

Real-Time Data Warehousing and On-Line Analytical Processing at Aberdeen Test Center's Distributed Center

***Mr. Michael J Reil
SFA, Inc***

***Dr. Samuel F Harley
US Army Aberdeen Test Center***

***Mr. T. George Bartlett
US Army Aberdeen Test Center***



Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the 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 the 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 Washington Headquarters Services, Directorate for Information Operations and Reports, 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 a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 2005		2. REPORT TYPE		3. DATES COVERED 00-00-2005 to 00-00-2005	
4. TITLE AND SUBTITLE Real-Time Data Warehousing and On-Line Analytical Processing at Aberdeen Test Center's Distributed Center				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) US Army Aberdeen Test Center,Aberdeen Proving Ground,MD,21005				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES Modeling and Simulation Conference, 2005 Dec 12-15, Las Cruces, NM					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 37	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

ATC Distributed Center



- 32 node Linux cluster (64 processors)
- 4.4 TB Panasas storage
- Hosted at ARL-MSRC – leverages existing support

Purpose

1. Provide real-time test data verification, analysis and warehousing
2. Provide OLAP tools for test data analysis and data mining



ATC DC Proposal

- Achieve real-time data fusion to provide real-time analytic and decision support
- Establish parallel post processing capabilities to effect knowledge extraction
- Institute a high performance data warehouse
- Real time quality control – utilizing historic data sets



ATC DC Timeline

- Oct-2003 - Proposal selected
- 4-May-2004 - System Delivered
- 28-June-2004 - System on network accepting connections**
- July-2004 - System Testing Complete
- Sept-2004 - Current data handling process (SunE10K) ported to DC
- Sept-2004 - Kerberized filters in place to allow web access to data warehouse (ARL-PET Dr. Walter Landry)
- Nov-2004 – OS Change from RHES to SuSE ES9 – Slave node NFS issues**
- Dec-2004 - Processing apps running with mpiJava
- Dec-2004 - Tomcat running in a JavaParty environment
- Nov/Dec-2004 - Army Science Conference demo of Data Warehouse
- Feb-2005 - Processing apps running with Javaparty
- April-2005 - Automated scripts to poll ATC concentrator for new data files



ATC DC Proposal

- Achieve real-time data fusion to provide real-time analytic and decision support
- Establish parallel post processing capabilities to effect knowledge extraction
- Institute a high performance data warehouse
- Real time quality control – utilizing historic data sets

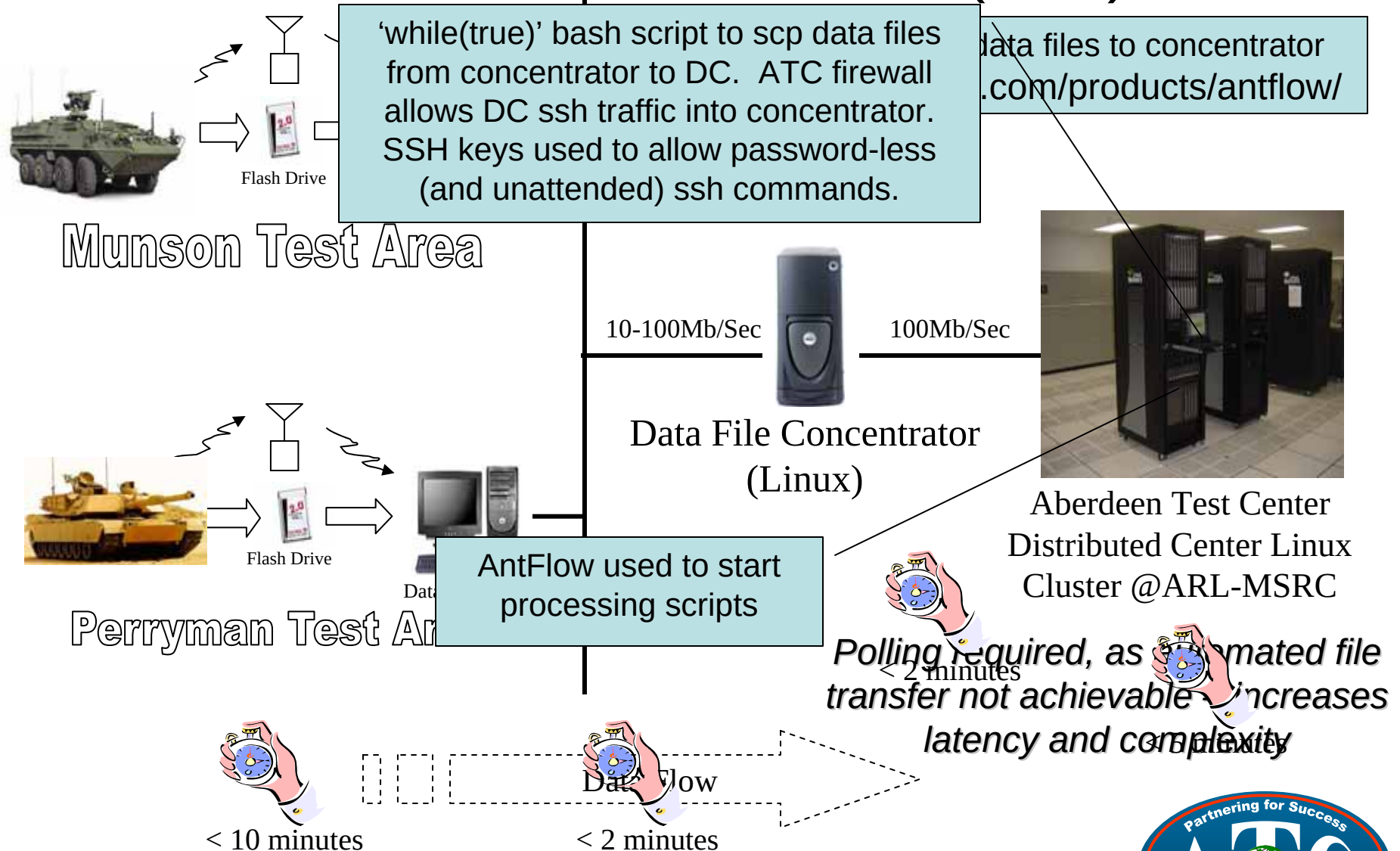


Real Time Data Fusion

- Test data collected via on-board instrumentation -VxWorks based computer. Each instrument produces a continuous time history record of up to 250 parameters, up to 10KHz ea. Files closed approx. every 15-30 minutes. Single file size from 10KB to 100MB. Test item may have multiple instruments recording simultaneously.
- Must move raw data files from instrumentation to cluster for processing. Wireless or PC-Card harvesting.
- When raw data files show up on cluster – Java based conversion (raw to HDF5) process must fire automatically.
- Report applications fire, creating reports (PDF, Excel etc.) on the just processed data.
- Reports auto-published to web based Digital Library for consumption by decision makers.
- HDF5 data files registered in data warehouse.

