RSC Trending Tool for Archiving, Graphing, and Analysis of State of Health Telemetry System Design and Theory

Presented at the GSAW 2005, Manhattan Beach, CA

Air Force Space and Missile Systems Center (SMC) Vehicle Operations Directorate at the Ground Systems Research, Development, Test, and Evaluation Support Complex (RSC)

Kirtland AFB, New Mexico

Joel Castellanos – ASRC Aerospace
Rick Kapalko – ASRC Aerospace
Lt. Joseph Spagnolia – United States Air Force
Chris Worth – Lockheed Martin Mission Systems
Purpose

1. To present a ground system trending tool that is useful to the RSC and may be useful to others
   a) Either as is or,
   b) With source code that can be site customized.

2. To present a case for small group development of in-house software rather than using Commercial Off-the-Shelf (COTS) software.

3. To present a few specialized database tricks, algorithms, and design strategies.
Topics

• General Description.
• User Interface.
• Data Patterns and Access Patterns that allow for specialized data management offering significant advantages over general database programs.
• Hardware Environment.
• Maintainability with Dynamic Requirements.
Input Data

- Unique Vehicles: not constellations
- Not Real-Time
- GMT Time Series, Engineering Unit (EU) State of Health (SOH) data.
- ASCII, Comma Delimited.
- Support 3 to 7 Satellites.
- Each Satellite: 1 to 7 year Missions.
- Each Satellite: Thousands of mnemonics each sampled as much as tens of times per second (i.e. up to 1 trillion points for some vehicles).
- Not all Mnemonics Sampled at the Same Rate.
- Stored SOH (SSOH), Real-time SOH (RTSOH), etc.
- One or More Input File Per Contact.
- Time Interlaced Data on Successive Contacts.
Supported Vehicles

- **C/NOFS**
  - Not yet launched
  - FCT Data used
- **CloudSat**
  - Not yet launched
  - FCT Data used
- **Orbital Express**
  - Not yet launched
  - Data not yet available
- **STPSat-1**
  - Not yet launched
  - Data not yet available

- **Coriolis**
  - Operational
  - Launched Jan, 2003
Requirements

- Short-Term Anomaly Resolution.
- Long-Term Trend Analysis.
- Reasonable Query Results in Interactive Time (seconds to minutes).
- Support Simultaneous Users.
- Simple, but Expandable Statistical Analysis.
- Query Accumulated Statistical Results.
- Interactive Graphing.
- Export results to GIF, Excel, MatLab, etc.
- Delete / Replace data by Contact and by Point.
- All GUI functions and configurations must be savable to a script that can be run automatically from a cron job on different date ranges.
Administration Tab

RSC Trending Tool - Version 2005-01

File  OutPut
IRON: gsaw_2004

Query Tool  Statistical Database  Derived Mnemonics  System Monitor  Administration

Add Vehicle  Delete Selected Vehicle

Vehicle Configuration

Add SOH Type
Import Mnemonic Definition File

RAID Allotment (MB): 30000
Default Ingest Path: ../TrendIngest/
Tape Staging Path: ../tape/
Default Data & Graph Output Path: /
Default Query Script Path: /
Allowed Ingest Dates: 2002-01-01 Through (days offset to current local): 5
Maximum Days in Single Ingest File: 7

SOH List:
SSOH
RTSOH
System Monitor Tab

RAID Status
- Total MB: 30,000
- Used MB: 2,630
- Free MB: 27,370

RAID Management
- Action: View Selected Ingest
- View Selected Ingest
- Copy Selected Ingest(s) to Tape Directory
- Restore IRON's Data from Tape Directory to RAID
- Remove Selected Ingest(s) from RAID
- Expunge Selected Ingests(s) from Database

