

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 22-12-2011		2. REPORT TYPE Technical		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE VICTORY Architecture Position Service Validation Testing				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Venu Siddapureddy				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES) Vehicle Electronics And Architecture (VEA), TARDEC/RDECOM 6501 E. 11 Mile Rd. Warren MI 48397 - 5000				8. PERFORMING ORGANIZATION REPORT NUMBER 22509	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S) RDECOM-TARDEC	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT This document describes a set of validation experiments and will provide verification and validation of the VICTORY 1.0 Architecture Position (GPS) service specifications. The Position service is implemented according to the standard format specified by the VICTORY 1.0. This service provides position data anywhere on the VDB (VICTORY Data Bus). The main components and interfaces evaluated on the VDB to verify the implementation included the - o Position data VDM (VICTORY Data Message) - o VDB Position Data Interface - o VDB Position Management Interface					
15. SUBJECT TERMS VICTORY, Position Service, Validation, Verification					
16. SECURITY CLASSIFICATION OF: unclassified			17. LIMITATION OF ABSTRACT SAR	18. NUMBER OF PAGES 15	19a. NAME OF RESPONSIBLE PERSON Venu Siddapureddy
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			19b. TELEPHONE NUMBER (include area code) 586-282-9865

Registration No.

22509 RC

VICTORY Architecture Position Service Validation Testing

Prepared By: **Venu Siddapureddy**
Engineer, Vehicle Electronics And Architecture (VEA)
TARDEC/RDECOM
Warren, MI 58397

DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.

Disclaimer: Reference herein to any specific commercial company, product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the Department of the Army (DoA). The opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or the DoA, and shall not be used for advertising or product endorsement purposes.

December 2011

U.S. Army Tank Automotive Research,
Development, and Engineering Center
Detroit Arsenal
Warren, Michigan 48397-5000

Table of Contents

1.0	Introduction.....	4
2.0	Experiment Goals	5
3.0	Experiment Design.....	5
3.1	Wireshark VDB Plug-in	5
3.2	Terminal Client and GUI Clients for Position VDM Management.....	6
4.0	Specifications Evaluated:.....	7
5.0	List of Experiments	8
6.0	Procedure 1: Position Management Static Parameters	8
7.0	Procedure 2: Position Data Enabled	9
8.0	Procedure 3: Position Data Validity	10
9.0	Procedure 4: Update Period	11
10.0	Procedure 5: Sequence number continuity	12
11.0	Procedure 6: System Resource Usage	12
12.0	Procedure 7: End-to-End Latency	13
13.0	Conclusion:	14
14.0	Open Issues for Future Testing:	15

Figures

1.	VICTORY Services network as implemented in the VIDS	3
2.	Block Diagram for Position Service Testing.....	4
3.	Wireshark VDB Plug-in Screen Shot.....	5
4.	Terminal Client for Position VDM Management	5
5.	GUI Client for Position VDM Management	6
6.	VDM build time for Position Service	12
7.	VDM Transmit time for Position Service	13

Tables

1.	Position service Static Parameters results	9
2.	Position Service Update Period results	12

This document describes a set of validation experiments and will provide verification and validation of the VICTORY 1.0 Architecture Position (GPS) service specifications. The Position service when implemented correctly and with the standard format specified by the VICTORY 1.0 will provide position data to the VDB (Victory Data Bus) as shown in Figure 1. The main components and interfaces being evaluated on the VDB (VICTORY Data Bus) include

-
- The diagram illustrates a complex network architecture for a threat detection and weapons interface system. The components and their connections are as follows:
- Position Service (Circled in Blue):**
 - Position Server:** Connected to a **GPS Receiver** via **RS 232**.
 - Threat Message Server:** Connected to the Position Server.
 - Threat Message Service:**
 - Threat Detection System:** Connected to the Threat Message Server.
 - Threat Simulator:** Connected to the Threat Detection System.
 - Orientation & Direction of Travel Message Service:**
 - Orientation Server:** Connected to the Threat Message Server.
 - Direction of Travel Server:** Connected to the Orientation Server.
 - Remote Weapons Station Interface:**
 - RWS Interface:** Connected to the Direction of Travel Server.
 - RWS System:** Connected to the RWS Interface.
 - Network Infrastructure:**
 - Precision Time Protocol H/W:** Connected to the Network Time Protocol Server.
 - Network Time Protocol Server:** Connected to the Network File Servers.
 - Network File Server 1** and **Network File Server 2:** Connected to the Network Time Protocol Server.