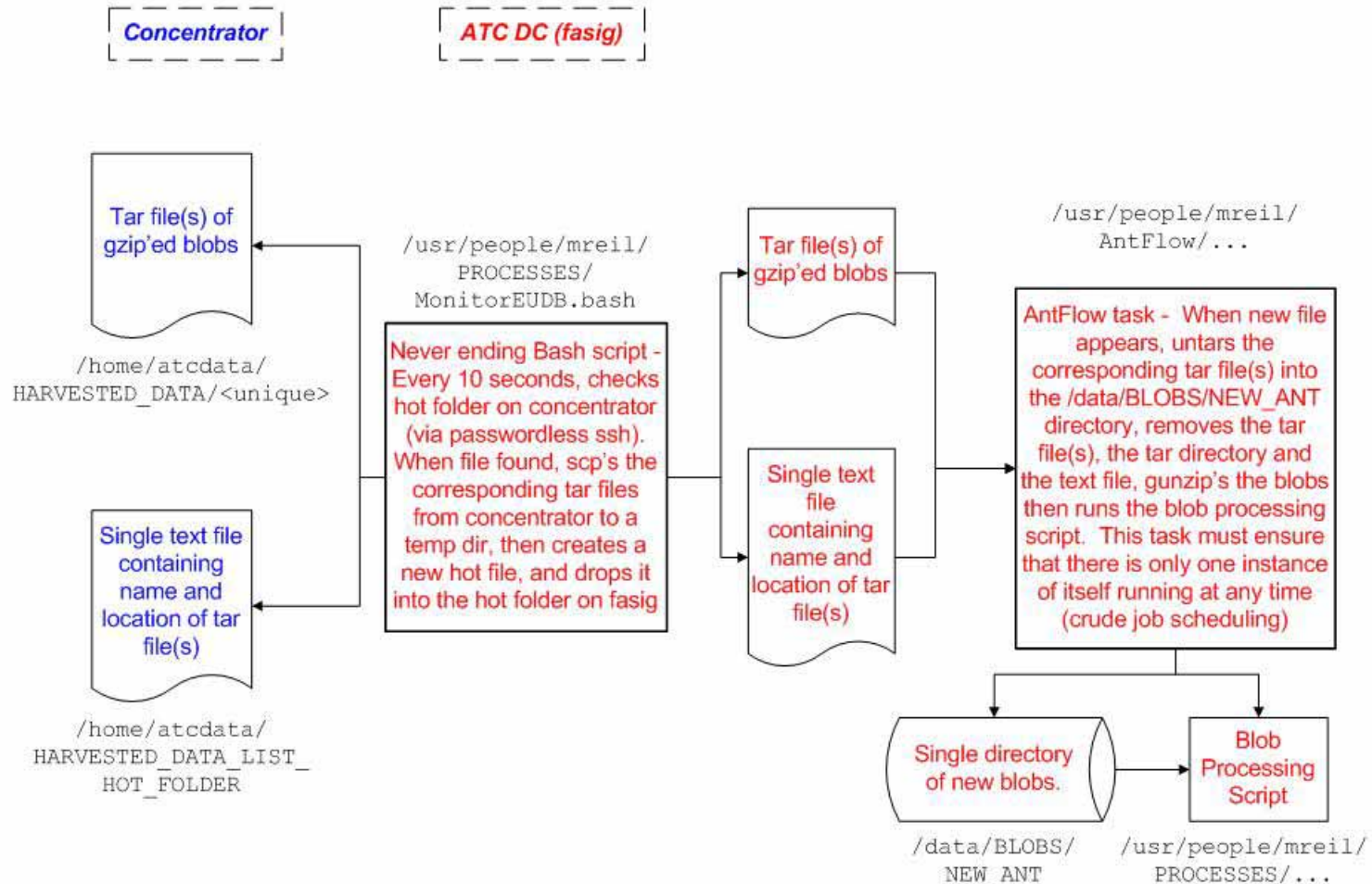


Real Time Data Fusion (cont)





Real Time Data Fusion (cont)



ATC DC Proposal

- Achieve real-time data fusion to provide real-time analytic and decision support
- Establish parallel post processing capabilities to effect knowledge extraction
- Institute a high performance data warehouse
- Real time quality control – utilizing historic data sets



Establish post processing capabilities to effect knowledge extraction

- Raw data files are converted to a common format – HDF5 chosen. (<http://hdf.ncsa.uiuc.edu/HDF5>)
- Existing library of java classes and *nix scripts to convert raw data files to HDF5. Originally single threaded java code, extended to utilize multiple java threads. Worked well on SMP machines (Sun E10K), but not on distributed processor/memory systems (Linux cluster). Processing is easy to parallelize. Each thread gets one data file to convert. Java classes used lots of memory – object oriented nature of code contributed to this – each data point was a java object. Garbage collection times also large.
- mpiJava – thin java wrapper around MPICH. Created java app that distributed processing of data files via message passing (MPI). Worked well, but required knowledge of the MPI framework and library. Also dependent on availability of MPICH for your OS/distro.



Establish post processing capabilities to effect knowledge extraction (cont)

- JavaParty – <http://www.ipd.uka.de/JavaParty/features.html> - *allows easy port of multi-threaded Java programs to distributed environments such as clusters. Regular Java already supports parallel applications with threads and synchronization mechanisms. While multi-threaded Java programs are limited to a single address space, JavaParty extends the capabilities of Java to distributed computing environments.

*From the JavaParty Web Site

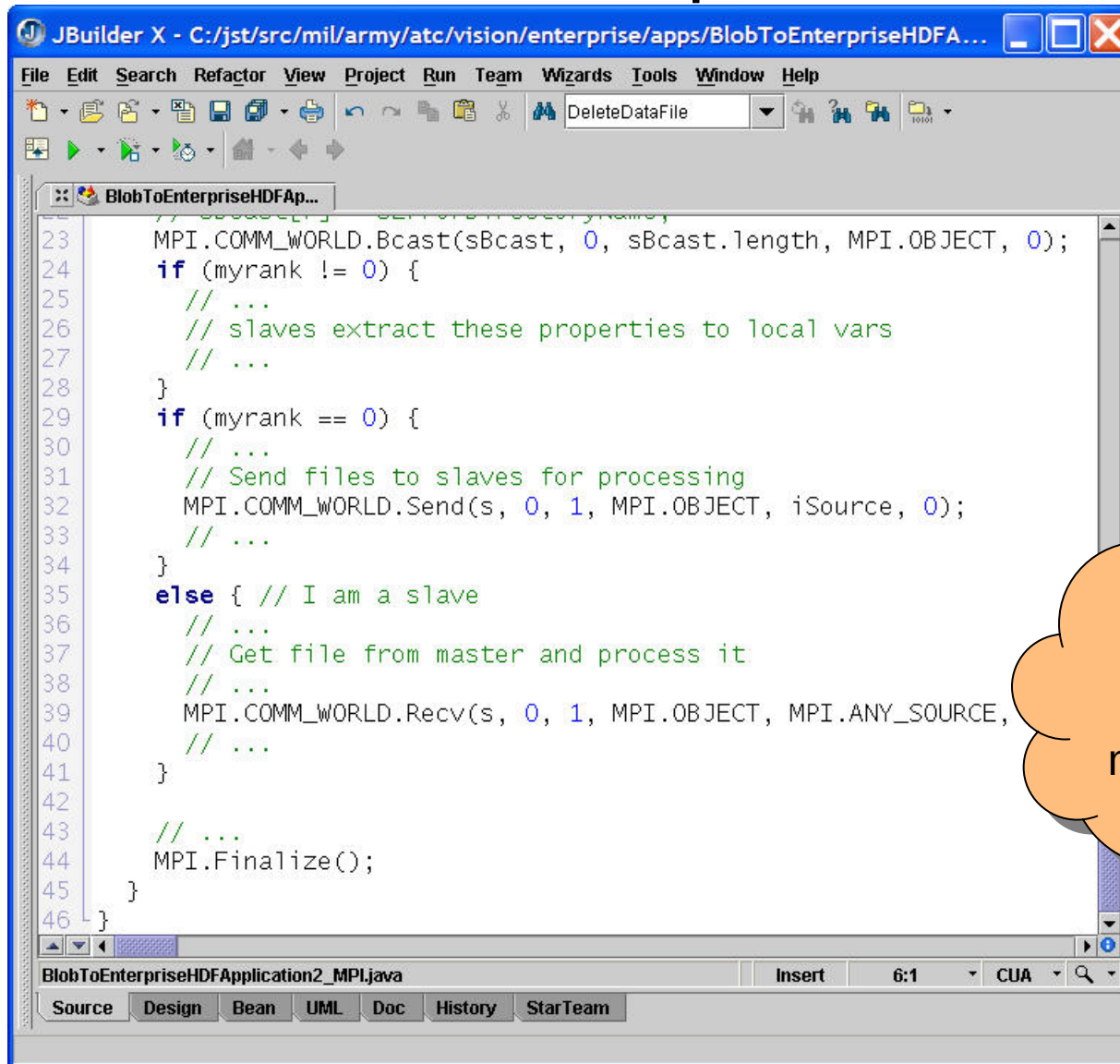


Multiple Java Threads

```
public class ConvertToHDF5 extends Thread {  
    ...  
    ...  
    ConvertToHDF5 worker = new ConvertToHDF5(...);  
    worker.start();  
    ...  
    ...  
    public void run() {  
        ...  
        ...  
    }  
}
```

Each thread is mapped to a physical processor by the JVM (Java Virtual Machine) – for SMP machines only!

mpiJava



The screenshot shows the JBuilder X IDE with a project named 'BlobToEnterpriseHDFAp...'. The main editor window displays a Java file named 'BlobToEnterpriseHDFApplication2_MPI.java'. The code is as follows:

```
23 MPI.COMM_WORLD.Bcast(sBcast, 0, sBcast.length, MPI.OBJECT, 0);
24 if (myrank != 0) {
25     // ...
26     // slaves extract these properties to local vars
27     // ...
28 }
29 if (myrank == 0) {
30     // ...
31     // Send files to slaves for processing
32     MPI.COMM_WORLD.Send(s, 0, 1, MPI.OBJECT, iSource, 0);
33     // ...
34 }
35 else { // I am a slave
36     // ...
37     // Get file from master and process it
38     // ...
39     MPI.COMM_WORLD.Recv(s, 0, 1, MPI.OBJECT, MPI.ANY_SOURCE,
40     // ...
41 }
42
43 // ...
44 MPI.Finalize();
45 }
46 }
```

The IDE interface includes a menu bar (File, Edit, Search, Refactor, View, Project, Run, Team, Wizards, Tools, Window, Help), a toolbar with various icons, and a status bar at the bottom showing 'BlobToEnterpriseHDFApplication2_MPI.java' and tabs for Source, Design, Bean, UML, Doc, History, and StarTeam.

Works well on
SMP or
Distributed
machines – it is
just MPI!