SSOH
Check Syntax Only
Ingest
### RSC Trending Tool - Version 2005-01

#### 7.00_ST_01_EU1_trend

<table>
<thead>
<tr>
<th>GMT</th>
<th>BAT_1_VOLTAGE</th>
<th>BAT_2_VOLTAGE</th>
<th>KGRSTCNT</th>
<th>KGRSTCN</th>
<th>SBNDATXP</th>
<th>SBRX_A_LS</th>
<th>SBRX_B_LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003-01-06 21:33:35.88...</td>
<td>1.2122511</td>
<td>1.212524</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-185.73149</td>
<td>-130.93133</td>
</tr>
<tr>
<td>2003-01-06 21:33:37.28...</td>
<td>1.216368</td>
<td>1.2167335</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-185.73149</td>
<td>-130.93133</td>
</tr>
<tr>
<td>2003-01-06 21:33:38.88...</td>
<td>1.2206015</td>
<td>1.220603</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-185.73149</td>
<td>-130.93133</td>
</tr>
<tr>
<td>2003-01-06 21:33:40.08...</td>
<td>1.224618</td>
<td>1.2252045</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-185.73149</td>
<td>-130.93133</td>
</tr>
<tr>
<td>2003-01-06 21:33:41.48...</td>
<td>1.2288199</td>
<td>1.2294663</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-185.73149</td>
<td>-130.93133</td>
</tr>
<tr>
<td>2003-01-06 21:33:42.88...</td>
<td>1.2330029</td>
<td>1.2337457</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-58.17494</td>
<td>3.17157</td>
</tr>
<tr>
<td>2003-01-06 21:33:44.28...</td>
<td>1.2372037</td>
<td>1.2380428</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-58.17494</td>
<td>3.17157</td>
</tr>
<tr>
<td>2003-01-06 21:33:45.68...</td>
<td>1.2414215</td>
<td>1.2423576</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-58.17494</td>
<td>3.17157</td>
</tr>
<tr>
<td>2003-01-06 21:33:47.08...</td>
<td>1.2456663</td>
<td>1.2466902</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-58.17494</td>
<td>3.17157</td>
</tr>
<tr>
<td>2003-01-06 21:33:48.48...</td>
<td>1.249908</td>
<td>1.2510405</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>-58.17494</td>
<td>3.17157</td>
</tr>
<tr>
<td>2003-01-06 21:33:49.88...</td>
<td>1.2541769</td>
<td>1.2554086</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>70.12539</td>
<td>27.763372</td>
</tr>
<tr>
<td>2003-01-06 21:33:51.28...</td>
<td>1.2584629</td>
<td>1.2597948</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>70.12539</td>
<td>27.763372</td>
</tr>
<tr>
<td>2003-01-06 21:33:52.68...</td>
<td>1.2627661</td>
<td>1.264199</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>70.12539</td>
<td>27.763372</td>
</tr>
<tr>
<td>2003-01-06 21:33:54.09...</td>
<td>1.2670665</td>
<td>1.268212</td>
<td>4145151</td>
<td>4320000</td>
<td>0</td>
<td>70.12539</td>
<td>27.763372</td>
</tr>
<tr>
<td>2003-01-06 21:33:55.49...</td>
<td>1.2714243</td>
<td>1.2730614</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>70.12539</td>
<td>27.763372</td>
</tr>
<tr>
<td>2003-01-06 21:33:56.89...</td>
<td>1.2757794</td>
<td>1.2775198</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-186.10338</td>
<td>46.203033</td>
</tr>
<tr>
<td>2003-01-06 21:33:58.29...</td>
<td>1.2801517</td>
<td>1.2819964</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-186.10338</td>
<td>46.203033</td>
</tr>
<tr>
<td>2003-01-06 21:33:59.69...</td>
<td>1.2845417</td>
<td>1.2864913</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-186.10338</td>
<td>46.203033</td>
</tr>
<tr>
<td>2003-01-06 21:34:01.09...</td>
<td>1.288949</td>
<td>1.2910044</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-186.10338</td>
<td>46.203033</td>
</tr>
<tr>
<td>2003-01-06 21:34:02.49...</td>
<td>1.293374</td>
<td>1.2955359</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-186.10338</td>
<td>46.203033</td>
</tr>
<tr>
<td>2003-01-06 21:34:03.89...</td>
<td>1.2978165</td>
<td>1.3000859</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-57.80305</td>
<td>70.42769</td>
</tr>
<tr>
<td>2003-01-06 21:34:05.29...</td>
<td>1.3022767</td>
<td>1.3046544</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-57.80305</td>
<td>70.42769</td>
</tr>
<tr>
<td>2003-01-06 21:34:06.69...</td>
<td>1.3067545</td>
<td>1.3092413</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
<tr>
<td>2003-01-06 21:34:08.09...</td>
<td>1.3112501</td>
<td>1.3138468</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-57.80305</td>
<td>70.42769</td>
</tr>
<tr>
<td>2003-01-06 21:34:09.49...</td>
<td>1.3157636</td>
<td>1.3184711</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>-57.80305</td>
<td>70.42769</td>
</tr>
<tr>
<td>2003-01-06 21:34:10.89...</td>
<td>1.3202947</td>
<td>1.3231139</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
<tr>
<td>2003-01-06 21:34:12.29...</td>
<td>1.3248348</td>
<td>1.3277755</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
<tr>
<td>2003-01-06 21:34:13.69...</td>
<td>1.3294108</td>
<td>1.3324559</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
<tr>
<td>2003-01-06 21:34:15.09...</td>
<td>1.3339859</td>
<td>1.3371552</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
<tr>
<td>2003-01-06 21:34:16.49...</td>
<td>1.338599</td>
<td>1.3418734</td>
<td>4136201</td>
<td>4320000</td>
<td>0</td>
<td>75.70367</td>
<td>88.86735</td>
</tr>
</tbody>
</table>
Statistics Tab

