
cp ./data_display DD."$APP".DA.LS
./DD."$APP".DA.LS –c dd.cfg.da.lscan –n $DS $STB_FLAG $1 &
#
# end
```

3.7.5 MANTA Lofargram: start_dd.ma.lscan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".MA.LS
#
sleep 2
#
cp ./data_display DD."$APP".MA.LS
./DD."$APP".MA.LS –c dd.cfg.ma.lscan –n $DS $STB_FLAG $1 &
#
# end
```

3.7.6 Sonobuoy Lofargram: start_dd.sb.lscan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".SB.LS
#
sleep 2
#
cp ./data_display DD."$APP".SB.LS
./DD."$APP".SB.LS –c dd.cfg.sb.lscan –n $DS $STB_FLAG $1 &
#
# end
```

3.7.7 DASM Ascan: start_dd.da.ascan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".DA.AS
#
sleep 2
#
cp ./data_display DD."$APP".DA.AS
./DD."$APP".DA.AS –c dd.cfg.da.ascan –n $DS $STB_FLAG $1 &
#
# end
```

3.7.8 MANTA Ascan: start_dd.ma.ascan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".MA.AS
#
sleep 2
#
cp ./data_display DD."$APP".MA.AS
./DD."$APP".MA.AS –c dd.cfg.ma.ascan –n $DS $STB_FLAG $1 &
#
# end
```

3.7.9 Sonobuoy Ascan: start_dd.sb.ascan

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
# source ./stb.cfg
source ./stb.cfg.tao
#
pkill DD."$APP".SB.AS
#
sleep 2
#
cp ./data_display DD."$APP".SB.AS
./DD."$APP".SB.AS –c dd.cfg.sb.ascan –n $DS $STB_FLAG $1 &
#
# end
```

3.8 Data Recording Start Scripts

3.8.1 Data Player: start_dp

```
#!/bin/tcsh -f
#
# $Source: /usr/local/cvsroot/src/atass/system/demo.1/start_ds,v $
# $Revision: 1.3 $
# $Date: 2003/01/21 14:42:32 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DP."$APP"
#
sleep 2
#
cp ./data_player DP."$APP"
./DP."$APP" -c dp.cfg -n $DS $STB_FLAG $1
#
# end
```

3.8.2 Data Logger: start_dl

```
#!/bin/tcsh -f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_dl,v $
# $Revision: 1.1 $
# $Date: 2003/11/26 17:35:09 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DL."$APP"
#
sleep 2
#
./DL."$APP" -c dl.cfg -n $DS $STB_FLAG $1 &
#
# end
```

3.8.3 Data Recorder: start_dr

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_dl,v $
# $Revision: 1.1 $
# $Date: 2003/11/26 17:35:09 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DR."$APP"
#
sleep 2
#
./DR."$APP" –c dr.cfg –n $DS $STB_FLAG $1 –x &
#
# end
```

3.8.4 DASM DAT Recording: start_dat.dasm

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_tr,v $
# $Revision: 1.1 $
# $Date: 2004/11/12 14:24:23 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DAT."$APP".DASM
#
sleep 2
#
cp ./dat16_recorder DAT."$APP".DASM
#
sleep 2
#
./DAT."$APP".DASM –c dat.cfg.dasm –n DS_DASM_IN $STB_FLAG $1 –x &
#
# end
```

3.8.5 MANTA DAT Recording: start_dat.manta

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_tr,v $
# $Revision: 1.1 $
# $Date: 2004/11/12 14:24:23 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DAT."$APP".MANTA
#
sleep 2
#
cp ./dat16_recorder DAT."$APP".MANTA
#
sleep 2
#
./DAT."$APP".MANTA –c dat.cfg.manta –n DS_MANTA_IN $STB_FLAG $1 –x &
#
# end
```

3.8.6 Sonobuoy DAT Recording: start_dat.sb

```
#!/bin/tcsh –f
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/start_tr,v $
# $Revision: 1.1 $
# $Date: 2004/11/12 14:24:23 $
# $Author: inglis $
#
source ./stb.cfg
#
pkill DAT."$APP".SB
#
sleep 2
#
cp ./dat16_recorder DAT."$APP".SB
#
sleep 2
#
./DAT."$APP".SB –c dat.cfg.sb –n DS_SB_IN $STB_FLAG $1 –x
#
# end
```

3.9 Utility Start Scripts

3.9.1 Data Editor

```
#!/bin/tcsh –f
#
# $Revision: 1.3 $
# $Source: /usr/local/cvsroot/src/atass/system/impact/start_sp.nuw,v $
# $Author: inglis $
# $Date: 2005/08/11 15:47:12 $
#
source ./stb.cfg
#
pkill DE."$APP"
#
sleep 2
#
./DE."$APP" –n $1 $STB_FLAG $2
# end
```

3.9.2 Java Data Editor

```
#!/bin/tcsh
#
source ./stb.cfg
#
java –cp javastbNoCoe.jar:stbSupport.jar –DORBInitRef.NameService=corbaloc:iiop:${NS}:${PORT}/NameService
com.gdc.javadataeditor.JDEConfiguration Files
```

4. Processor Configuration Files

4.1 Data Server Configuration Files

4.1.1 Nonacoustic Data Server Configuration Files

4.1.1.1 Nonacoustic Data Definition Configuration File: ds.cfg.nad.def

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.nad,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:39 $
# $Author: inglis $
#
$INCLUDE ds.cfg.nad.def
#
$GROUP
#         suggested
# group   records
#
PN_FD     1000      1000      $NBF
TD        2000      1000
ATD       2000      1000    # ais tracks
BTD       2000      1000    # sonobuoy tracks
STD       2000      1000    # sensor tracks
LTD       2000      1000    # localization tracks
RTD       2000      1000      # radar tracks
WF     10     10     # waveform data group
WT     10     10     # wavetrain data group
WP     10     10     # wavetrain data group
MV     10     10     # wavetrain data group
PG     10     10     # ping data group
SM     10     10     # system monitor
PS     10     10     # ping sequence
#
PF                  100       100
# end group list
#
$DATA
#
# name              group               rate      records   general axis    bits    size    value   res
WS                  PG        -1.0      500       1000      232       0
PD                  PG        -1.0      500       1000      232       0
PN                  PG        -1.0      500       1000      7960      0
PPN                 PG        -1.0      500       1000      7960      0
PS                  PS        -1.0      5000      1000      232       0
PT                  PS        -1.0      5000      1000      232       0
ARRAY               SD                  1.0       2000      1000      120       0
#
QUEST        TD          1.0     10000    10000          $TD    0
DAP         TD          1.0     10000    10000          $TD    0
```

```
#
SB           STD              1.0    5000        0         $TD    0
VP2                  STD      1.0    5000   0     $TD    0
DASM         STD              1.0    5000   0     $TD    0
MANTA         STD             1.0    5000   0     $TD    0
#
# OS_10S                      DTD      10.0    2000   0     128    0
#
DASM_NAD         SD           1.0    5000   0     184    0
MANTA_NAD     SD              1.0    5000   0     184    0
#
NADAS_DATA     SD               1.0    100    500    184    0
VP2_NAD                       SD           1.0    100    500    184    0
RAW_VP2_NAD     SD              1.0    100    500    184    0
VP2NAD_DATA     SD             1.0    100    500    184    0
# WINCH_DATA      SD              1.0    100    500    72  0
WINCH_NAD      SD           1.0    100    0    184      0
TIME_TG_DATA     SD                   0.25      100      0         56        0
TIME_MA_DATA     SD                   0.25      100      0         56        0
TIME_DA_DATA     SD                   0.25      100      0         56        0
TIME_TM_DATA     SD                   0.25      100      0         56        0
TIME_DATA        SD                   0.25      100      0         56        0
TIME_1S_DATA     SD                   1.0       100      0         56        0
TIME_5S_DATA     SD                   5.0       100      0         56        0
TIME_10S_DATA    SD                   10.0      100      0         56        0
TIME_60S_DATA    SD                   60.0      100      0         56        0
TIME_300S_DATA   SD                   300.0     100      0         56        0
PING_DATA        SD                   -1.0      100      0         8         0
ARRAY_DATA       SD                   -1.0      100      10        136       0
#
PS           SD           -1.0   500    1000   208    0
PC           SD           -1.0   500    1000   64     0
#
VP2_STATUS     SD           -1.0   5     0     80     0
#
GM                  SD          -1.0   500      0         96        0
GP                  SD          -1.0   500      0         $TD       0
#
CHAT          CHAT          -1     100    100    8192   0
MESSAGE        SC           -1     100    100    8192   0
#
# end data list
#
# end file
```

4.1.1.2 Nonacoustic Data Configuration File: ds.cfg.nad

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.nad,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:39 $
# $Author: inglis $
#
```

```
$INCLUDE ds.cfg.nad.def
#
$GROUP
#           suggested
# group  records
#
PN_FD    1000      1000      $NBF
TD       2000      1000
ATD      2000      1000    # ais tracks
BTD      2000      1000    # sonobuoy tracks
STD      2000      1000    # sensor tracks
LTD      2000      1000    # localization tracks
RTD      2000      1000      # radar tracks
WF      10      10      # waveform data group
WT      10      10      # wavetrain data group
WP      10      10      # wavetrain data group
MV      10      10      # wavetrain data group
PG      10      10      # ping data group
SM      10      10      # system monitor
PS      10      10     # ping sequence
#
PF                     100       100
# end group list
#
$DATA
#
# name              group               rate       records   general axis    bits    size    value   res
WS                  PG         −1.0       500       1000      232        0
PD                  PG         −1.0       500       1000      232        0
PN                  PG         −1.0       500       1000      7960       0
PPN                 PG         −1.0       500       1000      7960       0
PS                  PS         −1.0       5000      1000      232        0
PT                  PS         −1.0       5000      1000      232        0
ARRAY               SD                    1.0       2000      1000      120        0
#
QUEST          TD            1.0     10000    10000          $TD     0
DAP           TD            1.0     10000    10000          $TD     0
#
SB           STD               1.0     5000         0          $TD     0
VP2                 STD        1.0     5000    0      $TD     0
DASM          STD              1.0     5000    0      $TD     0
MANTA          STD             1.0     5000    0      $TD     0
#
# OS_10S                       DTD       10.0     2000    0       128     0
#
DASM_NAD          SD         1.0     5000    0      184     0
MANTA_NAD       SD         1.0     5000    0      184     0
#
NADAS_DATA      SD            1.0     100     500     184     0
VP2_NAD                      SD            1.0     100     500     184     0
RAW_VP2_NAD      SD            1.0     100     500     184     0
VP2NAD_DATA     SD            1.0     100     500     184     0
```