JavaParty

```
public remote class HelloJP {  
    public void hello() {  
        // Print on the console of the virtual machine where the object lives  
        System.out.println("Hello JavaParty!");  
    }  
    public static void main(String[] args) {  
        for (int n = 0; n < args.length; n++) {  
            // Create a remote object  
            HelloJP world = new HelloJP(n);  
            // Remotely invoke the hello() method  
            world.hello();  
        }  
    }  
}
```

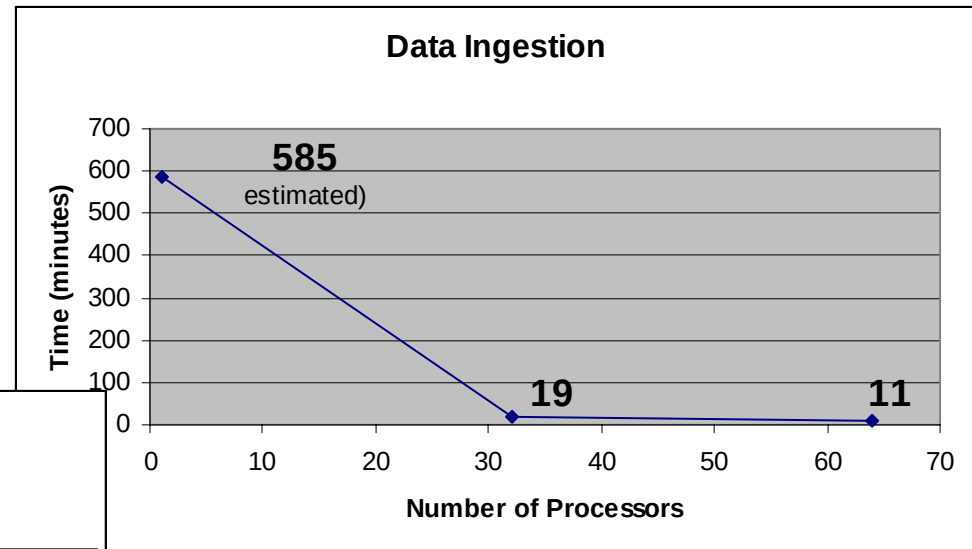
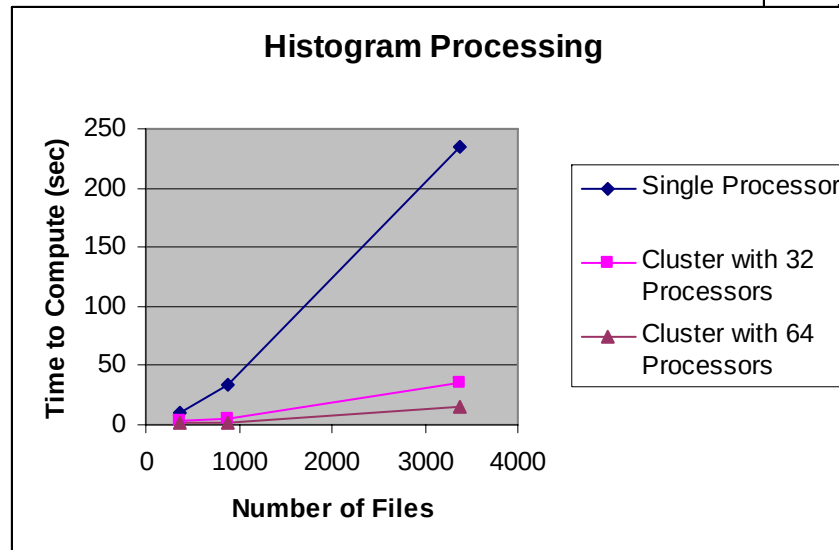
Each new 'remote' object is created on a slave processor. User can control which processor with the `/** @i */` construct in code

JavaParty

- Uses ssh to spawn JVMs on slave nodes of cluster (similar to MPI)
- One JVM per slave processor.
- Controlled via .jp-nodefile (similar to 'machines' file used with MPI).
- Pure java implementation – no native libraries required.
- Uses RMI to serialize java objects between JVMs.
- High performance RMI engine supplied (KaRMI).
- Possible to use without 'breaking' java source code – extend 'RemoteThread' class instead of using 'remote' keyword.
- This is the framework that we are now using.
- Regular java – invoke application :
 - `java <classname>`
- JavaParty – invoke application :
 - `jpinvite <classname>`



Establish post processing capabilities to effect knowledge extraction (cont)



ATC DC Proposal

- Achieve real-time data fusion to provide real-time analytic and decision support
- Establish parallel post processing capabilities to effect knowledge extraction
- Institute a high performance data warehouse
- Real time quality control – utilizing historic data sets



What is OLAP?

- Online Analytical Processing
- Software that enables decision support via rapid queries to large databases that store ~~corporate~~ data in multidimensional hierarchies and views.

T&E



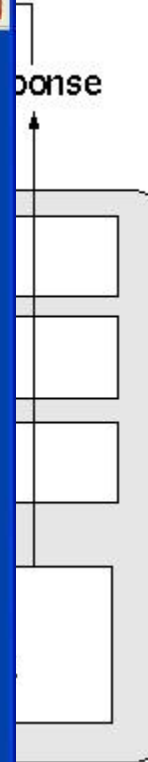
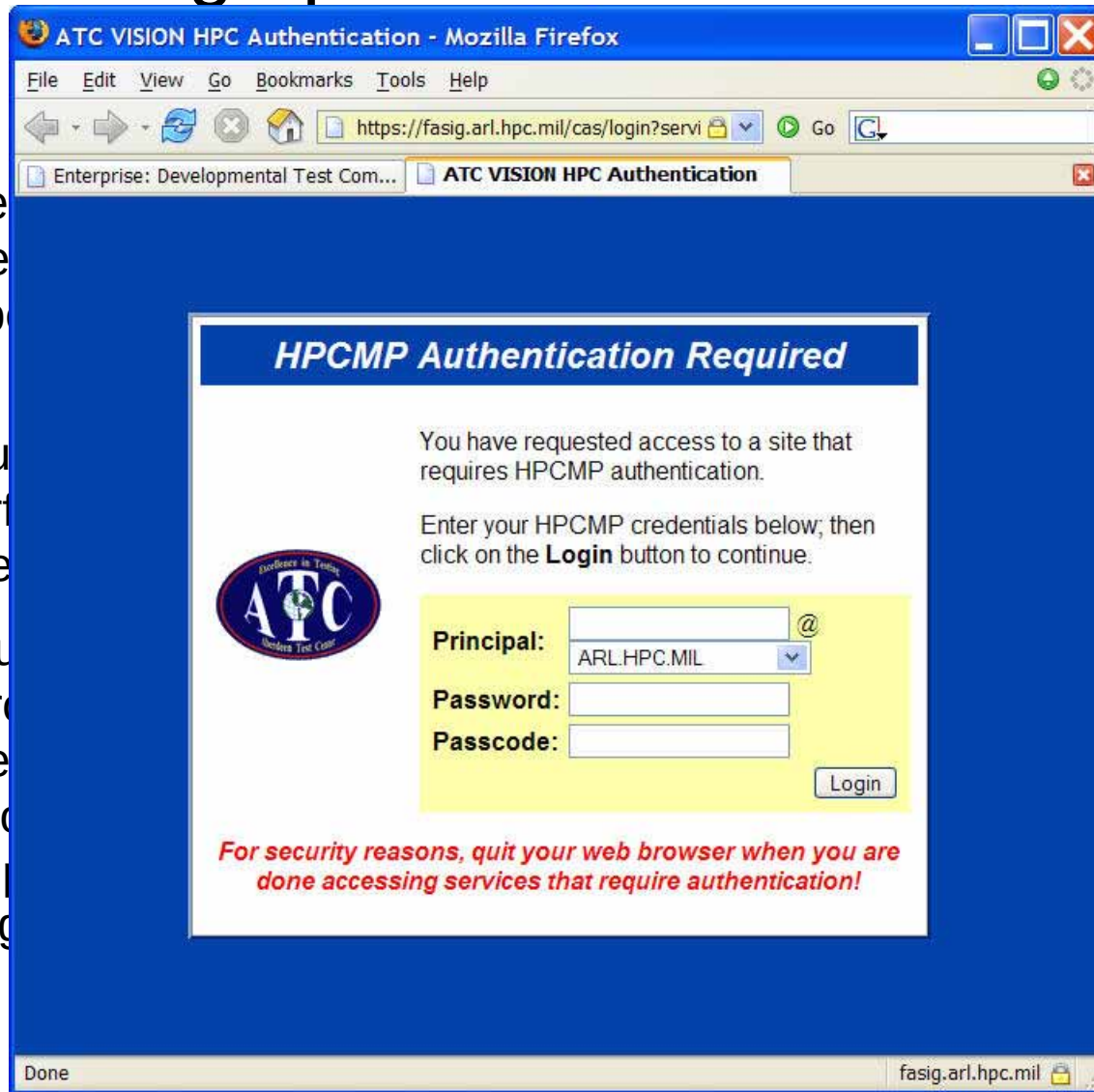
Institute a high performance data warehouse

- PostgreSQL 7.4 installed on dedicated filesystem (500 MB RAID5 JBOD) on head node.
- Java based web application ported to JavaParty. Allows data set queries submitted by the web app user to be run on all nodes of the cluster in parallel (for aggregate operations). Tomcat started via 'javaparty' rather than the standard 'java'. This allows servlets to create remote objects, which run on the remote nodes.
- Kerberos/SecureID authentication module written by PET IMT – Dr. Walter Landry @ ARL. Uses J2EE servlet filter framework and cookies to authenticate each HTTP request.
- GUI is java applet, which runs in users browser. GUI presents metadata to user, who selects filter settings, and applet then submits SQL statement on users behalf to data warehouse. List of data sets is returned – user can then request composite routines be run on the set of data files – these are run on the entire cluster in the JavaParty environment.