The experiments will evaluate these interface specifications by integrating software clients and services developed using the specifications, and evaluating the resulting functional behavior and performance. The TARDEC Vehicle Electronics and Architecture (VEA) group executed this set of additional validation experiments, utilizing their VICTORY Interoperability & Development System Integration Laboratory (VIDS)

2.0 Experiment Goals

- Evaluate the completeness and unambiguousness of the individual interface specifications of each included service.
- Investigate the resulting functional performance of reference implementations of the interfaces integrated with representative hardware.

3.0 Experiment Design

The experiments in this set all leverage a common hardware and software design. The Physical/Logical block diagram for testing the Position service is shown in Figure 2. The two test tools used for managing and monitoring the position VDM's are,

- Wireshark VDB plug-in
- Position Service Management Interface

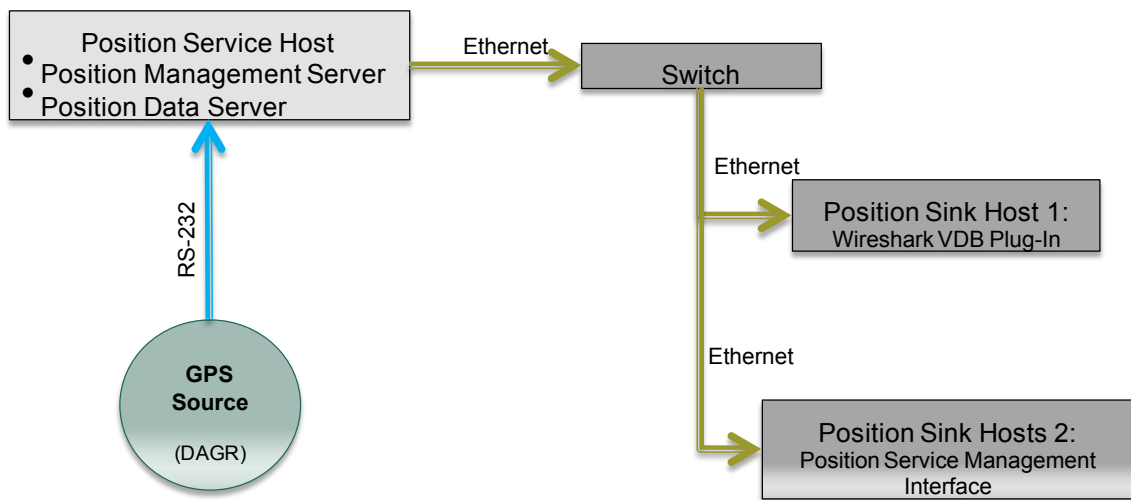


Figure 2: Block Diagram for Position Service Testing.

3.1 Wireshark VDB Plug-in

A custom dissector plug-in for Wireshark version 1.2.8 is developed for the VIDS lab and is used as a tool for testing and monitoring VDM's. This dissector captures UDP VICTORY Data Messages (VDMs) and breaks them down into their specific header and data fields as show in Figure 3. It also provides a filter to look for VDM messages and the ability to log captured VDMs to a formatted text file. Each properly formatted VDM packet contains a "Header" field and a "Data" field, which can be expanded upon as described in the following sections.