![RSC Trending Tool - Version 2005-01](image)

- **Type First Letters of Mnemonic:**
  - SBRX_B_LS
  - SBRX_A_LS
  - SBRX_A_SS

- **Query Mnemonics:**
  - SBRX_B_LS
  - SBRX_B_SS

- **Query Start Time:**
  - 2003-01-06 21:00:33.08185

- **Query End Time:**

- **List of Statistical Data Sets for Selected Mnemonic:**
  - id 1: start 2003-01-11 03:00:00.0 end 2003-01-11 05:00:00.0 SBRX_B_SS between -1000.0 1000.0 stats 1.0 min
  - id 2: start 2003-01-11 03:00:00.0 end 2003-01-11 06:00:00.0 SBRX_B_SS stats 1.0 min
  - id 3: start 2003-01-11 06:00:00.0 end 2003-01-11 09:00:00.0 SBRX_B_SS stats 1.0 min
Plot Window

BAT_1_VOLTAGE - between 0.0 10.0
BAT_2_VOLTAGE - threshold 0.0

Derived Mnemonics Tab

Enter algebraic expression of existing mnemonics and parameters.

\[ \text{AVG_VOLTAGE} = \frac{1}{2} (\text{BAT\_1\_VOLTAGE} + \text{BAT\_2\_VOLTAGE}) \]

Exponentiation: \( t^x \)
- if \( t \) is not an integer, and \( t < 0 \), then
- \( t^x = -(|t|^x) \)
- also, \( 0^0 = 1 \)

Logical Operators:
- \(< \): 1 if true, 0 if false
- \(\leq \): 1 if true, 0 if false
- \(> \): 1 if true, 0 if false
- \(\geq \): 1 if true, 0 if false
- \(== \): 1 if true, 0 if false
- \(<> \): 1 if true, 0 if false

Trigonometric Functions (in radians):

Clear Edit Fields  Delete  Save
Data and Access Patterns

- No need for a Relational aspect to our database.
- Most mnemonics (both discrete and floating point) change value much less frequently than sampled.
- Using Java’s ZipOutputStream() on Coriolis SOH data:
  - Same data written by Columns compresses (on average) 5 times more than when written by Rows.
- Saving disk space saves time: less disk reads, less network traffic.
- Each DB file (data from one contact) stored as a single ZIP archive composed of a logical file per mnemonic
  - Gives random access to data by mnemonic.
  - Archive file is a “good” size (One physical file per mnemonic creates too many inefficiently small files).
Data Base File Structure