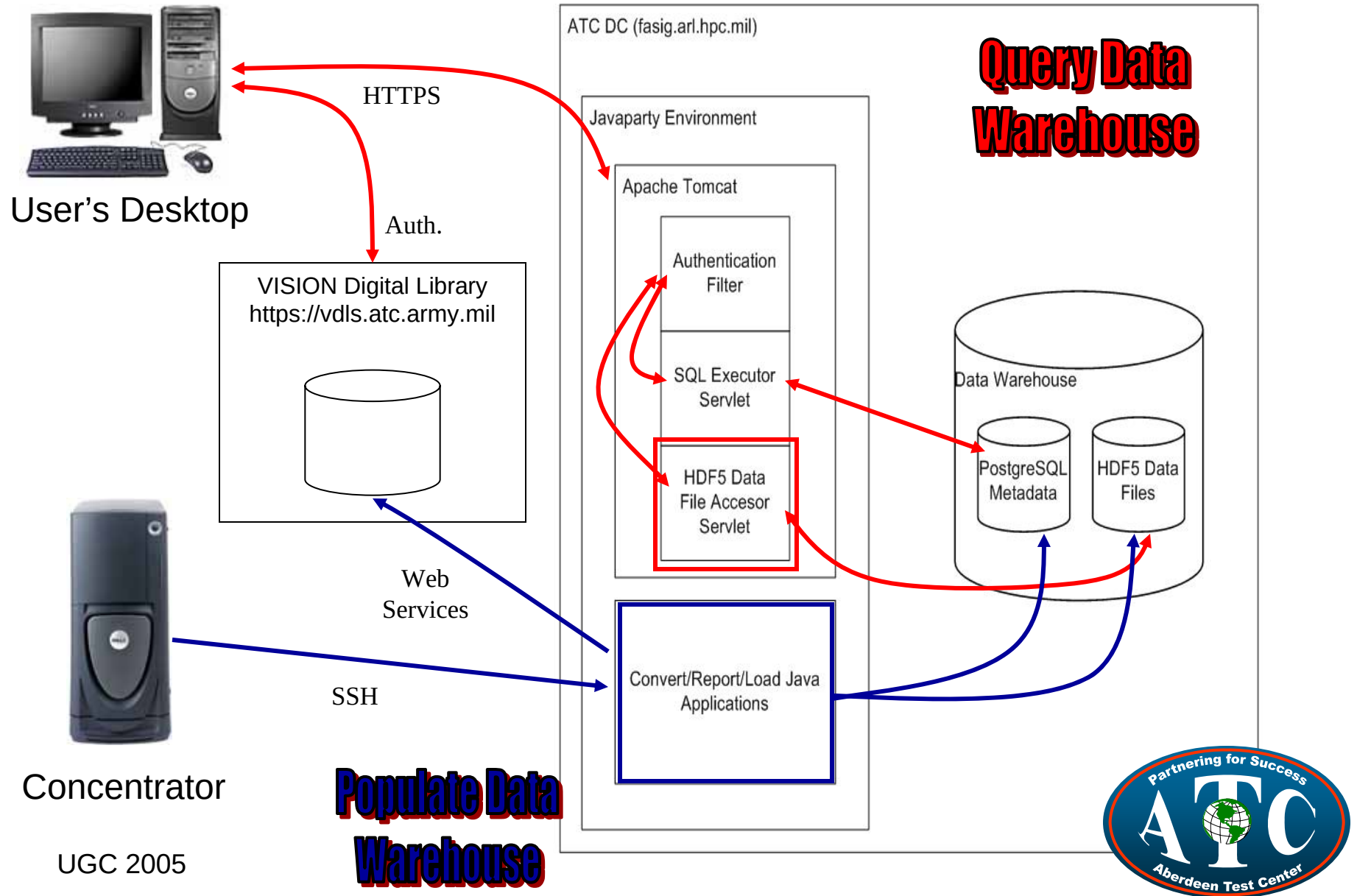


Institute a high performance data warehouse

- Filter re
HTTP re
the prop
page.
- User su
filter per
encrypted
- Each su
goes thro
contain
for this t
request
login pag



Institute a high performance data warehouse



Screenshots Of OLAP GUI

UGC 2005





File

Select Filters To Be Applied To List Of All Data Files, And Then Click Next Button

Select Minimums

File Duration Minutes

Your Selected Filters

Inventory Item is 'Test Item 1' AND COURSE_NAME is 'd' AND COURSE_COND is 'Wet' OR 'Damp' AND VEHICLE_CONFIG is 'Loaded Boulders' OR 'Loaded+Gravel'

Select Static Metadata Filter

Comments Inventory Item

Test Location Instrumentation

Test Item 1
Test Item 2

Clear Inventory Item

Select Dynamic Metadata Filter

VERSION_NUMBER Testing_Configuration TOWING_CONFIG

COURSE_COND VEHICLE_CONFIG COURSE_NAME DRIVER_NAME

Churchville_B
Belgian_Block_Gravel
Perryman_2
Perryman_Paved
Perryman_A
Paved_Between_Test_Courses
Munson_Gravel
Perryman_3

Clear COURSE_NAME

Operations On Data Files That Match Filters

Get Count Of Matching Files 4 Files

Calendar Of Matching Files (minus calendar filter)

Appointment Book Style Of Matching Files

Select Date Range

January 2005 May 19, 2005

January							February							March							
Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	
						1			1	2	3	4	5				1	2	3	4	5
2	3	4	5	6	7	8	6	7	8	9	10	11	12	6	7	8	9	10	11	12	
9	10	11	12	13	14	15	13	14	15	16	17	18	19	13	14	15	16	17	18	19	
16	17	18	19	20	21	22	20	21	22	23	24	25	26	20	21	22	23	24	25	26	
23	24	25	26	27	28	29	27	28					27	28	29	30	31				
30	31																				

April							May							June						
Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2	1	2	3	4	5	6	7				1	2	3	4
3	4	5	6	7	8	9	8	9	10	11	12	13	14	5	6	7	8	9	10	11
10	11	12	13	14	15	16	15	16	17	18	19	20	21	12	13	14	15	16	17	18
17	18	19	20	21	22	23	22	23	24	25	26	27	28	19	20	21	22	23	24	25
24	25	26	27	28	29	30	29	30	31				26	27	28	29	30			

July							August							September						
Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2	1	2	3	4	5	6					1	2	3	
3	4	5	6	7	8	9	7	8	9	10	11	12	13	4	5	6	7	8	9	10
10	11	12	13	14	15	16	14	15	16	17	18	19	20	11	12	13	14	15	16	17
17	18	19	20	21	22	23	21	22	23	24	25	26	27	18	19	20	21	22	23	24
24	25	26	27	28	29	30	28	29	30	31			25	26	27	28	29	30		
31																				

October							November							December						
Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat
						1			1	2	3	4	5					1	2	3
2	3	4	5	6	7	8	6	7	8	9	10	11	12	4	5	6	7	8	9	10
9	10	11	12	13	14	15	13	14	15	16	17	18	19	11	12	13	14	15	16	17
16	17	18	19	20	21	22	20	21	22	23	24	25	26	18	19	20	21	22	23	24
23	24	25	26	27	28	29	27	28	29	30			25	26	27	28	29	30	31	
30	31																			

