DTE-ARCH TENA Log Data Analysis

Written By:

Ben Matthews
GaN Corporation
Computer Engineer

Becky Mortlock
GaN Corporation
Engineer

ABSTRACT: *The DTE-Arch (Distributed Test and Evaluation - Architecture) Milestone 4 was an event undertaken by the United States (US) Developmental Test Command (DTC) to analyze the future of the distributed Modeling and Simulation (M&S) environment. This event took place on the 25th and 26th of January, 2005 with participants including Aberdeen Test Center (ATC), Aviation Technical Test Center (ATTC), Dugway Proving Grounds (DPG), Electronic Proving Grounds (EPG) at Fort Lewis and Fort Huachuca, Redstone Technical Test Center (RTTC), One-SAF Testbed Baseline (OTB), White Sands Missile Range (WSMR), and Yuma Proving Grounds (YPG). This event utilized the Test and Training Enabling Architecture (TENA) Middleware Release 4.0.4 to simulate the various components within the operational scenario. Preliminary data was collected and analyzed to indicate possible communication problems encountered during the execution of the event. The data came from a single Object Model (OM) called the Application Management Object (AMO), which every participant was requested to emulate. This OM Stateful Distributed Object (SDO) contained the status characteristics of the component emulation applications and provided this data to the supporting application called Starship. The emulating application for this OM logged all published and subscribed SDOs during the Milestone 4 event. This paper focuses on the areas of the log files where data is 'missing' (times when the application was known to be subscribing to the AMO SDO updates but no data was logged).*

1.0 Introduction

The DTE-Arch Milestone 4 event was executed over the Defense Research and Engineering Network (DREN). It included the following CONUS (Continental United States) test centers: ATC, ATTC, DPG, EPG Fort Lewis (EPG-FL), RTTC, WSMR, and YPG. The following figure shows the DREN network and all of the participating locations during the DTE-Arch Milestone 4 event.