```
# WINCH_DATA      SD             1.0     100     500     72  0
WINCH_NAD       SD             1.0     100     0     184       0
TIME_TG_DATA     SD                   0.25       100       0           56          0
TIME_MA_DATA     SD                   0.25       100       0           56          0
TIME_DA_DATA     SD                   0.25       100       0           56          0
TIME_TM_DATA     SD                   0.25       100       0           56          0
TIME_DATA        SD                   0.25       100       0           56          0
TIME_1S_DATA     SD                   1.0        100       0           56          0
TIME_5S_DATA     SD                   5.0        100       0           56          0
TIME_10S_DATA    SD                   10.0       100       0           56          0
TIME_60S_DATA    SD                   60.0       100       0           56          0
TIME_300S_DATA SD                     300.0      100       0           56          0
PING_DATA        SD                   −1.0       100       0           8           0
ARRAY_DATA       SD                   −1.0       100       10          136         0
#
PS             SD            −1.0    500     1000    208     0
PC             SD            −1.0    500     1000    64      0
#
VP2_STATUS      SD             −1.0    5       0       80      0
#
GM                     SD           −1.0    500       0           96          0
GP                     SD           −1.0    500       0           $TD         0
#
CHAT           CHAT           −1      100     100     8192    0
MESSAGE         SC            −1      100     100     8192    0
#
# end data list
#
# end file
```

4.1.2 DASM Data Server Configuration Files

4.1.2.1 DASM Definition Configuration File: ds.cfg.dasm.def

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.nad.def,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:39 $
# $Author: inglis $
#
$DEFINE
MD        224          # message data
AD        216          # acoustic data header bytes
SD        136          # sensor data bytes
TD        256          # track data v2 bytes
BFR       10                   # beamform spectra ram records - needed for audio
RR        20                   # ram records
DR        0                    # disk records
CHR     10                     # ch ram records
CHD     10                     # ch disk record
BBR     200          # bb ram records
BBD     100          # bb disk record
CWR     100          # cw ram records
CWD     100          # cw disk record
FMRA    10                     # fm ram records
FMRB    10                     # fm ram records
FMR     10                     # fm ram records
FMD     10                     # fm disk record
NBR     100          # nb ram records
NBRA    100          # nb ram records
NBD     100          # nb disk record
TNR     100          # nb ram records
TND     100          # nb disk record
#
BBF     100                    # max bb features
CWF     10000
FMF     10000
NBF     100
#
BBR     2000
#
IR      0.75
IP02    0.125        # integration period 01
IP01    0.25         # integration period 01
IP0     0.5        # integration period 0
IP1     1.0        # integration period 1
IP2     2.0        # integration period 2
IP3     4.0        # integration period 3
IP4     8.0        # integration period 4

# end file
```

4.1.2.2 DASM Channel Data Server Configuration File: ds.cfg.dasm.ch

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.dasm.ch,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:37 $
# $Author: inglis $
#
$INCLUDE ds.cfg.dasm.def
#
$DATA
# name                    group    rate      records   general axis    bits    size    value   res
#
CHANNEL_SPECTRA           SP     0.75    $RR    $DR    $AD    2     64     2048    0.0     1.0     $CH     0.0     0.0
#
CH_TIME_SERIES    SP      0.75    $RR    $DR    $AD    2     64     1536    1024.0  1.0     $CH     0.0     0.0
#
CH_1_0_AD       CH_AD             $P0      $CHR    0    $AD     2     4      2048    0.0     1.0     $CH     0.0     0.0
CH_2_0_AD       CH_AD             $P4      $CHR    0    $AD     2     4      2048    0.0     1.0     $CH     0.0     0.0
#
CH_1_0_FLOAT    CH_FLOAT          $P0      $CHR1    0        $AD      2     32      2048  0.0      1.0      $CH       0.0
0.0
CH_2_0_FLOAT    CH_FLOAT          $P4      $CHR1    0        $AD      2     32      2048  0.0      1.0      $CH       0.0
0.0
#
# end data list
#
# end file
```

4.1.2.3 DASM Input Data Server Configuration File: ds.cfg.dasm.in

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.dasm.in,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:38 $
# $Author: inglis $
#
$INCLUDE ds.cfg.dasm.def
#
$DATA
# name                  group   rate    records general axis    bits    size    value   res
#
CHANNEL_TIME_SERIES     SP      $BR     $RR     $DR     $AD     2       32      $CH     0.0     0.0
        $FR     0.0     $TR
DAT32_CHANNEL_TIME_SERIES       SP      $BR     $RR     $DR     $AD     2       32      262     0.0
        0.0     $FR     0.0     $TR
#
DASM_NAD                SD      $BR     $RR     $DR     $SD     0
FWD_DASM_NAD            SD      $BR     $RR     $DR     $SD     0
AFT_DASM_NAD            SD      $BR     $RR     $DR     $SD     0
#
DASM_CFG                SD    −1.0    5     0     32     1     32     $CH    0.0    0.0
DASM_CFG_V2             SD    −1.0    5     0     32     1     96     $CH    0.0    0.0
#
READY           SD      $BR     $RR     $DR     56      0
#
# end data list
#
# end file
```

4.1.3 MANTA Data Server Configuration Files

4.1.3.1 MANTA Definition Configuration File: ds.cfg.manta.def

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.manta.def,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:39 $
# $Author: inglis $
#
$DEFINE
CH       240                # array channels
FR       1024               # frames
TR       0.000244140625  # time resolution
IR       0.75           # front end processing
# IR     0.5            # front end processing
BR       0.25         # block resolution
AD       216          # acoustic data header bytes
SD       184          # sensor data bytes
#
RR       20                 # ram records
DR       0                  # disk records
CHR    20                   # ch ram records
CHR1   100                  # ch ram records
CHR2   400                  # ch ram records
#
P04   0.03125
P03   0.0625
P02   0.125
P01   0.25
P0    0.5
P1    1.0
P2    2.0
P3    4.0
P4    8.0
P5    16.0
P6    32.0
P7    64.0
#
B5   128
B6   256
B7   512
#
R03      8.0
R02      4.0
R01      2.0
R0       1.0
R1       0.5
R2       0.25
R3       0.125
R4       0.0625
#
# end file
```

4.1.3.2 MANTA Channel Data Server Configuration File: ds.cfg.manta.ch

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.manta.ch,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:38 $
# $Author: inglis $
#
$INCLUDE ds.cfg.manta.def
#
$DATA
# name                    group    rate       records   general axis    bits    size    value   res
#
CHANNEL_SPECTRA          SP       $IR   $RR    $DR    $AD    2     64     2048    0.0    1.0    $CH    0.0    0.0
#
CH_TIME_SERIES   SP       $IR    $RR    $DR    $AD    2     64     1536    1024.0  1.0     $CH    0.0    0.0
#
CH_1_0_AD        CH_AD   0.5      $CHR2    0    $AD     2     4      2048    0.0    1.0    $CH     0.0    0.0
CH_2_0_AD        CH_AD   8.0      $CHR2    0    $AD     2     4      2048    0.0    1.0    $CH     0.0    0.0
#
CH_1_0_FLOAT     CH_FLOAT         0.5      $CHR1    0    $AD     2     32     2048    0.0    1.0    $CH     0.0    0.0
CH_2_0_FLOAT     CH_FLOAT         8.0      $CHR1    0    $AD     2     32     2048    0.0    1.0    $CH     0.0    0.0
#
# CH_1_0_NORM    CH_NORM                   0.5       $CHR1     0     $AD     2      32      2048    0.0     1.0     $CH
0.0     0.0
# CH_2_0_NORM    CH_NORM                   8.0       $CHR1     0     $AD     2      32      2048    0.0     1.0     $CH
0.0     0.0
#
# end data list
#
# end file
```

4.1.3.3 MANTA Input Data Server Configuration File: ds.cfg.manta.in

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.manta.in,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:39 $
# $Author: inglis $
#
$INCLUDE ds.cfg.manta.def
#
$DATA
# name                  group   rate    records general axis bits size  value res
#
CHANNEL_TIME_SERIES     SP      $BR     $RR     $DR     $AD     2       32      $CH    0.0    0.0    $FR
0.0     $TR
#
MANTA_NAD               SD      $BR     $RR     $DR     $SD     0
FWD_MANTA_NAD           SD      $BR     $RR     $DR     $SD     0
AFT_MANTA_NAD           SD      $BR     $RR     $DR     $SD     0
MANTA_CFG               SD      −1.0    5       0       32      1       32      $CH     0.0     0.0
MANTA_CFG_V2            SD      −1.0    5       0       32      1       96      $CH     0.0     0.0
#
READY                   SD      $BR     $RR     $DR     56      0
#
# end data list
#
# end file
```

4.1.4 Sonobuoy Data Server Configuration Files

4.1.4.1 Sonobuoy Data Definition Configuration File

Sonobuoy Data Definition Configuration File, 16 Channels, 40100: ds.cfg.sb.def.40100

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.sb.def,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:40 $
# $Author: inglis $
#
$DEFINE
AD    216         # acoustic data header bytes
CH       16               # channels
#
RR    20         # ram records
DR    0          # disk records
CHR    100        # ch ram records
CHR1    500       # ch ram records
#
BR       0.102144638404          # block rate - 4096/40100
SR       0.00002493765586035     # sample rate 1/40100
IR       0.612867830424          # input rate - 0.75*32768/40100
#
R0       1.22375488281   # resolution - 40100/32768
#
HF       1253.125        # heterodyne frequency - 1024*40100/32768
#
P0   0.408578554         # 0.5*32768/40100
P4   6.537256858         # 8.0*32768/40100
#
# end file
```