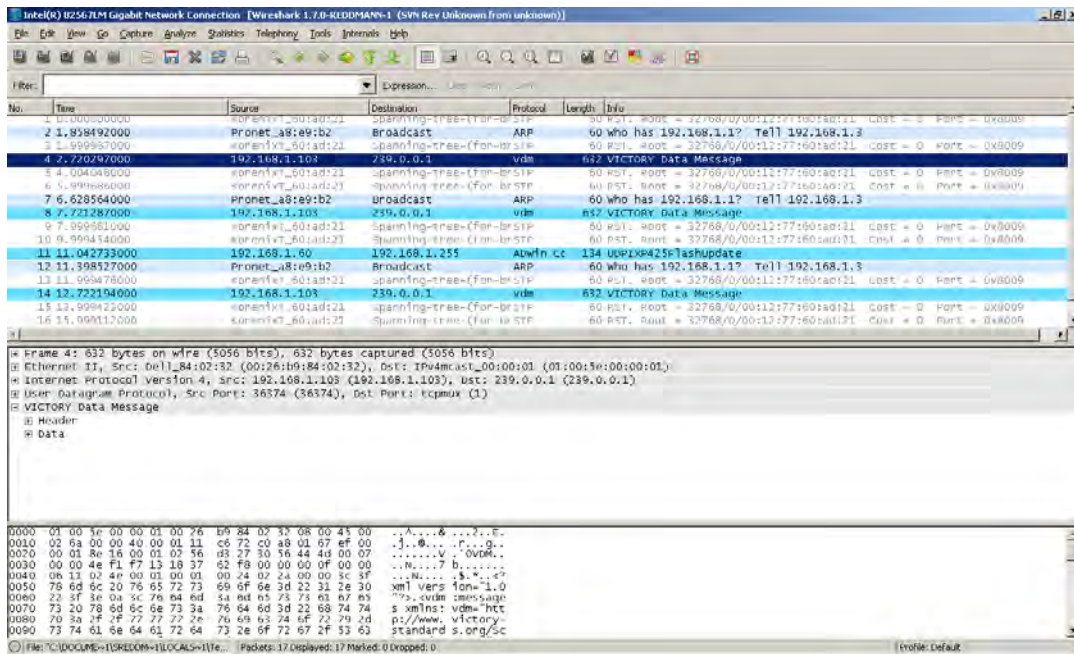


Figure 3: Wireshark VDB Plug-in Screen Shot

3.2 Terminal Client and GUI Clients for Position VDM Management

In addition to the Wireshark plug-in to view the data and header, two clients are developed to manage the VDM's. One is a command line client and the other is a GUI based client as shown in Figure 4 & Figure 5. Both of these clients perform the same VDM control functionality, i.e. to enable/disable, set data rates, set update period, etc.

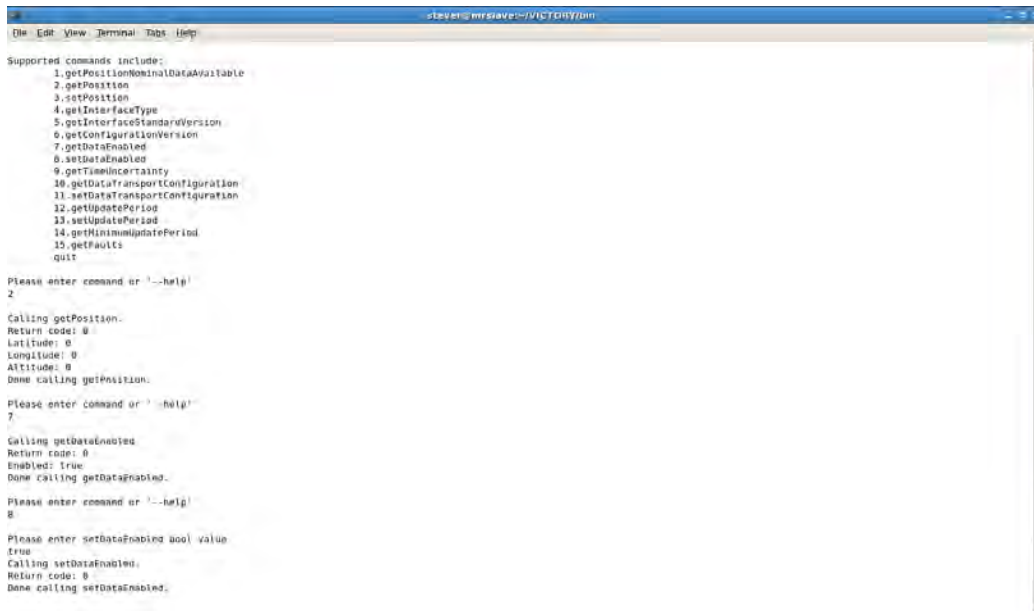


Figure 4: Terminal Client for Position VDM Management

VICTORY Manager

Position Orientation Threat Direction Of Travel Time

Position

Data

ID	Timestamp	Latitude/Uncertainty	Latitude Valid	Latitude Estimated	Longitude

Position

Latitude:

Longitude:

Altitude:

MGRS:

Nominal Data Available

Latitude: ☐

Longitude: ☐

Altitude: ☐

Geodetic Datum:

Logfile name:

Figure 5: GUI Client for Position VDM Management

4.0 Specifications Evaluated:

The VICTORY Architecture 1.0 standards are specified in the Version 1.0 document released on July 29, 2011. The following are the standards that were evaluated

- VICTORY Data Management Interface <10008-20110729, Pro>
- SOAP Compliance and Standards <10001-20110729, Pro>
- Application Layer Data Encoding <20001-20110729, Pro>
- Application Layer Message Encapsulation <20002-20110729, Pro>
- Timestamp Format for Application Layer Data <20004-20110729, Pro >
- Position Data Interface <20009-20110729, Pro>
- Position Management <15003-20110729, Pro>
- Position Interface Complex Types <20014-20110729, Pro>

5.0 List of Experiments

This experiment set is composed of the following procedures:

- Position Management Static Parameters
- Position Data Enabled
- Position Data Validity
- Update Period
- Sequence number continuity
- System Resource Usage
- End-to-End Latency

6.0 Procedure 1: Position Management Static Parameters

The position management interface must maintain a set of static parameters and supply them to the client's request. This procedure will evaluate whether the position management service can reply to the client's Get() requests on all static parameters. The list of static parameters includes: position data management specific parameters (Geodetic datum and nominal data available), version information management specific parameters (interface type, interface standard version and configuration version) and VICTORY data management specific parameters (time uncertainty and minimum update period).

A terminal position management client described in the previous section 3.0 is used to execute Get() commands on all of the static parameters mentioned above. The results of those Get() commands are recorded and compared to the specification to determine their validity.

Procedure Details

- a. Execute a get() command on every static parameter
- b. Write down the received values
- c. Although the specification doesn't declare the format and range of the static parameters, all static parameters are implicitly assumed to have the same values as the ones in the 'Static Configuration Settings' file. Therefore the received values from the get() request should be compared with the 'Static Configuration Settings' file.

Results

- The received values from the get() request compared with the 'Static Configuration Settings' file are shown in Table 1 below, the values from the get() compare identically with the settings from the 'Static Configuration Settings' file.

Static parameters	Values received by client	Values stored in the 'Static Configuration Settings' file
Interface ID	N/A	N/A
Source ID list	N/A	N/A
Interface type	Position	Position
Interface standard version	1.0	1.0
Configuration Version	TARDEC_VEA_VIDS_1.0	TARDEC_VEA_VIDS_1.0
Geodetic datum	N/A	N/A
Timestamp uncertainty	0	0
Minimum update period	0	0
Nominal data available	0, 0, 0	0, 0, 0

Table 1: Position service Static Parameters results

7.0 Procedure 2: Position Data Enabled

Data enabled is a parameter that identifies whether the position data interface is sending out data. This procedure validates whether the Position Management Interface can reply to Get() and Set() requests on data enabled and whether the flow of position data is properly controlled by the data enabled parameter.

This procedure starts by using a position management client to send a data enabled Set() command with value of True to the position management interface. After processing the Set() command, the position management interface is supposed to enable the flow of position VDMs, which will be verified using the Wireshark VDM Plug-in described in Section 3.0 with appropriate network filters. Then the position management client will send a Get() command to the position management interface for data enabled parameter, with an expected value of True as it was set previously. Next the position management client will send another Set() command on data enabled with the value of False this time. Wireshark will again verify that

there is no longer any position VDMs sent out. The position management client will send a data enabled Get() command to ensure that the received value is False. After that the position management client will set the value of data enabled to True again, and Wireshark will verify that position VDMs is sent out also. Finally, the position management client will send out a Get() request to ensure that the value of data enabled is True.