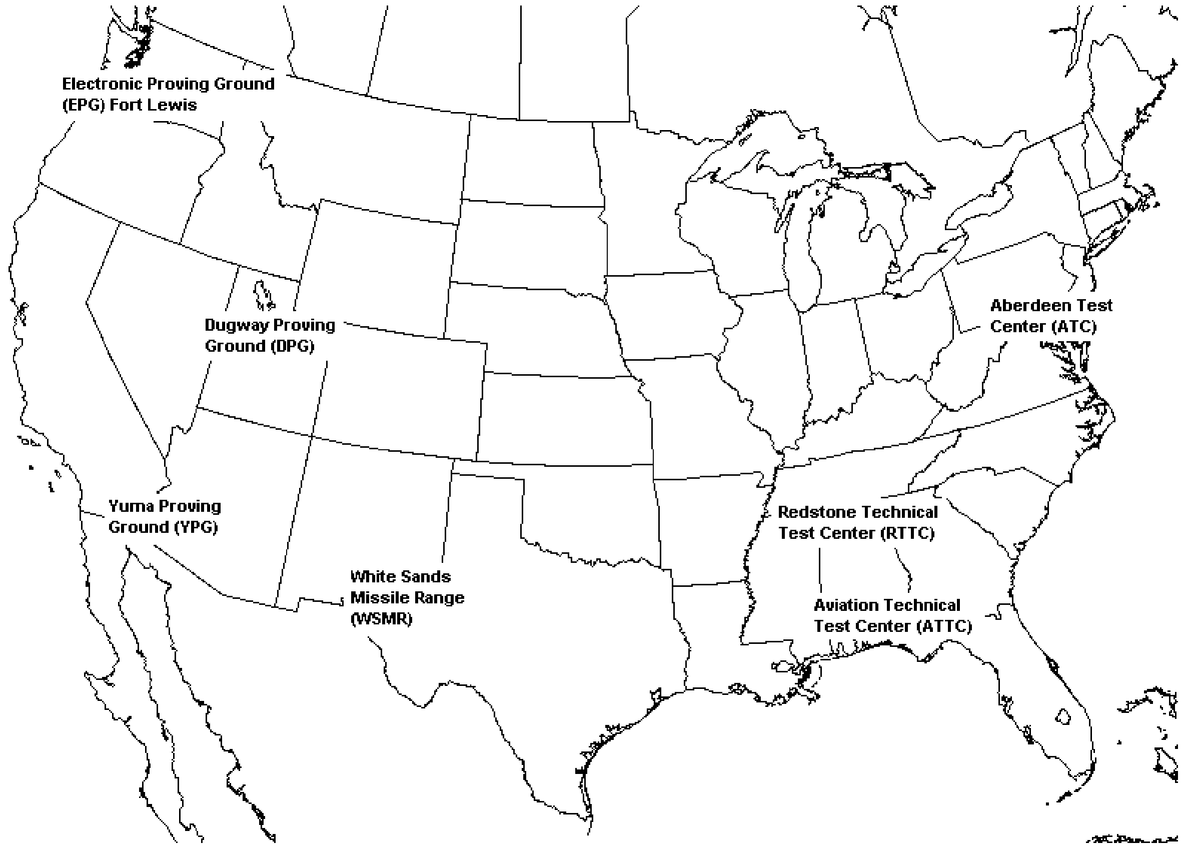

Figure 1: DTC sites participating in DTE-Arch Milestone 4.
(Modification of picture obtained from http://www.spc.noaa.gov/products/ wwa/warnsummary.html)

The DTE-Arch Milestone 4 event included a preliminary data collection from the published SDOs of one OM, Application Management Object (AMO), to provide insight into the communication between test centers during the testing of the distributed environment. This paper addresses portions of the log files where data is missing. Data was considered missing when a test center was known to be subscribing to AMO SDO updates from other sites but one or more updates were missing from the test center's log files.

During this event, all test centers were running TENA Release 4.0.4, which utilizes Lazy

Discovery. Lazy Discovery requires a subscriber to receive an update from the publisher before it realizes or “discovers” the publisher is participating. For this reason the AMO SDO updates were published every 10 seconds during the event. (Note: TENA release 5.0.1 utilizes Eager Discovery which gives the subscribers the ability to seek out publisher data when they begin subscription). Each SDO update of the AMO contains a sequential number (AMO Version) used to distinguish between consecutive updates. Every time the application starts, AMO Version 1 is published and then incremented by one for each additional publication. When the application is shut down and restarted, the AMO Version begins again at one.

An emulating application (AMO Emulator or a custom application) created using TENA Middleware subscribed and published the SDOs. The AMO Emulator was an application distributed for this event to emulate the AMO object. Three test centers (ATTC, OTB, and EPG-FL) gathered data using their own customized applications to emulate the AMO OM. These implementations were different from the AMO Emulator in the manner in which they operated and logged data. (These customized loggers are referred to as 'Other' in Tables 4 and 5.)

The AMO Emulator provided a visual output to the screen showing obtained SDO updates from other test centers AMO emulating applications. This was used during the event to show when the test centers were communicating. The AMO Emulator also created files containing all of the published and received updates. In total, four log files were produced by the AMO Emulator for each test center during the logging process: two files for the subscriber and two for the publisher. One of the two subscriber log files was created and managed by the TENA Middleware Logging function. This file contained the attributes pertinent to the AMO OM, including application status information present in the SDOs from the other test centers. It also contained a time stamp (to the seconds) indicating when the subscriber gathered the published data. The second subscriber file created by the AMO Emulator was a detailed time stamp logger that contained the application ID, SDO ID, AMO Version and a microsecond time stamp. The detailed time stamp logger was a customized logging layer implemented in combination with the TENA Middleware Logger. For more information on customized logging, see *DTE-Arch Milestone 4: Customized Logging for TENA Middleware* available at http://www.gancorp.com. The time stamp implemented in the customized logger provided higher precision, and more detail for data analysis (TENA log files were occasionally referenced for additional information). The publisher log files were created in a similar fashion but contain only publication data.

Time stamps recorded in the log files were in ZULU time (GMT-Greenwich Mean Time) to provide consistency across time zones. When a site published an AMO SDO, the time stamp was created with that particular site's system time (in GMT) and placed in the publisher log files. When the other sites received the SDO, a time stamp with the receiving system's time (in GMT) was logged in the subscriber log files. For sites utilizing the AMO Emulator, time stamps indicating the beginning and ending of the application run were also logged in the publication and subscription log files.

Section 1.1 details the data analyzed for this paper. Section 2 describes missing data and provides examples, while Section 3 seeks to explain the missing data. Conclusions are presented in Section 4.

1.1 Data Analyzed

Approximately 30 log files containing 6635 unique SDO updates and 17,551 subscriptions to these updates were examined. All of the log files generated by the AMO Emulator were in XML (eXtensible Markup Language) format. These files were processed into a Microsoft Access format which was then exported to Microsoft Excel files. The remainder of the data analysis was performed in Open Office Spreadsheet. Table 1 shows the available data from each test center.