Sonobuoy Data Definition Configuration File, 16 Channels, 44100: ds.cfg.sb.def.44100

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.sb.def,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:40 $
# $Author: inglis $
#
$DEFINE
AD    216          # acoustic data header bytes
CH      16                # channels
#
RR    20           # ram records
DR    0            # disk records
CHR    100          # ch ram records
CHR1   500          # ch ram records
#
BR      0.0928798185941043083900226757369 61  # block rate – 4096/44100
SR      0.000022675736961451247165532879818594      # sample rate 1/44100
IR      0.557278911564625850340136054421 77    # input rate – 0.75*32768/44100
#
R0      1.3458251953125                             # resolution – 44100/32768
#
HF      1378.125                        # heterodyne frequency – 1024*44100/32768
#
P0    0.371519274376417233560090702947 85      # 0.5*32768/44100
P4    5.944308390022675736961451247165 5       # 8.0*32768/44100
#
# end file
```

Sonobuoy Data Definition Configuration File, 16 Channels, 88200: ds.cfg.sb.88200

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.sb.def,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:40 $
# $Author: inglis $
#
$DEFINE
AD     216          # acoustic data header bytes
CH       16                  # channels
#
RR     20           # ram records
DR     0            # disk records
CHR    100          # ch ram records
CHR1   500          # ch ram records
#
BR        0.04643990929705215 42          # block rate - 4096/88200
SR        0.00001133786848072562          # sample rate 1/40100
IR        0.27863945578231292517          # input rate - 0.75*32768/88200
#
R0        2.691650390625 # resolution - 88200/32768
#
HF        2756.25 # heterodyne frequency - 1024*88200/32768
#
P0    0.18575963718820861678    # 0.5*32768/88200
P4    2.97215419501133786848    # 8.0*32768/88200
#
# end file
```

4.1.4.2 Sonobuoy Input Configuration File: ds.cfg.sb.in

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.sb.in,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:40 $
# $Author: inglis $
#
$INCLUDE ds.cfg.sb.def
#
$DATA
# name              group             rate      records   general axis    bits    size    value   res
#
SB_TIME_SERIES  TS         $BR      $RR      $DR      $AD     2     32    $CH      0.0     0.0     4096 0.0     $SR
#
# end data list
#
# end file
```

4.1.4.3 Sonobuoy Channel Configuration File: ds.cfg.sb.ch

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ds.cfg.sb.ch,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:40 $
# $Author: inglis $
#
$INCLUDE ds.cfg.sb.def
#
$DATA
# name           group           rate      records   general axis   bits   size   value   res
#
CH_SPECTRA     SP                $IR       $RR    $DR    $AD    2     64    16384      0.0    $R0     $CH     0.0
0.0

CH_TIME_SERIES   SP              $IR       $RR    $DR    $AD    2     64    1536    $HF  $R0    $CH    0.0    0.0
#
CH_1_0_AD       CH_AD      $P0    $CHR1    $DR  $AD     2     4     2048   0.0    $R0    $CH    0.0    0.0
CH_2_0_AD       CH_AD      $P4    $CHR1    $DR  $AD     2     4     2048   0.0    $R0    $CH    0.0    0.0
#
CH_1_0_FLOAT    CH_FLOAT        $P0    $CHR    $DR        $AD     2     32     2048   0.0    $R0    $CH     0.0
0.0
CH_2_0_FLOAT    CH_FLOAT        $P4    $CHR    $DR        $AD     2     32     2048   0.0    $R0    $CH     0.0
0.0
#
# end data list
#
# end file
```

4.2 Signal Processing Configuration Files

4.2.1 DASM Signal Processing Configuration Files

4.2.1.1 DASM Channel Signal Processing Configuration File: sp.cfg.dasm.ch

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.dasm.ch,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:45 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$BAND$ CH_BD $INPUT$ CHANNEL_SPECTRA $OUTPUT$ CH_TIME_SERIES
#
$NARROWBAND$ CH_1_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_1_0_AD $OUTPUT$ CH_1_0_FLOAT $MODE$ 1
$NARROWBAND$ CH_2_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_2_0_AD $OUTPUT$ CH_2_0_FLOAT $MODE$ 1
#
#
# end
#
```

4.2.1.2 DASM Time Series Signal Processing Configuration File: sp.cfg.dasm.ts

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.dasm.ts,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:47 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$TIME_SERIES$ INPUT $INPUT$ CHANNEL_TIME_SERIES..DS_DASM_IN $OUTPUT$ CHANNEL_SPECTRA $TRACK$
QUEST..DS_NAD $SENSOR_TRACK$ DASM..DS_NAD $SENSOR$ DASM_NAD..DS_NAD $CHANNEL$ 0 $CHANNELS$ 192
$GAIN$ -70.0 $MODE$ 1
# $TIME_SERIES$ INPUT $INPUT$ CHANNEL_TIME_SERIES..DS_DASM_IN $OUTPUT$ CHANNEL_SPECTRA $CHANNEL$ 0
$CHANNELS$ 192 $GAIN$ 60.0 $MODE$ 1

#
#
# end
#
```

4.2.2 MANTA Signal Processing Configuration Files

4.2.2.1 MANTA Time Series Signal Processing Configuration File: sp.cfg.manta.ts

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.manta.ts,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:48 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$TIME_SERIES$ INPUT $INPUT$ CHANNEL_TIME_SERIES..DS_MANTA_IN $OUTPUT$ CHANNEL_SPECTRA $TRACK$
QUEST..DS_NAD $SENSOR_TRACK$ MANTA..DS_NAD $SENSOR$ MANTA_NAD..DS_NAD $CHANNEL$ 0 $CHANNELS$ 240
$GAIN$ -70.0 $MODE$ 1
#$TIME_SERIES$ INPUT $INPUT$ CHANNEL_TIME_SERIES..DS_MANTA_IN $OUTPUT$ CHANNEL_SPECTRA $CHANNEL$ 0
$CHANNELS$ 240 $GAIN$ 60.0 $MODE$ 1
#
# end
#
```

4.2.2.2 MANTA Channel Signal Processing Configuration File: sp.cfg.manta.ch

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.manta.ch.nb.1,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:47 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$BAND$ CH_BD $INPUT$ CHANNEL_SPECTRA $OUTPUT$ CH_TIME_SERIES
#
$NARROWBAND$ CH_1_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_1_0_AD $OUTPUT$ CH_1_0_FLOAT $MODE$ 1
$NARROWBAND$ CH_2_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_2_0_AD $OUTPUT$ CH_2_0_FLOAT $MODE$ 1
#
# end
#
```

4.2.3 Sonobuoy Signal Processing Configuration Files

4.2.3.1 Sonobuoy Channel Signal Processing Configuration File: sp.cfg.sb.ch

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.manta.ch.nb.1,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:47 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$BAND$ BAND $INPUT$ CH_SPECTRA $OUTPUT$ CH_TIME_SERIES
#
$NARROWBAND$ CH_1_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_1_0_AD $OUTPUT$ CH_1_0_FLOAT $MODE$ 1
$NARROWBAND$ CH_2_0_FLOAT $INPUT$ CH_TIME_SERIES $LSCAN$ CH_2_0_AD $OUTPUT$ CH_2_0_FLOAT $MODE$ 1
#
# end
#
```

4.2.3.2 Sonobuoy Channel Signal Processing Configuration File: sp.cfg.sb.ts

```
#
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/sp.cfg.sb.1,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:48 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE..DS_NAD
#
$TIME_SERIES$ INPUT $INPUT$ SB_TIME_SERIES..DS_SB_IN $OUTPUT$ CH_SPECTRA $GAIN$ 15.0
#
# end
#
```

4.3 Interface Configuration Files

4.3.1 Dasm: di.cfg

```
#
# dasm array interface configuration file
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/di.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:37 $
# $Author: inglis $
#
$ENDIAN$       1
$OUTPUT$              CHANNEL_TIME_SERIES 32 192 191 -1
$FWD_SENSOR_DATA$     FWD_DASM_NAD  249
$AFT_SENSOR_DATA$     AFT_DASM_NAD  5
$VCI$   102    0
$VCI$   103    64
$VCI$   104    128
$VCI$   105    192
# $MESSAGE_DATA$  MESSAGE..DS_NAD

# $MESSAGE_DATA$      DI_MESSAGE..DS_SC
# $OWN_TRACK_DATA$    OS..DS_NAD
# $SENSOR_TRACK_DATA$ DASM..ds_nad
#
# end
#
```

4.3.2 Manta: mi.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/mi.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:42 $
# $Author: inglis $
#
$ENDIAN$          1
# $MESSAGE_DATA$           MI_MESSAGE..DS_SC
#
# before 9 june 04
# $OUTPUT$                 CHANNEL_TIME_SERIES 46 50 0
# $OUTPUT$                 CHANNEL_TIME_SERIES 97 95 50
# $OUTPUT$                 CHANN
EL_TIME_SERIES 193 47 145
# $VCI$   301      0
# $VCI$   302      96
# $VCI$   303      192
# $FWD_SENSOR_DATA$        FWD_MANTA_NAD24
#
# on and after 9 june 04
#$OUTPUT$                  CHANNEL_TIME_SERIES 1 95 0
#$OUTPUT$                  CHANNEL_TIME_SERIES 97 47 95
#$VCI$    302      0
#$VCI$    303      96
#
#
#
# full manta
$OUTPUT$                   CHANNEL_TIME_SERIES 46 50 0
$OUTPUT$                   CHANNEL_TIME_SERIES 97 95 50
$OUTPUT$                   CHANNEL_TIME_SERIES 193 95 145
$FWD_SENSOR_DATA$          FWD_MANTA_NAD 24
$AFT_SENSOR_DATA$          AFT_MANTA_NAD 35
$VCI$     301      0
$VCI$     302      96
$VCI$     303      192
# $MESSAGE_DATA$  MESSAGE..DS_NAD
#
# end
#
```