- **Index file:**
  - One file per day.
  - Lists all ingested files that span the given day.
  - Each record includes:
    - Ingest ID
    - Start and end time.
    - SOH type
    - Mnemonic set index.
    - Storage Location (RAID, TAPE, BOTH).
  - ASCII

- **Exactly one DB file per ingest:**
  - Isolates damage from bad data.
  - Directory: `data\IRON\YYYY\MM\DD\SOH_TYPE`
  - File Name: `IRON.YYYY-MM-DD.SOH_TYPE.INGEST_ID.trendDB`
  - Binary
  - Mixed data types (float32, float64, int8, ...)
  - Time saved as seconds offset from top-of-the-hour of start GMT.
  - ZIP Archive of mnemonic logical files.
Space & Time Results - Coriolis

- Operational for 24 months.
- 760 mnemonics collected.
- SSOH: every 4 seconds.
- RTSOH: 10 times/sec for 6 to 8 contacts per day, each about 800 sec.
- Over 50 billion data points stored.
- 13 GB of storage required.
  - Could be improved: Coriolis ingests each subsystem separately – making 17 copies of every Time value (low compression).

Query and Screen Plot for 7 Days of Coriolis Data

<table>
<thead>
<tr>
<th></th>
<th>Points</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1 Mnemonic: no filters</td>
<td>608,784</td>
</tr>
<tr>
<td>B</td>
<td>6 Mnemonics, all same subsystem: no filters</td>
<td>3,652,704</td>
</tr>
<tr>
<td>C</td>
<td>6 Mnemonics, all different subsystems: no filters</td>
<td>3,652,704</td>
</tr>
<tr>
<td>D</td>
<td>6 Mnemonics, 1 point/minute</td>
<td>60,480</td>
</tr>
<tr>
<td>E</td>
<td>6 Mnemonics, min/max/mean over 1 minute interval</td>
<td>60,480</td>
</tr>
<tr>
<td>F</td>
<td>6 Mnemonics, min/max/mean over 1 hour interval</td>
<td>1,008</td>
</tr>
</tbody>
</table>
Hardware

Sun Fire 280 w/ two 900Mhz UltraSparc III processors, 2 GB RAM

- 1 - 7 Query Processes of same or different vehicles
- 1 to 3 Ingest Processes of different vehicles

EtherNet 100 MBits/sec

Fiber: 1 GBits/sec

SUN StoreEdge
660 GB RAID 5 (9 x 73 GB)

440 GB SOH DB
30 GB Stats DB
10 GB Admin
100 GB tmp
73 GB RAID 5

ASCII
GIF

Excel COTS
MatLab COTS

Color Printer

12GB 4mm DDS-3

MUS
EU converted SOH Data

SunRay
...
Maintainability

- All code in 100% Pure Java
- Object Oriented Design
- Full JavaDoc for all classes
- JUnit Automated tests.

- The most common extensions are expected to be to Filters and Statistical Calculations.
  - Each filter is a single method of the filter class. Some efficiency was sacrificed for this modular design.
  - All filters appear in the GUI in a separate, single column Filter Panel. GUI space is reserved for additions to the filter panel.
½ Engineer year: 3 Software Engineers working part-time on this project for two 3 month periods separated by over a year.

<table>
<thead>
<tr>
<th></th>
<th>Lines of Code (including comments)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUI Classes</td>
<td>4178</td>
<td>40%</td>
</tr>
<tr>
<td>Database Classes</td>
<td>5719</td>
<td>30%</td>
</tr>
<tr>
<td>Plot Classes</td>
<td>3821</td>
<td>30%</td>
</tr>
</tbody>
</table>
Dissemination

The RSC Trending Tool software is the property of the United States Air Force.

- Requests for a runtime package and/or source code should be submitted to:
  
  XXXXXXXXXXX
  XXXXXXXXXXX
  Det 12 SMC, RSC
  Kirtland AFB, MN

- Direct technical Questions to:
  
  Joel Castellanos: joel.castellanos@kirtland.af.mil