Test Center	*Subscription Data*	*Publication Data*
ATC	available	available
ATTC	unavailable, not provided	available
DPG	unavailable, different format	unavailable, different format
EPG-FL (Fort Lewis)	available	unavailable, not provided
OTB (AMOEm)*	available	unavailable, not running
OTB	available	unavailable, not provided
RTTC-Radar	available	available
RTTC-ILH(Integration Level Hierarchy)**	unavailable, not provided	available
Starship (EPG Fort Huachuca)	available	available
WSMR	unavailable, not provided	available
YPG	available	available

Table 1: Available data by test center.

All efforts were made to include as much data as possible, however some data files were unavailable for the following reasons: (1) the log files were not provided, (2) the log files were not created because the publisher and/or subscriber was not running, or (3) the log files were in a format that was not easily processed and the time needed to convert this data to a usable format was prohibitive. Subscription and publication data listed as 'available' in Table 1 was used for this analysis.

*OTB (AMOEm) was the AMO Emulator run along with the OTB emulating application. Both of these were subscribing to the AMO object model.

**RTTC(ILH) subscription data is shown as unavailable because the subscription data from both RTTC(ILH) and RTTC-Radar was written to a single log file even though they were separate applications. This caused overwriting problems that resulted in a loss of data.

2.0 Missing Data

During analysis of the data log files from the DTE-Arch event, it became evident that updates were missing from the log files at times when applications were known to be subscribing. The following sections show examples of areas in the log files where data is missing. In Section 3, explanations are provided to account for these occurrences whenever possible.

2.1 Example of Missing Data

This small sample from the log files is provided to show a representative picture of missing data found throughout. Table 2 shows data from ATC's publication log file and the corresponding time stamps from subscriber log files. An update is considered missed if the subscriber is on ('y" in 'ON' column), and no time stamp is logged. It can be seen that updates are missed by ATC, RTTC, Starship and YPG. OTB AMOEm. and OTB were not subscribing and it was unknown if EPG-FL was running (custom logger did not place begin/end time stamps in the log files). Missed initial data and evidence of communication problems are seen in this example and addressed in Sections 3.

Publisher			***Subscribers***													
App ID	AMO Ver.	Pub. Time	ON	OTB AMOEm	ON	ATC	ON	OTB	ON	RTTC	ON	Star ship	ON	YPG	ON	EPG FL
100	1	1106692965.751160	n		y		n		y		y		y		?	
100	2	1106692966.753240	n		y	1106692976.727000	n		y		y		y		?	
100	3	1106693002.232910	n		y	1106693002.232910	n		y	1106693101.312500	y		y		?	
100	4	1106693012.269300	n		y	1106693012.269300	n		y	1106693111.390630	y		y		?	
100	5	1106693022.305690	n		y	1106693022.321350	n		y	1106693121.453130	y		y		?	

Table 2: Sample Data

2.2 Scenario Data

It is instructive to analyze the data during a time when communication was well-established. The time period selected was the final scenario execution occurring January 26 (21:27 to 22:15 GMT). Figure 2 shows the applications' run times and the order in which they joined the execution.