4.3.3 Sonobuoy: bi.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/bi.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:33 $
# $Author: inglis $
#
# $REMOTE$ 192.168.0.143 50000
$REMOTE$ 192.168.0.13 50000

$OUTPUT$ SB_TIME_SERIES
# $MESSAGE_DATA$  MESSAGE..DS_NAD

# $OWN_TRACK$  RV2..DS_NAD
# $SENSOR_TRACK$        SB..DS_NAD
# $MESSAGE_DATA$  BI_MESSAGE..DS_SC $PERIOD$ 10.0
# $SENSOR_TRACK$        UAT..DS_NAD
#
# end
#
```

4.3.4 Sonobuoy NAD: sni.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ri.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:44 $
# $Author: inglis $
#
# keyword         name
$TIME_DATA$ TIME_1S_DATA
$TRACK_GROUP$ BTD
$PORT$ /dev/ttyS0
$BAUD$ 9600
#
# end
```

4.3.5 NADAS: ni.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ni.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:42 $
# $Author: inglis $
#
# keyword      name
$NADAS_DATA$  NADAS_DATA
# $MESSAGE_DATA$  MESSAGE..DS_NAD
# $PERIOD$ 30
$PORT$     /dev/ttyS0
$BAUD$     4800
$TRACK_DATA$  QUEST..DS_NAD
#
# end
```

4.3.6 Radar: ri.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/ri.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:44 $
# $Author: inglis $
#
# keyword      name
$OWN_SHIP_TRACK$   QUEST
$TRACK_GROUP$     TD
$PORT$     /dev/ttyS0
$BAUD$     4800
#
# $MESSAGE_DATA$  RI_MESSAGE..DS_SC $PERIOD$ 5.0
# $MESSAGE_DATA$  MESSAGE
# end
```

4.3.7 AIS: ai.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/IF/gps_interface/gi.cfg,v $
# $Revision: 1.1 $
# $Date: 2003/10/22 18:41:24 $
# $Author: inglis $
#
# keyword              name
# $TRACK_DATA$ SB      R2
# $MESSAGE_DATA$  MESSAGE..DS_NAD
$TIME_DATA$    TIME_1S_DATA
$PERIOD$    10.0
$PORT$      /dev/ttyS0
$BAUD$      38400
$TRACK_GROUP$   ATD.DS_NAD
# $STB_SERVER$  DS_FD
$USE_SERIAL_PORT$ 1
#
# end
```

4.4 Data Processing Configuration Files

4.4.1 Geomagnetic Manager Configuration File

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/gm.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:00:41 $
# $Author: inglis $
#
$GEOMAGNETIC_DATA$    GM      QUEST   0.1  60.0
$MESSAGE_DATA$  MESSAGE..DS_NAD
#
# end
```

4.4.2 Ping Manager Configuration Files

4.4.2.1 Default: pm.cfg

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary - default 0
# $ENDIAN$        1
#
$TIME_DATA$             TIME_1S_DATA
$MESSAGE$                       MESSAGE
#
$TRACK_GROUP$                            STD
# $WAVEFORM_GROUP$                       WF
$WAVETRAIN_GROUP$               WT
#
$PING_SCHEDULE_DATA$            WS
$PING_NOTIFICATION_DATA$        PN
$PING_PRENOTIFICATION_DATA$     PPN
#
# $PING_SEQUENCE_DATA$                   PS
# $PING_CONTROL_DATA$                    PC
#
# start up delay, period, number of times
$TIME$        5.0  20.0      90
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$        -30.0       1.0       3         0.0
#
$WAVETRAIN$     LFM_1150_100_2_WT         VP2       200.0
# $WAVETRAIN$     CW_1200_1_WT            VP2       200.0
# $WAVETRAIN$     HFM_1200_400_4_WT       VP2       210.0
#
# end
#
```

4.4.2.2 Whale Warning: pm.cfg.ww

```
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
$TIME_DATA$          TIME_1S_DATA
$MESSAGE$                MESSAGE
#
$TRACK_GROUP$                  STD
$WAVETRAIN_GROUP$        WT
#
$PING_SCHEDULE_DATA$       WS
$PING_NOTIFICATION_DATA$     PN
$PING_PRENOTIFICATION_DATA$   PPN
#
#        delay   period  pings
$TIME$     10.020.0     93
#        start   step    repeat  end
$GAIN$     -30.0       1.0      3       0.0
#
$WAVETRAIN$   LFM_1150_100_2_WT      VP2      216.0
#
# end
#
```

4.4.2.3 Run 1: pm.cfg.run1

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary - default 0
# $ENDIAN$      1
#
$TIME_DATA$        TIME_DATA
$MESSAGE$               MESSAGE
#
$TRACK_GROUP$                   STD
# $WAVEFORM_GROUP$              WF
$WAVETRAIN_GROUP$          WT
#
$PING_SCHEDULE_DATA$        WS
$PING_NOTIFICATION_DATA$      PN
$PING_PRENOTIFICATION_DATA$    PPN
#
# $PING_SEQUENCE_DATA$          PS
# $PING_CONTROL_DATA$           PC
#
# start up delay, period, number of times
$TIME$      10.090.0    4
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$      -30.0     1.0      3       0.0
#
$WAVETRAIN$    HFM_1050_200_0D500_WT   VP2     210.0
$WAVETRAIN$    HFM_1050_200_1_WT       VP2     210.0
$WAVETRAIN$    HFM_1050_200_2_WT       VP2     210.0
$WAVETRAIN$    HFM_1050_200_4_WT       VP2     210.0
#
# end
#
```

4.4.2.4 Run 2: pm.cfg.run2

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary - default 0
# $ENDIAN$        1
#
$TIME_DATA$            TIME_DATA
$MESSAGE$                      MESSAGE
#
$TRACK_GROUP$                          STD
# $WAVEFORM_GROUP$                     WF
$WAVETRAIN_GROUP$             WT
#
$PING_SCHEDULE_DATA$           WS
$PING_NOTIFICATION_DATA$        PN
$PING_PRENOTIFICATION_DATA$     PPN
#
# $PING_SEQUENCE_DATA$                 PS
# $PING_CONTROL_DATA$                  PC
#
# start up delay, period, number of times
$TIME$        10.090.0      4
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$        -30.0       1.0       3          0.0
#
$WAVETRAIN$    HFM_1100_300_0D500_WT  VP2       210.0
$WAVETRAIN$    HFM_1100_300_1_WT      VP2       210.0
$WAVETRAIN$    HFM_1100_300_2_WT      VP2       210.0
$WAVETRAIN$    HFM_1100_300_4_WT      VP2       210.0
#
# end
#
```

4.4.2.5 Run 3: pm.cfg.run3

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary – default 0
# $ENDIAN$      1
#
$TIME_DATA$        TIME_DATA
$MESSAGE$               MESSAGE
#
$TRACK_GROUP$                   STD
# $WAVEFORM_GROUP$              WF
$WAVETRAIN_GROUP$          WT
#
$PING_SCHEDULE_DATA$        WS
$PING_NOTIFICATION_DATA$      PN
$PING_PRENOTIFICATION_DATA$    PPN
#
# $PING_SEQUENCE_DATA$          PS
# $PING_CONTROL_DATA$           PC
#
# start up delay, period, number of times
$TIME$      10.090.0    4
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$      –30.0     1.0     3       0.0
#
$WAVETRAIN$    HFM_1050_100_0D500_WT  VP2       210.0
$WAVETRAIN$    HFM_1050_100_1_WT      VP2       210.0
$WAVETRAIN$    HFM_1050_100_2_WT      VP2       210.0
$WAVETRAIN$    HFM_1050_100_4_WT      VP2       210.0
#
# end
#
```

4.4.2.6 System Integration Test: pm.cfg.sit

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary - default 0
# $ENDIAN$        1
#
$TIME_DATA$            TIME_DATA
$MESSAGE$                   MESSAGE
#
$TRACK_GROUP$                        STD
# $WAVEFORM_GROUP$                   WF
$WAVETRAIN_GROUP$               WT
#
$PING_SCHEDULE_DATA$            WS
$PING_NOTIFICATION_DATA$        PN
$PING_PRENOTIFICATION_DATA$     PPN
#
# $PING_SEQUENCE_DATA$               PS
# $PING_CONTROL_DATA$                PC
#
# start up delay, period, number of times
$TIME$        10.090.0      4
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$        -30.0       1.0       3          0.0
#
$WAVETRAIN$    HFM_1450_125_1_WT       VP2      217.0
$WAVETRAIN$    HFM_1450_250_1_WT       VP2      217.0
$WAVETRAIN$    HFM_1450_500_1_WT       VP2      217.0
# $WAVETRAIN$    HFM_1450_1000_1_WT    VP2      217.0
#
# end
#
```

4.4.2.7 Standby: pm.cfg.standby

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary – default 0
# $ENDIAN$        1
#
$TIME_DATA$           TIME_1S_DATA
$MESSAGE$                  MESSAGE
#
$TRACK_GROUP$                        STD
# $WAVEFORM_GROUP$                   WF
$WAVETRAIN_GROUP$               WT
#
$PING_SCHEDULE_DATA$             WS
$PING_NOTIFICATION_DATA$           PN
$PING_PRENOTIFICATION_DATA$         PPN
#
# $PING_SEQUENCE_DATA$               PS
# $PING_CONTROL_DATA$                PC
#
# start up delay, period, number of times
$TIME$        10.0300.0    100
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$        –30.0      1.0       3        0.0
#
$WAVETRAIN$      LFM_1150_100_2_WT            VP2      190.0
# $WAVETRAIN$      CW_1200_1_WT               VP2      200.0
# $WAVETRAIN$      HFM_1200_400_4_WT          VP2      210.0
#
# end
#
```