☐ Use Calendar In Filter

UGC 2005

VISION EU Database -

Test Item 1

File

Select One Or More Data Files And One Of The Functions Selected 1 of 216

res_uid	Asset	Test Item	Test Ctr. Id	Instr.	Loaded Int...	File Start Time	File End Time	DRIVER_N...	COURSE_NAME
7C12440103			K2	Analog	2005-01-12 1	2004-03-05 07:44:01 EST	2004-03-05 07:44:09 EST	Jim+Craig	Perryman_Paved
FC12440203	Test Item 1		K2	GPS	2005-01-12 1	2004-03-05 07:44:02 EST	2004-03-05 07:44:09 EST	Jim+Craig	Perryman_Paved
8912454603			K2	GPS	2005-01-12 2	2004-03-05 07:45:46 EST	2004-03-05 07:46:21 EST	Jim+Craig	Perryman_Paved
5C12454603			K2	Analog	2005-01-12 1	2004-03-05 07:45:46 EST	2004-03-05 07:46:21 EST	Jim+Craig	Perryman_Paved
D214533303			K2	GPS	2005-01-12 1	2004-03-08 09:53:33 EST	2004-03-08 10:23:34 EST	Jim Craig	Perryman_Paved
5914533303			K2	Analog	2005-01-12 2	2004-03-08 09:53:33 EST	2004-03-08 10:23:35 EST	Jim Craig	Perryman_Paved
1B15233603			K2	Analog	2005-01-12 1	2004-03-08 10:23:36 EST	2004-03-08 10:53:38 EST	Jim Craig	Perryman_Paved
F015233603			K2	GPS	2005-01-12 1	2004-03-08 10:23:36 EST	2004-03-08 10:53:38 EST	Jim Craig	Perryman_Paved
C215534003			K2	GPS	2005-01-12 1	2004-03-08 10:53:40 EST	2004-03-08 11:23:42 EST	Jim Craig	Perryman_Paved
1915534003			K2	Analog	2005-01-12 1	2004-03-08 10:53:40 EST	2004-03-08 11:23:42 EST	Jim Craig	Perryman_Paved
8816234403			K2	GPS	2005-01-12 2	2004-03-08 11:23:44 EST	2004-03-08 11:29:59 EST	Jim Craig	Perryman_Paved
DA16234403			K2	Analog	2005-01-12 2	2004-03-08 11:23:44 EST	2004-03-08 11:29:59 EST	Jim Craig	Perryman_Paved
7B17284803			K2	GPS	2005-01-12 1	2004-03-08 12:28:48 EST	2004-03-08 12:58:49 EST	Jim Craig	Perryman_Paved
BF17284903			K2	Analog	2005-01-12 1	2004-03-08 12:28:49 EST	2004-03-08 12:58:50 EST	Jim Craig	Perryman_Paved
6217585103			K2	Analog	2005-01-12 1	2004-03-08 12:58:51 EST	2004-03-08 13:28:54 EST	Jim Craig	Perryman_Paved
8017585103			K2	GPS	2005-01-12 1	2004-03-08 12:58:51 EST	2004-03-08 13:28:53 EST	Jim Craig	Perryman_Paved
9C18285503			K2	Analog	2005-01-12 2	2004-03-08 13:28:55 EST	2004-03-08 13:41:38 EST	Jim Craig	Perryman_Paved
2618285503			K2	GPS	2005-01-12 1	2004-03-08 13:28:55 EST	2004-03-08 13:41:38 EST	Jim Craig	Perryman_Paved
EA19092203			K2	GPS	2005-01-12 1	2004-03-08 14:09:22 EST	2004-03-08 14:39:23 EST	Jim Craig	Perryman_Paved
0119092303			K2	Analog	2005-01-12 1	2004-03-08 14:09:23 EST	2004-03-08 14:39:24 EST	Jim Craig	Perryman_Paved
4119392503			K2	GPS	2005-01-12 1	2004-03-08 14:39:25 EST	2004-03-08 14:41:47 EST	Jim Craig	Perryman_Paved
DC19392503			K2	Analog	2005-01-12 1	2004-03-08 14:39:25 EST	2004-03-08 14:41:47 EST	Jim Craig	Perryman_Paved
0222034503			K2	GPS	2005-01-12 1	2004-03-08 17:03:45 EST	2004-03-08 17:33:46 EST	Jim Blackburn	Perryman_Paved
6722034803			K2	Analog	2005-01-12 1	2004-03-08 17:03:48 EST	2004-03-08 17:33:49 EST	Jim Blackburn	Perryman_Paved

Your Selected Filters

Inventory Item is Test Item 1

AND COURSE_NAME is 'Perryman_Paved'

GPS Map

Data File Summary

One Channel Across Files

Multi Channel Multi File Plot

Reprocess with Alternate Config

Download Data File(s)

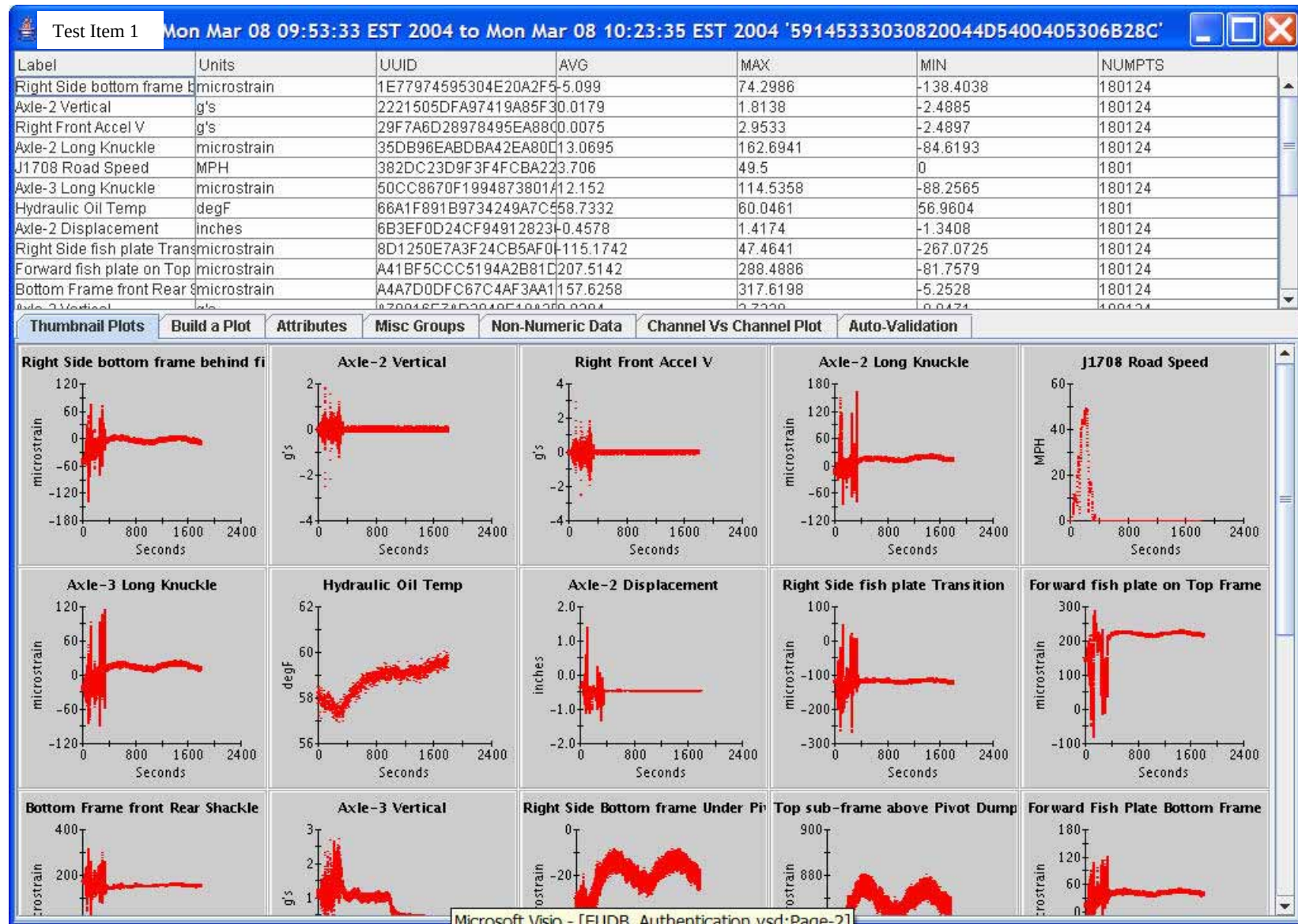
Associate Comment To File(s)