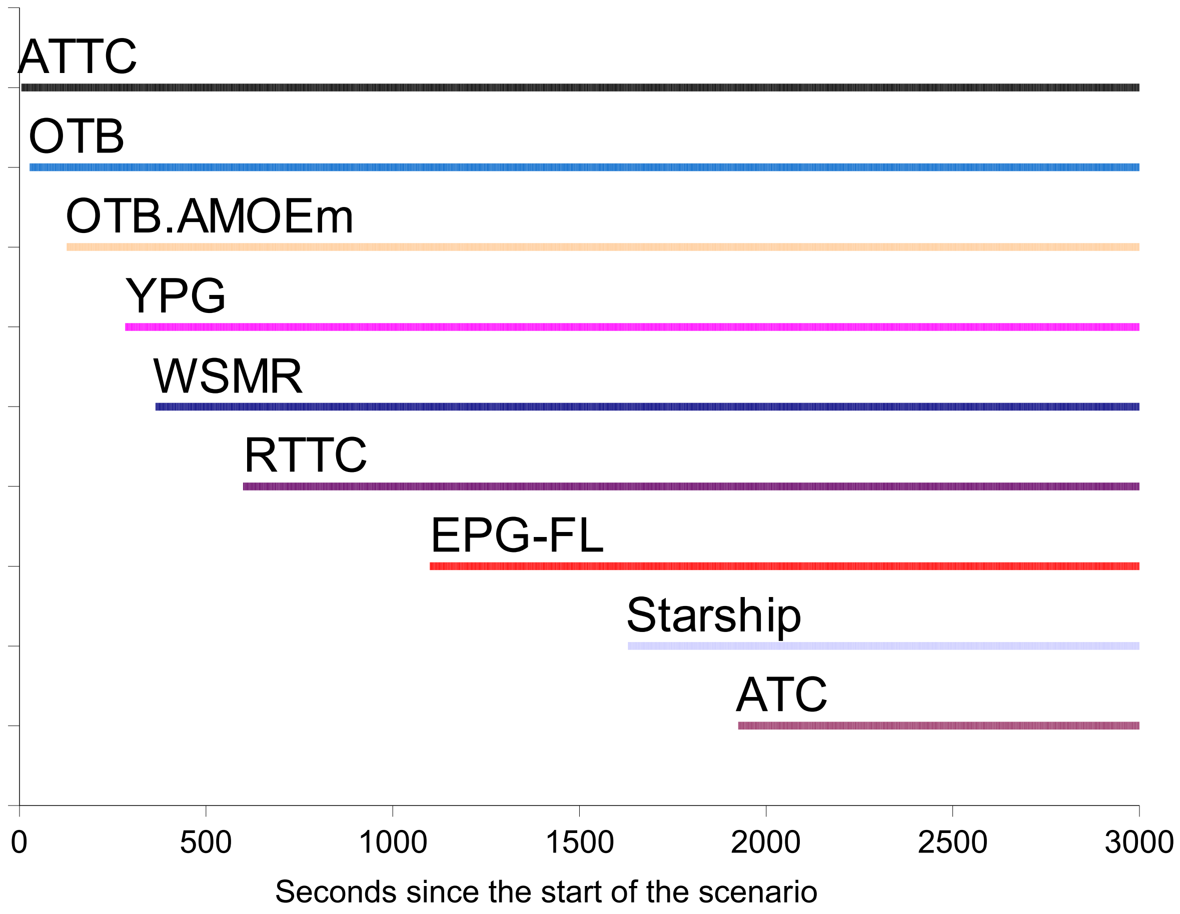

Figure 2: Test Center Application Run Times during Scenario Event
on 26 Jan 2005 (21:27 to 22:15 GMT)

Even though sites were communicating, updates were missed. Table 3 shows the number of updates missed during the scenario. The top third of Table 3 shows the estimated number of updates based on application begin/end time stamps. The next third displays the actual number of updates obtained during this time and the bottom third is the difference between the two (or the number of updates considered missed).

Subscribers		***Publishers***						
		ATC	***ATTC***	***RTTC***	***Starship***	***WSMR****	***YPG***	***Totals***
SDO Updates Published		95	258	229	123	251	254	1210
Estimated # of Updates By:	ATC	95	85	95	94	95	95	559
	EPG-FL**	?	?	?	?	?	?	?
	OTB	95	258	229	123	251	254	1210
	OTB.AMOEm	95	247	229	123	251	254	1199
	RTTC	95	204	229	123	227	227	1105
	Starship	95	112	126	123	125	124	705
	YPG	95	235	229	123	251	254	1187
Actual # of Updates By:	ATC	94	77	86	86	86	86	515
	EPG-FL	94	160	178	122	178	178	910
	OTB	0	256	227	121	251	253	1108
	OTB.AMOEm	93	247	227	121	251	253	1192
	RTTC	0	204	228	121	227	226	1006
	Starship	93	109	124	122	122	122	692
	YPG	0	233	226	121	251	253	1084
Missed # of Updates By:	ATC	1	8	9	8	9	9	44
	EPG-FL	?	?	?	?	?	?	?
	OTB	95	2	2	2	0	1	102
	OTB.AMOEm	2	0	2	2	0	1	7
	RTTC	95	0	1	2	0	1	99
	Starship	2	3	2	1	3	2	13
	YPG	95	2	3	2	0	1	103

Table 3: Estimated/Actual/Missed SDO updates by test center for scenario run.

*WSMR's publication data was partially overwritten. This overwriting problem was caused by the ungraceful exiting of an application. WSMR published AMO Versions 1-30 are not available for analysis.

**'?' indicates subscriber run time is unknown for EPG-FL (no begin/end time stamps were logged by the custom logger).

The lower portion of the table, 'Missed # of Updates', shows the number of missed SDO updates during the final scenario run. These missed updates were all initial missed updates (discussed in Section 3.1) and usually ranged between 0 and 3. Data for ATC, however, falls outside these values and the following observations were made:

1) ATC misses 8 or 9 updates from all publishers except itself. Further analysis of the DTE-Arch Milestone 4 data (not presented here) suggests ATC's clock was not synchronized with the other systems, and appeared to be behind by approximately 65 seconds. If this is taken into account, these numbers change to:

	ATC	*ATTC*	*RTTC*	*Starship*	*WSMR*	*YPG*	*Totals*
ATC	1	2	3	2	2	2	12

These values are consistent with those seen for the other test sites.

2) OTB's, RTTC's, and YPG's log files did not contain any updates from ATC during this time period. It was independently recorded in an observation log (maintained by Jonn Kim during the event) that RTTC and YPG were not receiving updates from ATC. It was also noted in the log that OTB received ATC's updates, however this information was obtained from the OTB.AMOEm application and not OTB. These site communication issues are discussed in Section 3.2.

3.0 Explanation of Missing Data

3.1 Missed Initial Data

It was seen that all subscribers missed the initial (and sometimes more) AMO SDO update(s) from a publisher after the publisher joined the execution; these missed updates are referred to as missed initial data. Table 4 shows the number of AMO versions missed by each test center (subscriber) when a particular test center began publishing. This encompasses every instance of each test center's application beginning publication over the course of all the data collected (applications were started three or more times). For example whenever ATC began publishing, RTTC consistently missed the first two updates. YPG missed either one or two updates every time Starship began publishing. (If no value is listed it means the subscriber was always off when that publisher began).

		Subscribers						
		OTB (AMOEm)	ATC (AMOEm)	OTB (Other)	RTTC (AMOEm)	Starship (AMOEm)	YPG (AMOEm)	EPG-FL (Other)*
Publishers	ATC(AMOEm)	2	1	2	2	2	CP**	-------
	ATTC(Other)	2		2		2		-------
	RTTC(Radar) (AMOEm)							-------
	RTTC(ILH) (AMOEm)	2		2	1	2	3	-------
	Starship (AMOEm)	2		2	2	1	1,2	-------
	WSMR (AMOEm)	1	14	1	1	1	21	-------
	YPG(AMOEm)	1		1			1	-------

Table 4: Missed updates when publisher joins execution.

Note: CP stands for Communication Problem (see footnote for details).

Table 4 shows the subscribers consistently missed the first one to three AMO SDO updates published by any publisher (with the exception of ATC and YPG, who missed 14 and 21 consecutive updates respectively from WSMR).

After thorough investigation, the following explanations for missing initial updates are offered:

1.) When a connection between a publisher and subscriber is made in TENA Release 4.0.4 propagation time is needed for a publisher to detect that a subscriber is interested in updates and to establish the subscription process. The SDO update may not be received by the subscriber during this time. For details on the discussion, see TENA Help Desk Case #325 at http://support.fi2010.org/. There was a single instance of EPG-FL (using its own customized logger) obtaining the first update (AMO Version 1) from WSMR. It is possible there was no discernible propagation delay at this point.

2.) In TENA the first AMO SDO update accepted by the subscribers is processed as a discovery callback. The next update, and remaining updates, are processed as state change callbacks until a destruction callback is sent. The AMO Emulator implemented its detailed time stamp in the state change callback. This means the first received AMO version was missed by the detailed time stamp because the discovery callback implementation was executed instead of the state change callback implementation. Future implementations of a customized logger should include a logging function in the discovery callback implementation. For more details reference *DTE-Arch*

*EPG-FL is shown here with "--------" in all of its spaces. There was not enough data in the EPG-FL log files to know when the subscriber was on, but not receiving updates.

** CP (Communication Problem) is an issue that affected ATC and YPG. It is discussed in more detail in Section 3.2.

Milestone 4: Customized Logging for TENA Middleware available at http://www.gancorp.com.