4.4.2.8 Don’s Personal Pinger: pm.cfg.don

```
#
# ping manager configuration file
#
# $Revision: 1.1 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/pm.cfg,v $
# $Author: inglis $
# $Date: 2005/03/28 12:08:13 $
#
# force endian swap on all output if necessary – default 0
# $ENDIAN$        1
#
$TIME_DATA$           TIME_DATA
$MESSAGE$                 MESSAGE
#
$TRACK_GROUP$                       STD
# $WAVEFORM_GROUP$                  WF
$WAVETRAIN_GROUP$            WT
#
$PING_SCHEDULE_DATA$          WS
$PING_NOTIFICATION_DATA$        PN
$PING_PRENOTIFICATION_DATA$     PPN
#
# $PING_SEQUENCE_DATA$              PS
# $PING_CONTROL_DATA$               PC
#
# start up delay, period, number of times
$TIME$        10.030.0      4
# start gain dB, increase rate, number per step (before dB increase), stop level dB
# $GAIN$        –30.0       1.0       3         0.0
#
$WAVETRAIN$    CW_1800_1_WT     VP2       200.0
$WAVETRAIN$    LFM_1800_400_1_WT          VP2       200.0
$WAVETRAIN$    LFM_1800_N400_1_WT         VP2       200.0
$WAVETRAIN$    CW_1800_2_WT     VP2       200.0
#
# end
#
```

4.4.3 Waveform Manager Configuration File: wm.cfg

```
#
# STB waveform manager configuration file
#
# $Revision: 1.2 $
# $Source: /usr/local/cvsroot/src/atass/system/demo.3/wm.cfg,v $
# $Author: inglis $
# $Date: 2005/06/06 14:15:49 $
#
# keyword                   name
$WAVEFORM_GROUP$            WF
$WAVETRAIN_GROUP$          WT
$PING_SEQUENCE_GROUP$   PS
#
# passive waveform definitions
#
# keyword           name                         type      freq      bw        duration,s
$WAVEFORM$      LFM_1150_100_2_WF          LFM       1100.0  100.0     2.0
$WAVEFORM$      LFM_1800_400_1_WF          LFM       1800.0  400.0     1.0
$WAVEFORM$      LFM_1800_N400_1_WF         LFM       1800.0  -400.0     1.0
#
$WAVEFORM$      HFM_1050_100_0D500_WF  HFM       1050.0  200.0     0.5
$WAVEFORM$      HFM_1050_100_1_WF          HFM       1050.0  200.0     1.0
$WAVEFORM$      HFM_1050_100_2_WF          HFM       1050.0  200.0     2.0
$WAVEFORM$      HFM_1050_100_4_WF          HFM       1050.0  200.0     4.0
#
$WAVEFORM$      HFM_1050_200_0D500_WF  HFM       1050.0  200.0     0.5
$WAVEFORM$      HFM_1050_200_1_WF          HFM       1050.0  200.0     1.0
$WAVEFORM$      HFM_1050_200_2_WF          HFM       1050.0  200.0     2.0
$WAVEFORM$      HFM_1050_200_4_WF          HFM       1050.0  200.0     4.0
#
$WAVEFORM$      HFM_1100_300_0D500_WF  HFM       1100.0  300.0     0.5
$WAVEFORM$      HFM_1100_300_1_WF          HFM       1100.0  300.0     1.0
$WAVEFORM$      HFM_1100_300_2_WF          HFM       1100.0  300.0     2.0
$WAVEFORM$      HFM_1100_300_4_WF          HFM       1100.0  300.0     4.0
#
$WAVEFORM$      HFM_1200_N100_1D5_WF    HFM       1200.0  -100.0     1.5
$WAVEFORM$      HFM_1200_100_1_WF          HFM       1200.0  100.0     1.0
$WAVEFORM$      HFM_1200_200_1_WF          HFM     1200.0  200.0     1.0
$WAVEFORM$      HFM_1200_400_1_WF          HFM     1200.0  400.0     1.0
#
$WAVEFORM$      HFM_1200_100_4_WF          HFM       1200.0  100.0     4.0
$WAVEFORM$      HFM_1200_200_4_WF          HFM     1200.0  200.0     4.0
$WAVEFORM$      HFM_1200_400_4_WF          HFM     1200.0  400.0     4.0
#
$WAVEFORM$     HFM_1450_125_1_WF        HFM           1450.0  125.0     1.0
$WAVEFORM$     HFM_1450_250_1_WF        HFM           1450.0  250.0     1.0
$WAVEFORM$     HFM_1450_500_1_WF        HFM           1450.0  500.0     1.0
$WAVEFORM$     HFM_1450_1000_1_WF        HFM          1450.0  1000.0    1.0
#
$WAVEFORM$      CW_1200_1_WF       CW      1200.0  0.0     1.0
```

```
$WAVEFORM$     CW_1200_2_WF      CW      1200.0  0.0     2.0
$WAVEFORM$     CW_1200_4_WF      CW      1200.0  0.0     4.0
$WAVEFORM$     CW_1200_8_WF      CW      1200.0  0.0     8.0
#
$WAVEFORM$     CW_1800_1_WF      CW      1800.0  0.0     1.0
$WAVEFORM$     CW_1800_2_WF      CW      1800.0  0.0     2.0
#
# keyword            name of wave train   name of waveform        delay between waveforms
$WAVETRAIN$    LFM_1150_100_2_WT    LFM_1150_100_2_WF       0.0
#
$WAVETRAIN$    LFM_1800_400_1_WT    LFM_1800_400_1_WF       0.0
$WAVETRAIN$    LFM_1800_N400_1_WT   LFM_1800_N400_1_WF       0.0
#
$WAVETRAIN$    HFM_1050_100_0D500_WT    HFM_1050_100_0D500_WF   0.0
$WAVETRAIN$    HFM_1050_100_1_WT       HFM_1050_100_1_WF       0.0
$WAVETRAIN$    HFM_1050_100_2_WT       HFM_1050_100_2_WF       0.0
$WAVETRAIN$    HFM_1050_100_4_WT       HFM_1050_100_4_WF       0.0
#
$WAVETRAIN$    HFM_1050_200_0D500_WT    HFM_1050_200_0D500_WF   0.0
$WAVETRAIN$    HFM_1050_200_1_WT       HFM_1050_200_1_WF       0.0
$WAVETRAIN$    HFM_1050_200_2_WT       HFM_1050_200_2_WF       0.0
$WAVETRAIN$    HFM_1050_200_4_WT       HFM_1050_200_4_WF       0.0
#
$WAVETRAIN$    HFM_1100_300_0D500_WT    HFM_1100_300_0D500_WF   0.0
$WAVETRAIN$    HFM_1100_300_1_WT       HFM_1100_300_1_WF       0.0
$WAVETRAIN$    HFM_1100_300_2_WT       HFM_1100_300_2_WF       0.0
$WAVETRAIN$    HFM_1100_300_4_WT       HFM_1100_300_4_WF       0.0
#
$WAVETRAIN$    HFM_1200_N100_1D5_WT    HFM_1200_N100_1D5_WF       0.0
$WAVETRAIN$    HFM_1200_100_1_WT    HFM_1200_100_1_WF       0.0
$WAVETRAIN$    HFM_1200_200_1_WT    HFM_1200_200_1_WF       0.0
$WAVETRAIN$    HFM_1200_400_1_WT    HFM_1200_400_1_WF       0.0
#
$WAVETRAIN$    HFM_1200_100_4_WT    HFM_1200_100_4_WF       0.0
$WAVETRAIN$    HFM_1200_200_4_WT    HFM_1200_200_4_WF       0.0
$WAVETRAIN$    HFM_1200_400_4_WT    HFM_1200_400_4_WF       0.0
#
$WAVETRAIN$    HFM_1450_125_1_WT    HFM_1450_125_1_WF       0.0
$WAVETRAIN$    HFM_1450_250_1_WT    HFM_1450_250_1_WF       0.0
$WAVETRAIN$    HFM_1450_500_1_WT    HFM_1450_500_1_WF       0.0
$WAVETRAIN$    HFM_1450_1000_1_WT    HFM_1450_1000_1_WF      0.0
#
$WAVETRAIN$    CW_1200_1_WT    CW_1200_1_WF       0.0
$WAVETRAIN$    CW_1200_2_WT    CW_1200_2_WF       0.0
$WAVETRAIN$    CW_1200_4_WT    CW_1200_4_WF       0.0
$WAVETRAIN$    CW_1200_8_WT    CW_1200_8_WF       0.0
#
$WAVETRAIN$    CW_1800_1_WT    CW_1800_1_WF       0.0
$WAVETRAIN$    CW_1800_2_WT    CW_1800_2_WF       0.0
#
# end
```

4.4.4 Time Generator Configuration File: tg.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/tg.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:01:05 $
# $Author: inglis $
#
# $MESSAGE$ MESSAGE
#
# for live
$TIME_DATA$ TIME_DATA
# for sim
# $TIME_DATA$ TIME_DATA $PERIOD$ 1.0
# for both
$TIME_DATA$ TIME_1S_DATA $INPUT$ TIME_DATA
$TIME_DATA$ TIME_10S_DATA $INPUT$ TIME_1S_DATA
$TIME_DATA$ TIME_60S_DATA $INPUT$ TIME_10S_DATA
$TIME_DATA$ TIME_300S_DATA $INPUT$ TIME_60S_DATA
#
# end
```

4.4.5 Track Manager Configuration File: tm.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/tm.cfg,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:01:05 $
# $Author: inglis $
#
$MESSAGE$ MESSAGE
#
$TRACK_GROUP$ LTD
#
$EXTRAPOLATE_FLAG$   1
#
$INPUT_TIME_DATA$ TIME_10S_DATA
#
# end
```