Select All

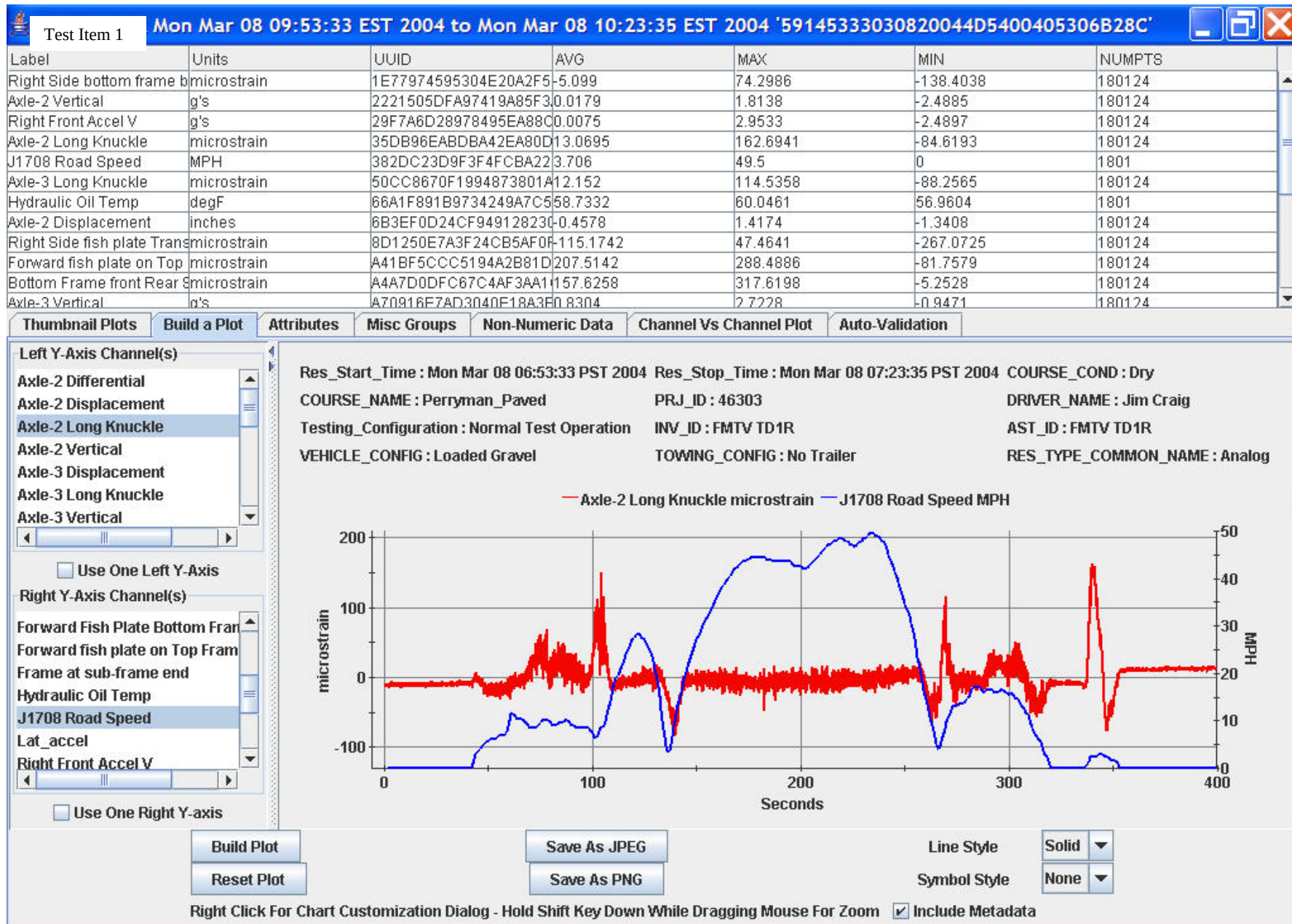
Clear All

Microsoft Visio - [EUIDB_Authentication.vsd:Page-2]

UGC 2005



UGC 2005



UGC 2005

Test Item 1 Mon Mar 08 09:53:33 EST 2004 to Mon Mar 08 10:23:34 EST 2004 'D2145333030820044D5400405300BE6D'						
Label	Units	UUID	AVG	MAX	MIN	NUMPTS
Road Speed	MPH	3BA27795418411D6A1BD000	3.7019	49.5	0	14988
Throttle Position	%	3BA27797418411D6A1BD000	9.5888	99.6	2.8	14988
Engine Speed	RPM	3BA2779F418411D6A1BD000	870.3943	2768.5	529.25	14988
Total Miles	miles	3BA277A5418411D6A1BD000	1589.4011	1589.6	1587.8	188
Trans Range Selected	ASCII	3BA277A9418411D6A1BD000				3657
Trans Range Attained	ASCII	3BA277AA418411D6A1BD000				3657
Ticker	Tenths Of mSecs	3BA277D3418411D6A1BD000	2073753983.0028	4177089792	484352	1801
UTC	DateTime	3BA277D4418411D6A1BD000				1801
Latitude	Degrees	3BA277D5418411D6A1BD000	39.4595	39.4616	39.4417	1801
Longitude	Degrees	3BA277D6418411D6A1BD000	-76.1915	0	-76.2054	1801
WGS84 Altitude	Meters	3BA277D7418411D6A1BD000	17.3537	27	9	1801
UTM Easting	Meters	3BA277D8418411D6A1BD000	207407.4533	207623.3640	206204.4610	1801
Thumbnail Plots Build a Plot Attributes Misc Groups Misc Images Non-Numeric Data Channel Vs Channel Plot Auto-Validation						
UUID	Channel Name	Points	Time	Trans Range Selected	Trans Range Attained	
3BA277A9418411D6A1BD000	Trans Range Selected	3657	154.8832	7	L6	
3BA277AA418411D6A1BD000	Trans Range Attained	3657	155.3928	7	L6	
3BA277D4418411D6A1BD000	UTC	1801	155.8666	7	L6	
3BA277DB418411D6A1BD000	UTM Zone (Latitude)	1801	156.3678	7	L6	
			156.8492	7	L6	
			157.351	7	L6	
			157.8404	7	L6	
			158.3346	7	L6	
			158.8154	7	L6	
			159.317	7	L6	
			159.7984	7	L6	
			160.3044	7	L6	
			160.7818	7	L6	
			161.283	7	L7	
			161.7648	7	L7	
			162.2844	7	L7	
			162.7526	7	L7	
			163.2492	7	L7	
			163.731	7	L7	
			164.2322	7	L7	
			164.727	7	L7	
			165.2152	7	L7	
			165.6972	7	L7	
			166.1984	7	L7	
			166.68	7	L7	
			167.195	7	L7	
			167.6862	7	L7	

UGC 2005



Your Selected Filters

Inventory Item is Test Item 1
AND COURSE_NAME is 'Perryman_Paved'