3.) Sites using the AMO Emulator published two initial AMO SDO updates within approximately one second of each other at the very beginning of every run. This can be seen in Table 2 with the AMO Versions 1 and 2 being approximately one second apart. The AMO Emulator code publishes the first update (AMO Version 1) when it initializes the data, then modifies the publication state and publishes again (AMO Version 2). After these two updates the application sleeps for 10 seconds, and updates occur every ten seconds thereafter. These initial one second updates, coupled with propagation delay, may have resulted in additional lost updates e.g. YPG missing three updates from RTTC-ILH.

Starship, however, published initial AMO versions seven and nine seconds apart during two separate runs; possibly due to a "freeze" in the application. A study is currently underway on the problems associated with an application freeze which may provide insight.

It was also seen that some subscribers missed the initial (and sometimes more) AMO SDO update(s) from a publisher when the subscriber joined the execution (Table 5). This table encompasses all the data collected; all test centers' applications were started three or more times each. For example whenever ATC began subscribing, it consistently missed eight of ATTC's updates. Starship missed either two or three of ATTC's updates when Starship begins subscribing. The numbers of updates missed by the subscribers were inconsistent and may be associated with any or all of the issues described above.

	Subscribers							
		OTB (AMOEm)	ATC (AMOEm)	OTB (Other)	RTTC (AMOEm)	Starship (AMOEm)	YPG (AMOEm)	EPG-FL (Other)
Publishers	ATC(AMOEm)							-------
	ATTC(Other)	0	8	1	0	2,3	2	-------
	RTTC(Radar) (AMOEm)	13	49			22	3	-------
	RTTC(ILH) (AMOEm)	25	9,10			2		-------
	Starship (AMOEm)		8,27	50	0		1	-------
	WSMR (AMOEm)		9		0,1	0,1,3		-------
	YPG(AMOEm)		9		1	1,2,10	0	-------

Table 5: Missed updates when subscriber joins execution.

3.2 Communication Problems

The following communication problems were identified by examining all the available data files for areas of missing data:

- ATC experienced communication problems with OTB, RTTC, Starship, and YPG during both days of the event.
- ATC started and stopped publishing five different times; RTTC missed all updates from ATC during two of these application runs, and OTB missed all ATC updates during one of these application runs.
- Starship and ATC did not communicate with each other on the 25th of January, however communication was evident on the 26th.
- YPG did not receive any updates from ATC although ATC received updates from YPG. This communication problem was verified both by observations of a YPG operator during the execution and by inspection of the YPG log files (see also Table 2).

3.3 Missing Data Near Subscriber/Publisher Shut Down

Areas of missing data were also noted for two specific time periods; 15:34 and 22:37 GMT on January 26th (Table 4). This occurred near the time a subscriber/publisher shut down and may have been related to the shut down process. It is understood that ungraceful exit during TENA application execution yields unpredictable results, and it is believed that some of the missing data may be attributed to this.

Subscribing Test Center	***Publishing Test Center***	***Time***
OTB	WSMR	~15:34 GMT
OTB (AMOEm)	WSMR	~15:34 GMT
ATC	WSMR	~15:34 GMT
RTTC	WSMR	~15:34 GMT
Starship	WSMR	~15:34 GMT
OTB	ATC	~15:35 GMT
Starship	RTTC	~22:37 GMT
Starship	WSMR	~22:37 GMT
Starship	ATC	~22:38 GMT

Table 6: Missing updates at the end of subscriber run.

4.0 Conclusions

Data collected from log files generated during DTE-Arch Milestone 4 were analyzed to provide preliminary information about communications between participating test centers. The following conclusions can be drawn:

1.) The AMO Emulator missed the first AMO version and can be modified to include a logging function in the discovery callback implementation to remedy this situation. (For more details reference *DTE-Arch Milestone 4: Customized Logging for TENA Middleware* available at http://www.gancorp.com)

2.) TENA Release 4.0.4 use of Lazy Discovery caused initial publication and subscription data to be missed due to propagation time needed to establish communication between publisher and subscriber (see TENA Help Desk Case #325 at http://support.fi2010.org/ for a detailed discussion).

3.) ATC experienced communication problems with OTB, RTTC, Starship, and YPG during both days of the DTE-Arch Milestone 4. The reason for this is unknown and should be further investigated.

4.) Ungraceful exits during TENA applications may cause SDO updates to be missed.

5.0 Acknowledgments

Special thanks to Gary Fee with Technologies Engineering, Inc. for his help with the entire data analysis process. Gary created the AMO Emulator used during the DTE-Arch Milestone 4 event He also imported all of the data from the XML log files into Microsoft Access and provided the Microsoft Excel files used for this data analysis.