4.4.6 Sensor Manager Configuration Files

4.4.6.1 DASM Sensor Manager Configuration File: xm.cfg.dasm

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/xm.cfg.dasm,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:01:07 $
# $Author: inglis $
#
# $MESSAGE$            MESSAGE
#
$INPUT_TIME$             TIME_1S_DATA
$GEOMAGNETIC$                    GM
# $WINCH$                WINCH_NAD
$INPUT_TRACK$            QUEST
$OUTPUT_TRACK$                   DASM
$INPUT_SENSOR$           FWD_DASM_NAD..DS_DASM_IN       HDTV
$INPUT_SENSOR$           AFT_DASM_NAD..DS_DASM_IN       HDV
$OUTPUT_SENSOR$                  DASM_NAD
$HEADING_AVERAGES$      1
#
# overrides
# $SOUND_SPEED$                  1505.0
$SALINITY$               32.0
$OFFSET$                 59.0
$SCOPE$                  263.0
#
# end
```

4.4.6.2 MANTA Sensor Manager Configuration File: xm.cfg.manta

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/xm.cfg.manta,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:01:07 $
# $Author: inglis $
#
# $MESSAGE$            MESSAGE
#
$INPUT_TIME$               TIME_1S_DATA
$GEOMAGNETIC$                      GM
# $WINCH$                  WINCH_NAD
$INPUT_TRACK$              QUEST
$OUTPUT_TRACK$                     MANTA
#$INPUT_SENSOR$                     FWD_MANTA_NAD..DS_MANTA_IN    HDV
$INPUT_SENSOR$             FWD_MANTA_NAD..DS_MANTA_IN    HTV
$INPUT_SENSOR$             AFT_MANTA_NAD..DS_MANTA_IN    HTD
$OUTPUT_SENSOR$                    MANTA_NAD
$HEADING_AVERAGES$     1
#
# overrides
$SOUND_SPEED$              1505.0
$SALINITY$                 32.0
$OFFSET$                   186.0
$SCOPE$                    263.0
#
# end
```

4.4.6.3 VP2 Sensor Manager Configuration File: xm.cfg.vp2

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q283/xm.cfg.vp2,v $
# $Revision: 1.1 $
# $Date: 2004/09/22 14:01:07 $
# $Author: inglis $
#
# $MESSAGE$            MESSAGE
#
$INPUT_TIME$               TIME_1S_DATA
$INPUT_TRACK$              QUEST
$OUTPUT_TRACK$                     VP2
# $INPUT_SENSOR$                   RAW_VP2_NAD
$OUTPUT_SENSOR$                    VP2_NAD
#
# overrides
$SOUND_SPEED$              1505.0   # m/s
$DEPTH$                    92.0     # metres
$SALINITY$                 38.0     # ppt
$SCOPE$                    0.0
$OFFSET$                   0.0
#
# end
```

4.5 Operator Interface Configuration Files

4.5.1 Chat Configuration File: dd.cfg.chat

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
# $MESSAGE$ MESSAGE
#
# $COLOUR$ 000 000 000 background
# $COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
#
$TIME$ TIME_1S_DATA
#
$CHAT_DATA$ CHAT
#
$NAME$ lab
$FRAMES$ 2 1
$POSITION$ 1 1
$SIZE$  680 150
$TITLE$ "CHAT LAB"
$CHAT$ CHAT
#
# end
```

4.5.2 Ping Summary Configuration File: dd.cfg.ps

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
$COLOUR$ 40 40 40 text
$COLOUR$ 220 220 220 select
$COLOUR$ 200 200 200 highlight_on
$COLOUR$ 160 160 200 on
$COLOUR$ 140 140 140 highlight
$COLOUR$ 120 120 140 active
$COLOUR$ 100 100 120 inactive
$COLOUR$ 080 080 100 background
#
$MESSAGE$ MESSAGE
#
$KEY$ $TOGGLE$  "QUEST TRACK"    EDIT              0              1              ""
"<component>track_plotter<event>show<group>TD<property>symbol"
"<component>track_plotter<event>hide<group>TD<property>symbol"
"<component>track_plotter<group>TD<property>symbol<state>on"
"<component>track_plotter<group>TD<property>symbol<state>off"
$KEY$ $TOGGLE$  "SENSOR TRACKS"EDIT              1              1              ""
"<component>track_plotter<event>show<group>STD<property>symbol"
"<component>track_plotter<event>hide<group>STD<property>symbol"
"<component>track_plotter<group>STD<property>symbol<state>on"
"<component>track_plotter<group>STD<property>symbol<state>off"
$KEY$ $TOGGLE$  "AIS TRACKS"     EDIT              2              0              ""
"<component>track_plotter<event>show<group>ATD<property>symbol"
"<component>track_plotter<event>hide<group>ATD<property>symbol"
"<component>track_plotter<group>ATD<property>symbol<state>on"
"<component>track_plotter<group>ATD<property>symbol<state>off"
$KEY$ $TOGGLE$  "RADAR TRACKS"   EDIT              3              0              ""
"<component>track_plotter<event>show<group>RTD<property>symbol"
"<component>track_plotter<event>hide<group>RTD<property>symbol"
"<component>track_plotter<group>RTD<property>symbol<state>on"
"<component>track_plotter<group>RTD<property>symbol<state>off"
$KEY$ $TOGGLE$  "BUOY TRACKS"    EDIT              4              0              ""
"<component>track_plotter<event>show<group>BTD<property>symbol"
"<component>track_plotter<event>hide<group>BTD<property>symbol"
"<component>track_plotter<group>BTD<property>symbol<state>on"
"<component>track_plotter<group>BTD<property>symbol<state>off"
#
# $KEY$ $TOGGLE$  "TIME"                EDIT 6 0 "" "TIME EVENT START" "TIME EVENT STOP" "TIME STATE ON"
"TIME STATE OFF"
# $KEY$ $PULSE$   "TIME STEP"      EDIT 7 0 "" "TIME EVENT STEP" "" "" ""
#
$KEY$ $TOGGLE$  "QUEST PATH"     EDIT 9 1 "" "<component>track_plotter<event>show<group>TD<property>path"
"<component>track_plotter<event>hide<group>TD<property>path"
```