Procedure Details

- a. Use the client to set the update period to 1 second
- b. Execute a set() command on data enabled property to True
- c. Verify on Wireshark that position data is being sent out periodically for at least 10 update periods
- d. Execute a get() command on data enabled property
- e. Verify that the received value is 1
- f. Execute a set() command on data enabled property to False
- g. Verify on Wireshark that position data stops sending out for a time out of at least 10 update periods
- h. Execute a get() command on data enabled property
- i. Verify that the received value is 0
- j. Set the data enabled property back to True
- k. Verify on Wireshark that the position data is sent out periodically for at least 10 update periods
- l. Use the client to request data enabled property and verify that it is 1

Results

- Both Get() and Set() commands work on the data enabled setting.
- When the data enabled property is 1 (True), position data is sent out periodically.
- When the data enabled property is 0 (False), position data stops sending out.
- Overall, the outcome matches the expected response.

8.0 Procedure 3: Position Data Validity

Before sending out position VDMs, the position data interface must populate all necessary fields of the binary header, formulate the XML payload according to the position data schema and preserve the integrity of the raw data sampled from any GPS source. This procedure will determine the integrity of position data values and the validity of binary header and XML payload. It will also determine if position data can be extracted from the position VDMs in an unambiguous way.

In this procedure simulated position data will be generated and fed to the position data interface. And the output position VDMs will be analyzed and validated.

Procedure Details

- a. Use DAGR data input to feed the current location
- b. Run Position Server to read DAGR data and generate Position VDM
- c. Start a client connection to the Position Service Management

- d. Execute a set() command on data enabled property to True
- e. Run the Position Data Sink for an hour to interpret VDM multicast messages and validate the data
- f. The Position Data Sink will write the validation result into a log file

Results

- All messages have valid binary header.
- For the implementation evaluated:
 - The position values received and extracted from the VDB matched the values that the service encapsulated and sent in the VDM.
 - The position values sent out by the data service is identical to the DAGR values.

9.0 Procedure 4: Update Period

The position management service must allow dynamic Get() and Set() functions on update period parameter. And the position data service should arrange transmission delay between each position VDM in consistence with the update period parameter.

The position management client will set the update period to different values. At each value, position VDMs will be collected for an hour and analyzed to determine if the update period value is consistent with the transmission period between each position VDM. Consistency is a subjective criterion in this case.

Procedure Details

- a. Use the client to query the Position Management Service for minimum update period
- b. Do a set() command to change the update period to its minimum
- c. Get() the value and verify it was set correctly
- d. Use the Position Data Sink to listen to position multicast stream and collect position data for an hour. The Position Data Sink will record the timestamps and differences between successive timestamps into a file.
- e. At the same time, use Wireshark with the appropriate capture filter to capture position data messages
- f. Calculate the average of the list of period between successive timestamps from the record file
- g. Use Wireshark to generate average packets/sec statistic. Its inverse will be the average update period.
- h. Record the value from Wireshark

Results

Wireshark results were captured for a period of 1 hour. The average update period for the VDM message set for a 1 sec update period is calculated and show below,

Update Period Setting	Average value update for VDM message from Wireshark
1 sec	1.000837923 sec

Table 2: Position Service Update Period results

10.0 Procedure 5: Sequence number continuity

The position data interface must tag each message with a monotonically increasing sequence number before sending it out. To validate the sequence number of position VDMs, the number of discontinuities will be recorded. Position VDMs will be continuously sent out for an hour, and a position data sink will analyze the continuity of sequence numbers of all position VDMs.

Procedure Details

- Use the client to query for the minimum update period
- Set the update period to twice the minimum period
- Perform a get() command to verify that the update period was set correctly
- Use the Position Data Sink to check if the sequence number increases monotonically
- Allow Position Data Sink run for one hour and record sequence number discontinuities

Result

Received sequence numbers increases monotonically without any discontinuities.

11.0 Procedure 6: System Resource Usage

This procedure will investigate the computing resources required to interpret position messages. The recommended metrics for system resource usage is XML processing time.

Procedure Details

- Start the Position Data Service at a reasonable update rate
- Use the Position Data Sink to collect values of processing time and memory used during the interpretation and validation of VDM messages
- Display the results in the log file graphically

Result

The processing time for creating position VDM ranges from 52,000 nsec. to 91,000 nsec. with some sporadic peaks up to 114,000 nsec as shown in Figure 6. The average XML build time is 83,000 nsec and Std Dev of 43,840 nsec.

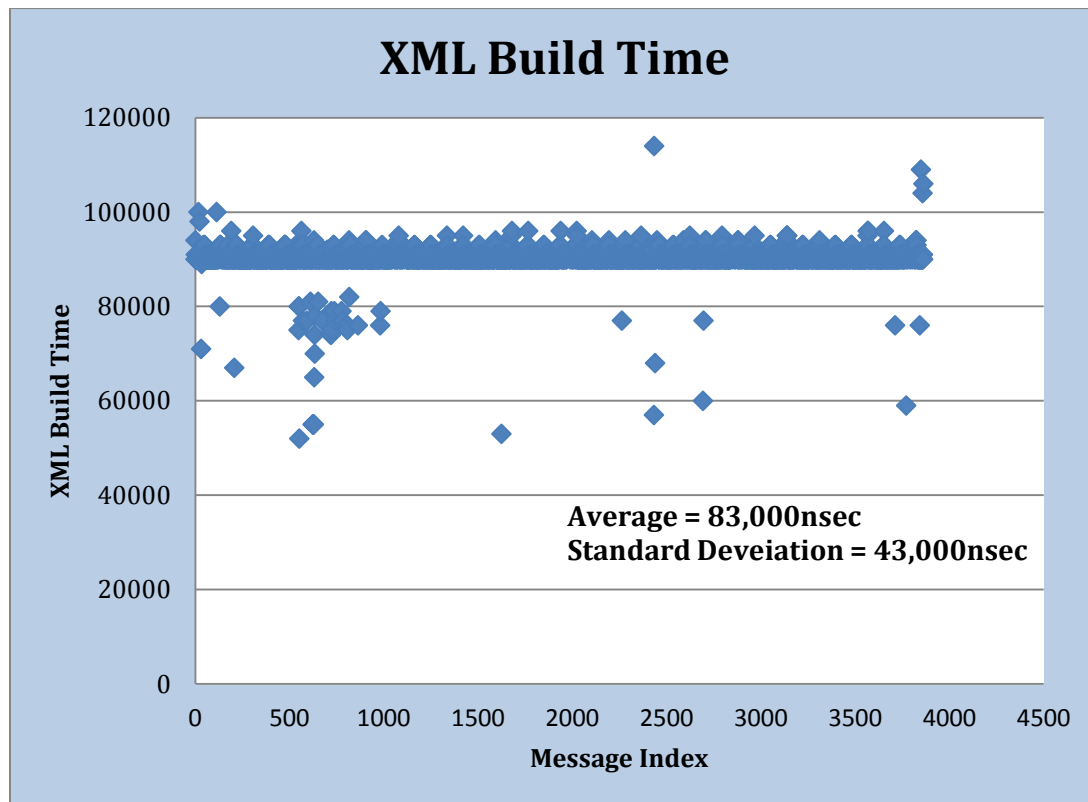


Figure 6: VDM build time for Position Service

12.0 Procedure 7: End-to-End Latency

The accuracy of position data heavily depends on the latency from the GPS sensor output to the received corresponding update from the position data service. This procedure will evaluate the end-to-end latency by measuring the time it takes to create and transfer a position VDM with raw data received from a position sensor and sunk into by a position data client. The raw position device is a RMC (recommended minimum data for GPS) string output from a serial port of the DAGR GPS device.

Procedure Details

- Configure the DAGR to broadcast the GPS data in RMC sentence. The Position Server mentioned in Figure 1.0 receives the RMC sentences from the DAGR over the RS-232 connection.
- Start the Position Data Service at the minimum update period (0ms for this reference implementation), the data service will parse and create the Position VDM's and will start broadcasting them.
- Start Wireshark Position Data Sink. It will continually receive position VDM until it finds the same GPS values inside the XML payload. While doing that, it

keeps track of 2 timestamps: one when the Position Data Service just finishes parsing and creating the VDM's and another when the updated VDM is received. The end to end latency of position data is calculated by subtracting those 2 timestamps and recorded into a log file.

- d. Stop the Position Data Sink after about 60 minutes.
- e. Open the latency log file and display the results graphically.

Result:

The results displayed below show the complete position VDM creation, broadcast and consumption time, i.e. end-to-end latency as shown in Figure 7.

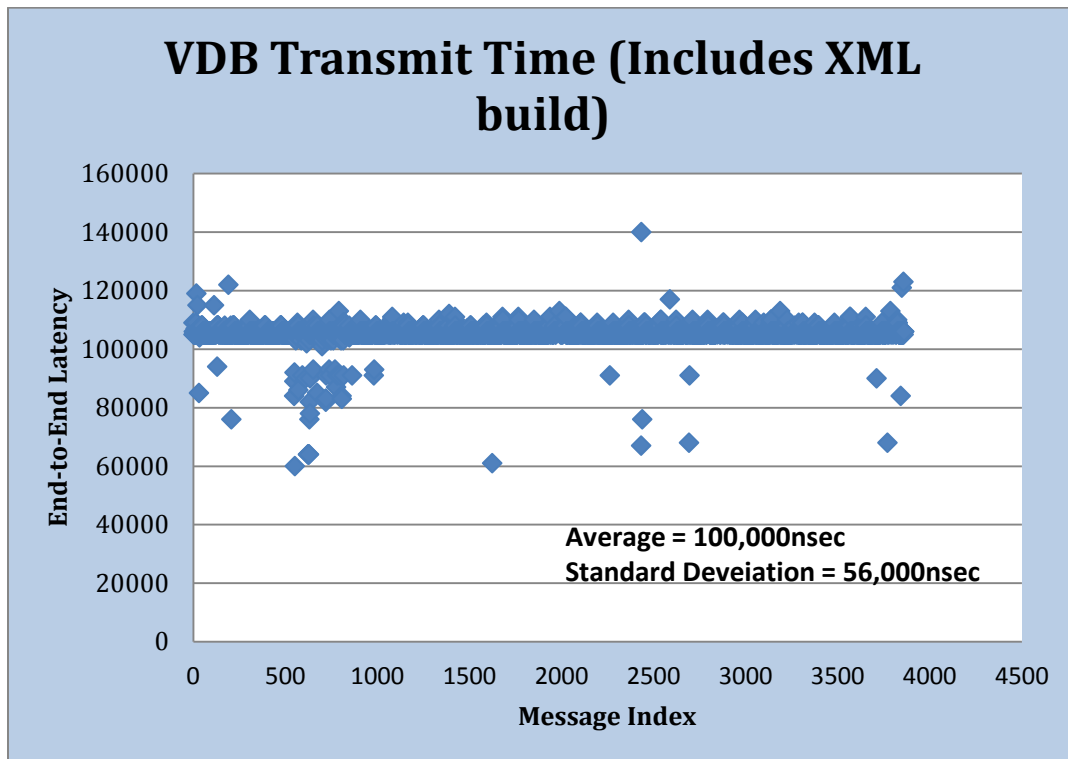


Figure 7: VDM end-to-end time for Position Service

The End-To-End time for creating position, transmitted and sinking the VDM ranges from 60,000 nsec to 105,000 nsec with some sporadic peaks up to 140,000 nsec.

13.0 Conclusion:

The experiments performed in the VIDS evaluated the VICTORY Architecture Standards 1.0 interface specifications by integrating software clients and services developed to the specifications, and evaluated the resulting functional behavior and performance. In conclusion, the Position service as specified for the 1.0 standard as implemented in the VIDS is

- Complete and unambiguous of the individual interface specifications of each included service.

- The resulting functional performance of this reference implementation with the representative hardware is adequate to the current development needs of the VICTORY architecture.

The next step in validating the service is determining whether interoperability is achieved when multiple implementers develop to the specifications. These tests will be conducted in the VIDS and the results will be published.

14.0 Open Issues for Future Testing:

The testing completed at this stage included only the verification of the standard for the standalone Position Service. Future testing will include interoperability testing, system level testing and end-user testing. To include the services for integrating to end user applications the following tests will be done and the ensuing development will be recommended.

- Notification to the user and the application when the GPS signal is lost. Recommend mitigation actions to the standard specifications or at the application end.
- Test and report the limit to data broadcast frequency i.e. the update period and limit to the number of connections (open sockets) to the data server when multiple services (Position, DOT, Orientation, Threat, RWS etc.) are implemented on the same server.