Summary Histogram Threshold Crossing Time In Limits Trend

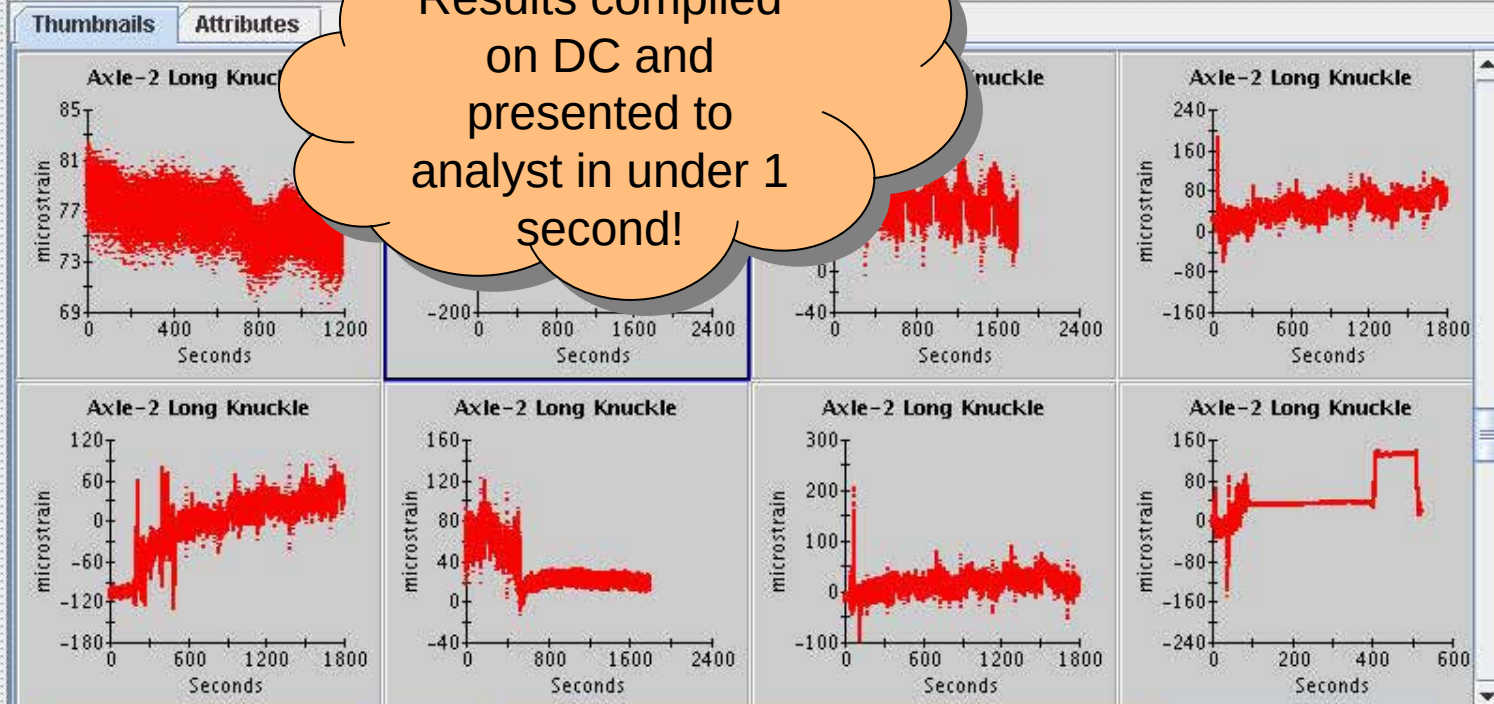
Show Summary

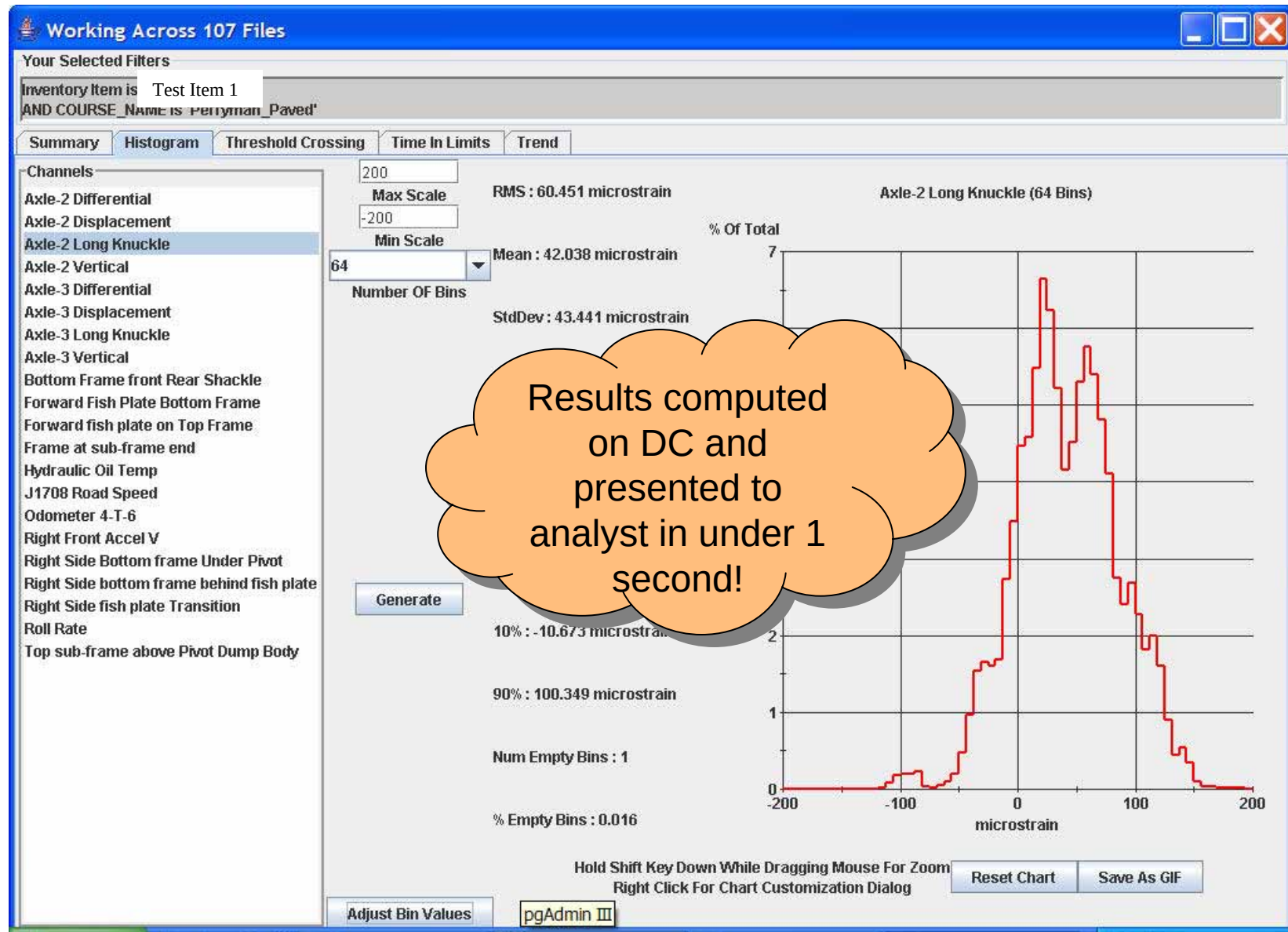
Channels

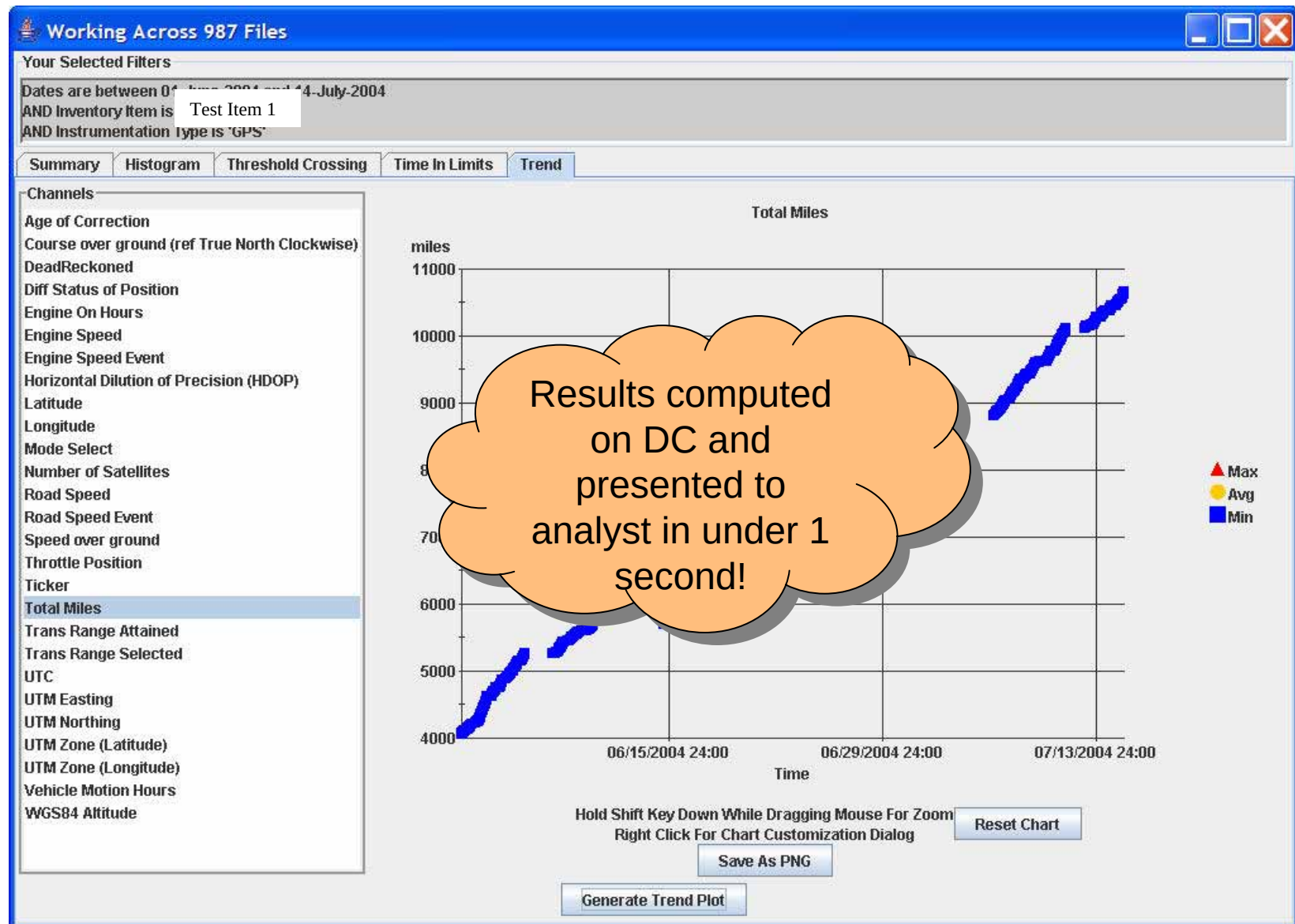
Axle-2 Differential
Axle-2 Displacement
Axle-2 Long Knuckle
Axle-2 Vertical
Axle-3 Differential
Axle-3 Displacement
Axle-3 Long Knuckle
Axle-3 Vertical
Bottom Frame front Rear Shackl
Forward Fish Plate Bottom Frame
Forward fish plate on Top Frame
Frame at sub-frame end
Hydraulic Oil Temp
J1708 Road Speed
Odometer 4-T-6
Right Front Accel V
Right Side Bottom frame Under P
Right Side bottom frame behind f
Right Side fish plate Transition
Roll Rate
Top sub-frame above Pivot Dump

File UUID	Min	Max	Avg	Number Points
DA162344030820044D5400	1840.957595444	398.41105502	112.31489896300035	37365
EE224443062320044D5400	63.837621496000004	387.50354775	110.52505178026924	180014
57214928071420044D5400	35.363286728	353.862499012	255.21904543298464	179938
8F113011060320044D5400	143.5198325	345.136493196	0.5696571194218909	179745
86045342060320044D5400	80.485922066	344.792045598	7.295762830893989	180113
56135101062420044D5400	17.337195766	340.54385855600003	54.625971608784106	179684
3A193419060220044D5400	43.859660812	326.306691172	49.924054971597656	87483
89224934071420044D5400	33.985496336000004	316.31771083	132.162949719261	179958
0D144519062320044D5400	29.163229964	300.81756892	89.46369568277613	118716
9A234941071420044D5400	39.267026172		115.3701754150899	180059
14121503062320044D5400	2.4111		1.16588229179067	180018

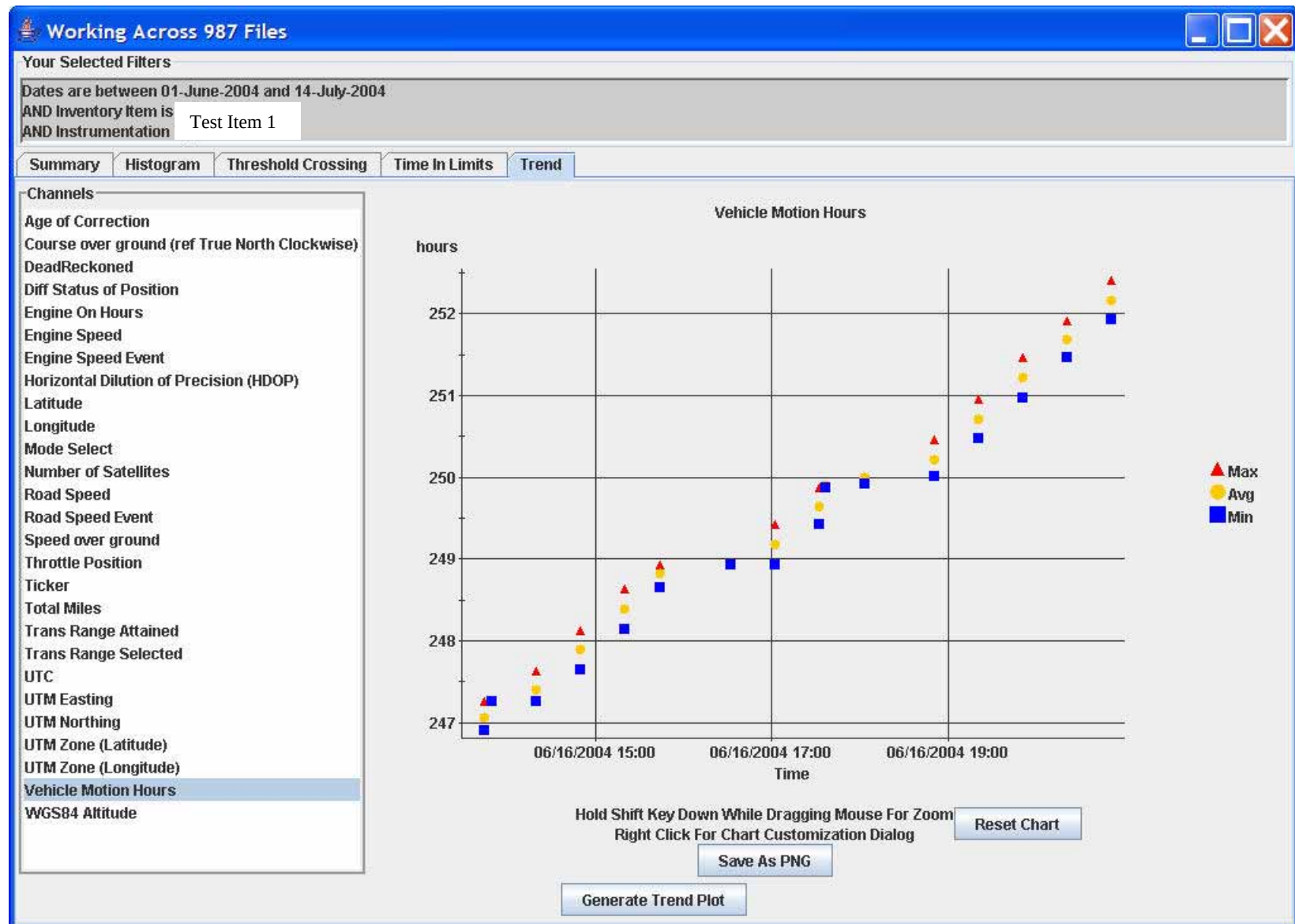
Results compiled
on DC and
presented to
analyst in under 1
second!







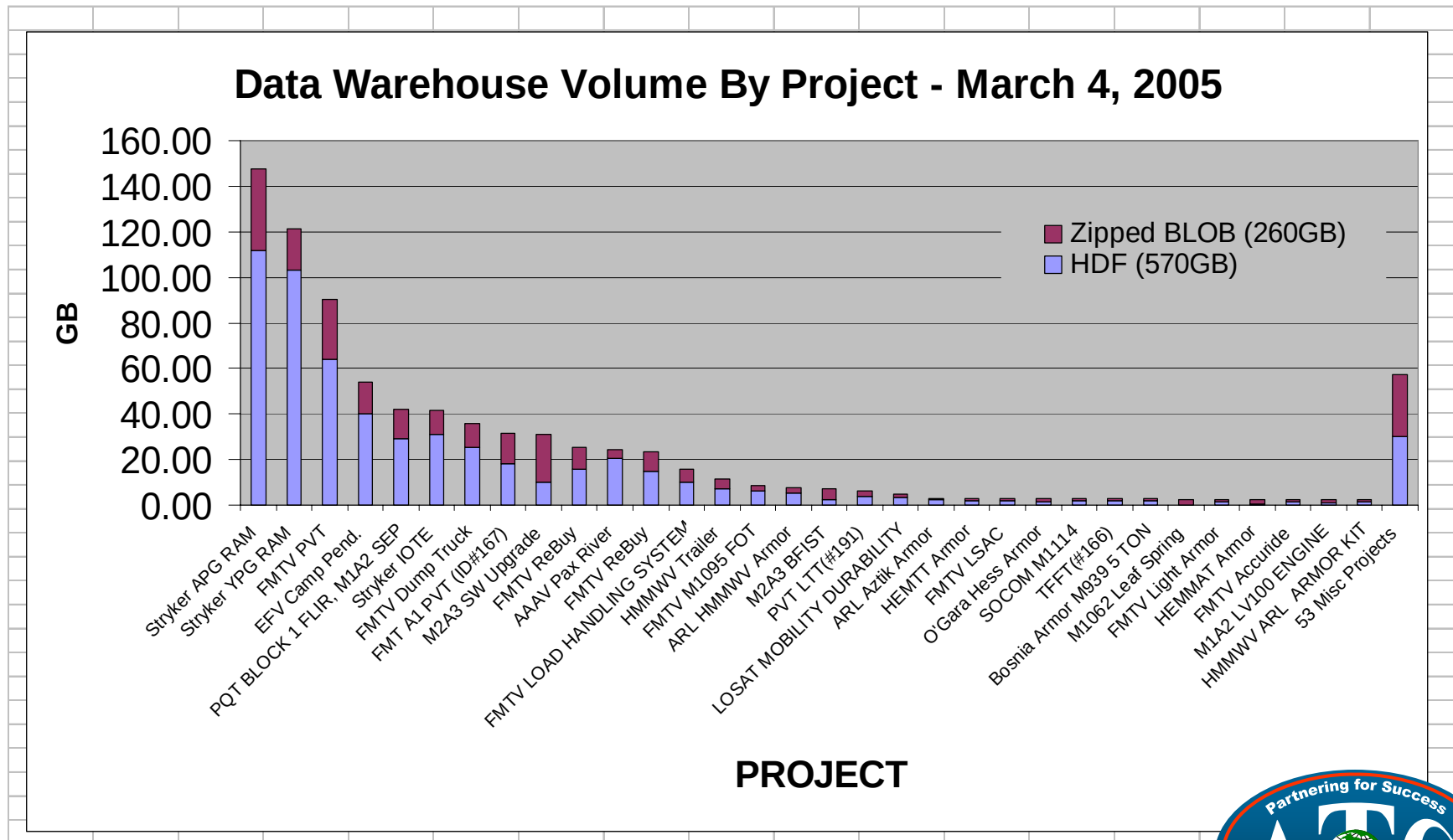
UGC 2005



UGC 2005

Institute a high performance data warehouse

Over 80 projects using Data Warehouse



UGC 2005



ATC DC Proposal

- Achieve real-time data fusion to provide real-time analytic and decision support
- Establish parallel post processing capabilities to effect knowledge extraction
- Institute a high performance data warehouse
- Real time quality control – utilizing historic data sets



Real time quality control – utilizing historic data sets

- New data sets compared with warehoused data from the same channel/test item for anomaly detection.
- Future Work



Summary

- Parallel java applications are running very well on cluster.
- Polling vs. interrupt (event) driven processing not ideal – but workable.
- ARL MSRC administering the system is ideal.
- Data warehouse access requiring kerberos/secureID does not fit well with our current Digital Library project based authentication. ATC customers must obtain HPCMP account in order to use data warehouse (they don't even know they are using HPCMP assets).
- *Special thanks to Tom Kendall, Chris Slaughter and Ryan Baxter at ARL-MSRC for assistance every step of the way!*

Partnering For Success