```
"<component>track_plotter<group>TD<property>path<state>on"
"<component>track_plotter<group>TD<property>path<state>off"
$KEY$ $TOGGLE$  "SENSOR PATHS"  EDIT                    10                    1                     ""
"<component>track_plotter<event>show<group>STD<property>path"
"<component>track_plotter<event>hide<group>STD<property>path"
"<component>track_plotter<group>STD<property>path<state>on"
"<component>track_plotter<group>STD<property>path<state>off"
$KEY$ $TOGGLE$  "AIS PATHS"     EDIT                    11                    1                     ""
"<component>track_plotter<event>show<group>ATD<property>path"
"<component>track_plotter<event>hide<group>ATD<property>path"
"<component>track_plotter<group>ATD<property>path<state>on"
"<component>track_plotter<group>ATD<property>path<state>off"
$KEY$ $TOGGLE$  "RADAR PATHS"   EDIT                    12                    1                     ""
"<component>track_plotter<event>show<group>RTD<property>path"
"<component>track_plotter<event>hide<group>RTD<property>path"
"<component>track_plotter<group>RTD<property>path<state>on"
"<component>track_plotter<group>RTD<property>path<state>off"
$KEY$ $TOGGLE$  "BUOY PATHS"    EDIT                    13                    1                     ""
"<component>track_plotter<event>show<group>BTD<property>path"
"<component>track_plotter<event>hide<group>BTD<property>path"
"<component>track_plotter<group>BTD<property>path<state>on"
"<component>track_plotter<group>BTD<property>path<state>off"
#
# $KEY$ $PULSE$  "CW 1200 1"          EDIT 9 0 "" "" "" "" ""
# $KEY$ $PULSE$  "HFM 1200 400 1"     EDIT 10 0 "" "" "" "" ""
# $KEY$ $PULSE$  "HFM 1200 400 2"     EDIT 11 0 "" "" "" "" ""
#
$KEY$ $PULSE$   "PING PULSE"   EDIT 15 0 "" "PING PULSE ON" "PING PULSE OFF" "" ""
$KEY$ $TOGGLE$  "PING"        EDIT 16 0 "" "PING ON" "PING OFF" "PING ON" "PING OFF"
#
# $KEY$ $PULSE$  "EXIT ALL"    EDIT 16 0 "" "<event>exit"
# $KEY$ $PULSE$  "EXIT DD"     EDIT 17 0 "" "<component>data_display<event>exit"
#
# $FRAMES$ 7 5
$FRAMES$ 2 9
$POSITION$ 0 840
# $POSITION$ 0 512
# $SIZE$  384 512
$SIZE$  640 150
#
$TIME$ TIME_1S_DATA
$TITLE$ "BUTTON"
$BUTTON$ MESSAGE
#
# end
```

4.5.3 Track Manager Configuration File: dd.cfg.ts

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
##############################
#
# $SIZE$  400 512
#
# $MENU$
#
##############################
#
# $COLOUR$ 000 000 000 background
# $COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
#
# $FRAMES$  5 1
$POSITION$  1 300
$SIZE$  680 240
$WAVETRAIN$ PPN..DS_NAD
$TITLE$ "PING SUMMARY"
$PING_SUMMARY$
#
$TIME$ TIME_1S_DATA
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.4 Button Manager Configuration File: dd.cfg.kp

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
#############################
#
$SIZE$  400 512
#
$MENU$
#
#############################
#
# $FRAMES$  1 1
$POSITION$  1 600
$SIZE$  680 240
$TITLE$ "TRACK SUMMARY"
$TRACK_SUMMARY$
#
$TIME$ TIME_1S_DATA
#
$TRACK_GROUP$ TD.DS_NAD
$TRACK_GROUP$ ATD.DS_NAD
$TRACK_GROUP$ BTD.DS_NAD
$TRACK_GROUP$ STD.DS_NAD
$TRACK_GROUP$ RTD.DS_NAD
#
#$TRACK_GROUP$ TD.DS_SIM
#$TRACK_GROUP$ ATD.DS_SIM
#$TRACK_GROUP$ STD.DS_SIM
#$TRACK_GROUP$ RTD.DS_SIM
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.5 Dasm Lofargram Configuration File: dd.cfg.da.lscan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
##############################
#
$SIZE$  400 512
#
$MENU$
#
##############################
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $BUTTON_GROUP$ FTI.DS_KP
# $FEATURE$ BEARING_FD..DS_OP
#
# $FEATURE_GROUP$ BB_HF_FD.DS_FD
# $FEATURE_GROUP$ NB_HF_FD.DS_FD $CREATE$ 1 $ROOT$ NH
#
# $FRAMES$ 7 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$FRAMES$ 1 1
$POSITION$ 1 1
$SIZE$  600 300
# $LSCAN$ HF_NB_0_AD..DS_NB_HF
$TITLE$ DASM_LOFARGRAM
$LSCAN$ CH_1_0_AD..DS_DASM_CH $LINE_CURSOR$ 1
```

```
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.6 Dasm Ascan Configuration File: dd.cfg.da.ascan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
##############################
#
$SIZE$  250 512
#
$MENU$
#
# $MESSAGE$ MESSAGE
#
##############################
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $BUTTON_GROUP$ FTI.DS_KP
# $FEATURE$ BEARING_FD..DS_OP
#
# $FEATURE_GROUP$ BB_HF_FD.DS_FD
# $FEATURE_GROUP$ NB_HF_FD.DS_FD $CREATE$ 1 $ROOT$ NH
#
# $FRAMES$ 7 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$TITLE$ DASM_ASCAN
$FRAMES$ 3 1
$POSITION$ 1 1
$SIZE$  600 500
```

```
# $LSCAN$ HF_NB_0_AD..DS_NB_HF
$ASCAN$ CH_1_0_FLOAT..DS_DASM_CH $DB$ 1 $YMINIMUM$ 0.0 $YMAXIMUM$ 180.0 $AUTO$ 0
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.7 Manta Lofargram Configuration File: dd.cfg.ma.lscan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
###############################
#
$SIZE$  250 512
#
$MENU$
#
###############################
#
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $FRAMES$ 7 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$FRAMES$ 1 1
$POSITION$ 1 1
$SIZE$  600 300
$TITLE$ MANTA_LOFARGRAM
$LSCAN$ CH_1_0_AD..DS_MANTA_CH
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.8 Manta Ascan Configuration File: dd.cfg.ma.ascan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
##############################
#
$SIZE$  250 512
#
$MENU$
#
# $MESSAGE$ MESSAGE
#
##############################
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $BUTTON_GROUP$ FTI.DS_KP
# $FEATURE$ BEARING_FD..DS_OP
#
# $FEATURE_GROUP$ BB_HF_FD.DS_FD
# $FEATURE_GROUP$ NB_HF_FD.DS_FD $CREATE$ 1 $ROOT$ NH
#
# $FRAMES$ 7 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$TITLE$ MANTA_ASCAN
$FRAMES$ 3 1
$POSITION$ 1 1
$SIZE$  600 500
# $LSCAN$ HF_NB_0_AD..DS_NB_HF
```

```
$ASCAN$ CH_1_0_FLOAT..DS_MANTA_CH $DB$ 1 $YMINIMUM$ 0.0 $YMAXIMUM$ 180.0 $AUTO$ 0
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.9 Sonobuoy Lofargram Configuration File: dd.cfg.sb.lscan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
###############################
#
$SIZE$  250 512
#
$MENU$
#
###############################
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
# $COLOUR$ 220 220 220 select
# $COLOUR$ 080 080 080 highlight
# $COLOUR$ 160 160 160 highlight_on
# $COLOUR$ 120 120 120 on
# $COLOUR$ 040 040 040 active
# $COLOUR$ 030 030 030 inactive
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $BUTTON_GROUP$ FTI.DS_KP
# $FEATURE$ BEARING_FD..DS_OP
#
# $FEATURE_GROUP$ BB_HF_FD.DS_FD
# $FEATURE_GROUP$ NB_HF_FD.DS_FD $CREATE$ 1 $ROOT$ NH
#
# $FRAMES$ 7 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$FRAMES$ 1 1
$POSITION$ 1 1
$SIZE$  1200 300
# $LSCAN$ HF_NB_0_AD..DS_NB_HF
$TITLE$ "SONOBUOY LOFARGRAM"
$LSCAN$ CH_1_0_AD..DS_SB_CH $LINE_CURSOR$ 1
```

```
#
# $MESSAGE$ MESSAGE
#
# end
```

4.5.10 Sonobuoy Ascan Confguration File: dd.cfg.sb.ascan

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dd.cfg.nuw,v $
# $Revision: 1.2 $
# $Date: 2005/06/13 19:50:14 $
# $Author: inglis $
#
##############################
#
$SIZE$  250 512
#
$MENU$
#
# $MESSAGE$ MESSAGE
#
##############################
#
$COLOUR$ 000 000 000 background
$COLOUR$ 120 120 120 text
$COLOUR$ 150 150 150 cursor
$COLOUR$ 220 220 220 cursor_text
$COLOUR$ 220 220 220 symbol
#
# $COLOUR$ 000 040 000 acoustic_colour_0
# $COLOUR$ 000 060 000 acoustic_colour_1
# $COLOUR$ 000 100 000 acoustic_colour_2
# $COLOUR$ 080 140 000 acoustic_colour_3
# $COLOUR$ 120 180 000 acoustic_colour_4
# $COLOUR$ 160 200 000 acoustic_colour_5
# $COLOUR$ 180 220 000 acoustic_colour_6
# $COLOUR$ 200 240 000 acoustic_colour_7
#
# $BUTTON_GROUP$ FTI.DS_KP
# $FEATURE$ BEARING_FD..DS_OP
#
# $FEATURE_GROUP$ BB_HF_FD.DS_FD
# $FEATURE_GROUP$ NB_HF_FD.DS_FD $CREATE$ 1 $ROOT$ NH
#
# $FRAMES$ 4 1
$FRAMES$ 1 1
# $POSITION$ 1 1030
# $SIZE$  1200 700
$SIZE$  1200 200
$TITLE$ "SONOBUOY ASCAN"
# $FRAMES$ 1 1
$POSITION$ 1 1
#$SIZE$  600 500
$ASCAN$ CH_1_0_FLOAT..DS_SB_CH $DB$ 1 $YMINIMUM$ 0.0 $YMAXIMUM$ 180.0 $AUTO$ 0
#
# $MESSAGE$ MESSAGE
#
# end
```

4.6 Data Storage Configuration Files

4.6.1 Data Player (STBIO): dp.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/impact/dp.cfg.impact,v $
# $Revision: 1.1 $
# $Date: 2005/06/08 15:49:20 $
# $Author: inglis $
#
$TIME_STEP$   0.25
# $TIME_LIMIT$ 30.00
$TIME_FACTOR$ 0.5
# $TIME_FACTOR$ 2.0
$REPEAT$ 1
#
# $SERVER$ DS_NAD
#
$FILE$ /media/data1/dr/q302_01JAN70_010353.dl

#
$MESSAGE$ MESSAGE
$REFERENCE$ TIME_DATA
#
# end
```

4.6.2 Data Recorder (STBIO): dr.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/dl.cfg,v $
# $Revision: 1.4 $
# $Date: 2004/03/04 13:52:19 $
# $Author: inglis $
#
# keyword                    name
#$DIRECTORY$ ./../data.q302/dr
$DIRECTORY$ /media/data1/dr
$PREFIX$ q302

$MESSAGE$ MESSAGE..DS_NAD
$TIME$ TIME_1S_DATA..DS_NAD
$DURATION$ 120.0

$GROUP$ TD.DS_NAD
$GROUP$ ATD.DS_NAD
$GROUP$ BTD.DS_NAD
$GROUP$ LTD.DS_NAD
$GROUP$ STD.DS_NAD
$GROUP$ RTD.DS_NAD

$GROUP$   SD.DS_NAD
#
# $DATA$ TIME_DATA..DS_NAD
# $DATA$ TIME_1S_DATA..DS_NAD
# $DATA$ NADAS_DATA..DS_NAD
# $DATA$ DASM_NAD..DS_NAD
# $DATA$ MANTA_NAD..DS_NAD
# $DATA$ WINCH_NAD..DS_NAD
#
$DATA$          PN..DS_NAD
$DATA$          PPN..DS_NAD
#
$DATA$          CHANNEL_TIME_SERIES..DS_DASM_IN
$DATA$          AFT_DASM_NAD..DS_DASM_IN
$DATA$          FWD_DASM_NAD..DS_DASM_IN
#
$DATA$          CHANNEL_TIME_SERIES..DS_MANTA_IN
$DATA$          AFT_MANTA_NAD..DS_MANTA_IN
$DATA$          FWD_MANTA_NAD..DS_MANTA_IN
#
$DATA$          SB_TIME_SERIES..DS_SB_IN
#
# end
```

4.6.3 Data Logger (Text): dl.cfg

```
#
# $Source: /usr/local/cvsroot/src/atass/system/q268.v2/dl.cfg,v $
# $Revision: 1.4 $
# $Date: 2004/03/04 13:52:19 $
# $Author: inglis $
#
# $MESSAGE$ MESSAGE
#
#$DIRECTORY$ ./../data.q302/dl
$DIRECTORY$ /media/data1/dl
#
$TRACK_GROUP$               TD.DS_NAD
$TRACK_GROUP$               ATD.DS_NAD
$TRACK_GROUP$               BTD.DS_NAD
$TRACK_GROUP$               RTD.DS_NAD
$TRACK_GROUP$               STD.DS_NAD
#
$WAVETRAIN_V4_DATA$   PN
$WAVETRAIN_V4_DATA$   PPN
#
$SENSOR_DATA$ VP2_NAD
$SENSOR_DATA$ RAW_VP2_NAD
#
$SENSOR_DATA$ AFT_MANTA_NAD..DS_MANTA_IN
$SENSOR_DATA$ FWD_MANTA_NAD..DS_MANTA_IN
$SENSOR_DATA$ MANTA_NAD
#
$SENSOR_DATA$ AFT_DASM_NAD..DS_DASM_IN
$SENSOR_DATA$ FWD_DASM_NAD..DS_DASM_IN
$SENSOR_DATA$ DASM_NAD
#
$TEXT_DATA$ CHAT
#
# end
```

4.6.4 DAT Configuration Files

4.6.4.1 DASM DAT: dat.cfg.dasm

```
#
# $Source$
# $Revision$
# $Date$
# $Author$
#
$COMMENT$ Q302 dasm time series
$PREFIX$ Q302_DASM
$DATA$ CHANNEL_TIME_SERIES
$X_AXIS$ 1    # 1 – time series, 2 – frequency
$Y_AXIS$ 1    # 1 – linear, 2 – power, 3 – dB
$HEADER$ 2    # 1 – dat16, 2 – dat32
#
$TRANSPOSE$ 1
#$DIRECTORY$ ./../data.q302/dat
$DIRECTORY$ /media/data1/dat
$WRITE$
$DURATION$ 20
#
# end
```

4.6.4.2 MANTA DAT: dat.cfg.manta

```
#
# $Source$
# $Revision$
# $Date$
# $Author$
#
$COMMENT$ Q302 manta time series
$PREFIX$ Q302_MANTA
$DATA$ CHANNEL_TIME_SERIES
$X_AXIS$ 1    # 1 – time series, 2 – frequency
$Y_AXIS$ 1    # 1 – linear, 2 – power, 3 – dB
$HEADER$ 2    # 1 – dat16, 2 – dat32
#
$TRANSPOSE$ 1
#$DIRECTORY$ ./../data.q302/dat
$DIRECTORY$ /media/data1/dat
$WRITE$
$DURATION$ 20
#
# end
```

4.6.4.3 Sonobuoy DAT: dat.sfg.sb

```
#
# $Source$
# $Revision$
# $Date$
# $Author$
#
$COMMENT$ Q302 sb time series
$PREFIX$ Q302_SB
$DATA$ SB_TIME_SERIES
$X_AXIS$ 1    # 1 - time series, 2 - frequency
$Y_AXIS$ 1    # 1 - linear, 2 - power, 3 - dB
$HEADER$ 2    # 1 - dat16, 2 - dat32
#
$TRANSPOSE$ 1
#$DIRECTORY$ ./../data.q302/dat
$DIRECTORY$ /media/data1/dat
$WRITE$
$DURATION$ 20
#
# end
```

4.7 Utility Configuration Files

5. Software Procedures

5.1 Bootup Procedure

5.1.1 Verify IP address

a. Open a terminal and run: ifconfig –a; and

b. Verify on each machine that the ip address matches those listed in /etc/hosts file on each machine.

5.1.2 Verify SSH

Run ssh –X <hostname> to each machine from each machine. If this is the first time the machines are setup, you will be prompted to accept the public key from the remote machine to your local authorized host file located in $HOME/.ssh/authorized_keys.

As long as the public and private key pairs match along with the authorized_keys file, you can simply copy the .ssh fold and its content to each machine to grant access. Thus you do not need to generate unique key for each machine.

5.1.3 Set Serial Ports

Depending on the user accounts access right, you will need to chmod the serial port on each machine for any device that requires its use.

Run: chmod 0777 /dev/ttyS0

Where ttyS0 is serial port 0.

5.1.4 Verify NTP

Make sure that each host has the following ntp.conf file in /etc. This file will allow the ntp clock to synchronize to paspgs.

As super user or root run:

```
/etc/init.d/ntp-server stop
/etc/init.d/ntpdate paspgs
/etc/init.d/ntp-server start
```

The ntpdate command will force an immediate synchonrization to the host machine povided. If this is not done and the local and host machine differ by more than an hour, ntp server will NOT keep sychonization.

```
# /etc/ntp.conf, configuration for ntpd

# ntpd will use syslog() if logfile is not defined
#logfile /var/log/ntpd

driftfile /var/lib/ntp/ntp.drift
statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable


# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example


# pool.ntp.org maps to more than 100 low-stratum NTP servers.
# Your server will pick a different set every time it starts up.
#  *** Please consider joining the pool! ***
#  ***  <http://www.pool.ntp.org/#join>  ***
#server pool.ntp.org

# ... and use the local system clock as a reference if all else fails
# NOTE: in a local network, set the local stratum of *one* stable server
# to 10; otherwise your clocks will drift apart if you lose connectivity.
server 127.127.1.0
fudge 127.127.1.0 stratum 13

# By default, exchange time with everybody, but don't allow configuration.
# See /usr/share/doc/ntp-doc/html/accopt.html for details.
restrict default kod notrap nomodify nopeer noquery

# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1 nomodify

# Clients from this (example!) subnet have unlimited access,
# but only if cryptographically authenticated
#restrict 192.168.123.0  mask  255.255.255.0 notrust

# If you want to provide time to your local subnet, change the next line.
# (Again, the address is an example only.)
#broadcast 192.168.123.255

# If you want to listen to time broadcasts on your local subnet,
# de-comment the next lines. Please do this only if you trust everybody
# on the network!
#disable auth
#broadcastclient

server paspgs
```

5.1.5 Verify ATM driver

The ATM driver on Poweredge and Goshawka are automatically loaded after the module probe command has been run the first time as superuser or root.

To verify that the atm driver is loaded. Run

Lsmod | grep he

Where “he” is the name of the driver. On the older troll machines fore_200e cards where used instead.

If the driver is not loaded, as root, type

Modprobe he

Type uname –a to get the <kernal version>.

Type /lib/modules/<kernal version>/kernel/drivers/atm and look for the driver name. In ubuntua atm is a part of the kernel.

Also, it may be necessary to set the node privileges for the atm device port.

Run mknod /dev/<device name>.

Then chmod the port such that the user may read and write to the device.

Another place to look if problem still persist is the /etc/conf.modules or /etc/modules. The device name should show up in this file if it is installed properly.

5.1.6 Startup Procedure

To start up most of the Q302 processes, go to the host stbdisplay.

Open a terminal and type cd stb.q302.

Run start_stb.q302.

This will call a series of remote and local scripts to start up the processes on each machine.

Note that Ping Manager and Ping Generator on paspgs should be started manually, and only used when required.

All of the local processes on a specific proc may be run using start_stb.proc.<N> where N is the designated proc number found in $HOME/stb.q302/stb.cfg.q302 file.

5.1.7 Verify the following

Confirm that ds.cfg.sb.def file has the field CH set to the number of channels setup on EDAC. This is presently hardcoded for the EDAC at 16 channels.

Edit the sensor manager files xm.cfg.dasm, xm.cfg.manta, and xm.cfg.vp2 for cable scope and depth settings. Also monitor the data logger file on bigxeon64 in /media/data1/dl/ using the “tail –f <filename>” command. Verfiy that NAD is being reported correctly.

5.2 Recording Procedure

By default, Data Logger is always running when the master script start_stb.q302 is run. Data Logger will log files persently to bigxeon64 on the mount USB 2.0 IDE drive at /media/data1/dl/. All recording of any kind is done on bigxeon64.

5.2.1 STBIO Recording

Run start_dr will recorde stbio data to /media/data1/dr/. It is vitally important that the Time Generator is running, otherwise dates and file size will be incorrect.

5.2.2 STBIO Playback

In general on the Data Service and Signal Processors, and possibly the data display apps need to be running when executing the STBIO player. The playback rate is specified in dp.cfg. Present playback rate is at 25%.

5.2.3 Generating DAT32

Dat is generated any time data is being written to the data servers. Thus DAT can be generated on playback or in live mode.

Run start_dat to start dat recording for MANTA, DASM, and Sonobuoy. Alternativly, you can start dat recording seperatly in start_dat.mant, start_data.dasm, and start_dat.sb.

5.2.4 Generating DL Text NAD

Run start_dl.

5.3 Shutdown Procedure

Run stop_stb.q302 to stop processes on all of the procs listed in stb.cfg.q302.

Run stop_stb to stop processes on the local machine only.

DOCUMENT CONTROL DATA

(Security markings for the title, abstract and indexing annotation must be entered when the document is Classified or Designated)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Chris Wheeler
General Dynamics Canada Ltd
3785 Richmond Road
Ottawa, Ontario
K2H 5B7

2a. SECURITY MARKING (Overall security marking of the document including special supplemental markings if applicable.)

UNCLASSIFIED

2b. CONTROLLED GOODS

(NON-CONTROLLED GOODS)
DMC A
REVIEW: GCEC APRIL 2011

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

APPENDIX A: Q302 Trial Software Description Report : Sonar test Bed Support for DRDC Atlantic Sea Trail Q302

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Wheeler, C.

5. DATE OF PUBLICATION (Month and year of publication of document.)

August 2007

6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)

136

6b. NO. OF REFS (Total cited in document.)

0

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

Defence Research and Development Canada – Atlantic
9 Grove Street
P.O. Box 1012
Dartmouth, Nova Scotia B2Y 3Z7

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

W7707-063585/A

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)

10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

DRDC Atlantic CR 2007-064

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

Unlimited

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

Unlimited

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Software Functionality; Interfaces; Signal Processing; Scripts; Configuration Files

Defence R&D Canada

Canada's Leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca