



A Dynamic Open Source Long Term Archiving and Trending Solution for Small Satellites

Ryan Melton
COSMOS Lead
Ball Aerospace
Boulder, CO

The Problem

- **Binary log files work well for historic analysis over small time periods**
 - Integration and Test
 - Immediate Anomaly Investigation
- **Things break down across large time periods (2+ weeks)**
 - Serially processing through gigabytes or even terabytes of data requires tremendous I/O time before doing any actual data analysis
 - Each query requires another very time consuming pass





Questions You Might Want to Ask Over Large Data Sets

- **What happened with the temperature sensor over the last year?**
 - What was the smallest value?
 - What was the largest?
 - What was the average?
- **How many image captures did the satellite take last week?**
- **Am I missing any data? How long were we operational?**
- **What happened during the anomaly last year?**
- **Has this value ever been that high before?**

The Solution

COSMOS DART (Data Archival and Retrieval)





What is COSMOS DART?

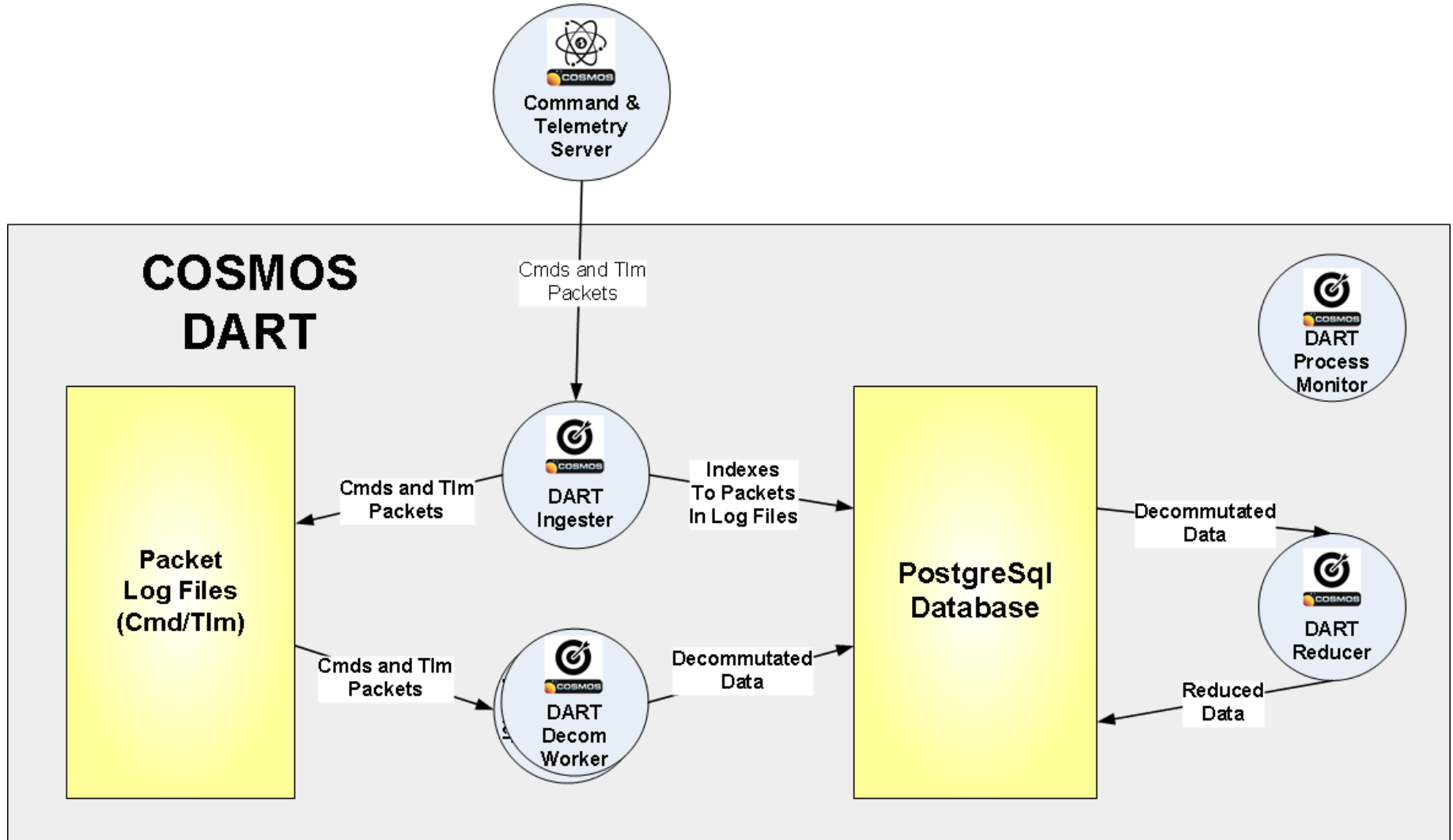
- PostgreSQL backed long term data archival and retrieval system
- Stores both raw packets and decommutated data
- Decommutated data is automatically reduced to minute, hour, and day data sets
- Raw data can be streamed out over a TCP/IP socket
- Decommutated and reduced data accessed over JSON based HTTP interface
- Cross Platform (Windows, Linux, Mac)
- Free, and Open Source



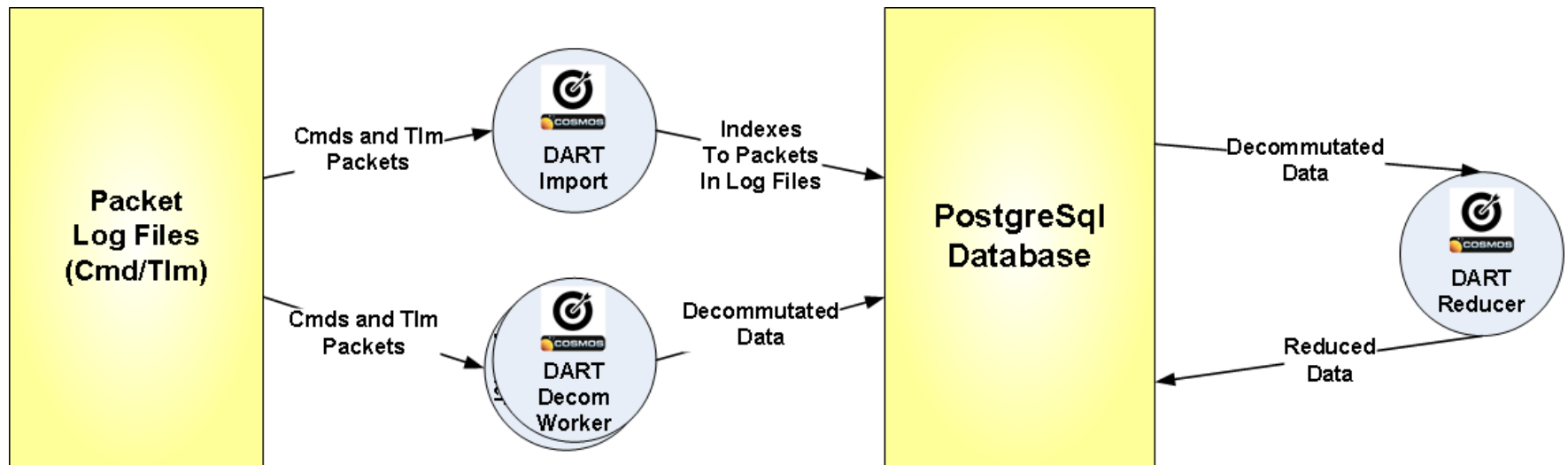
Data Input – Two Methods

- In nominal case, COSMOS CmdTImServer streams all packets into DART over a TCP/IP connection
- To recover from network outages, or for an alternative file based strategy, DART can import data directly from COSMOS files.
 - Note that DART uniquely imports files “in place”

Architecture - Nominal Data Flow

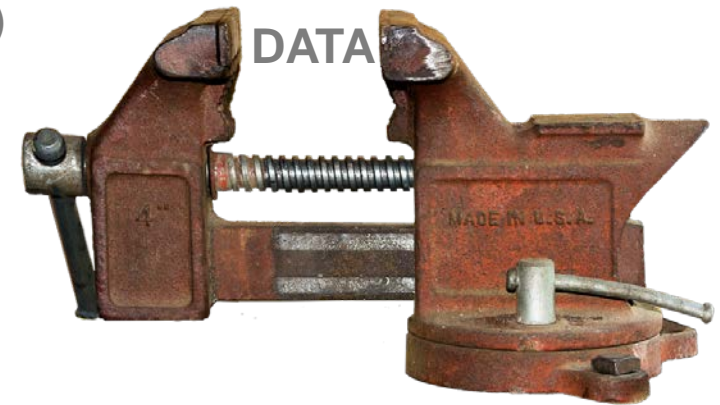


Architecture - File Based Data Flow



Data Reduction

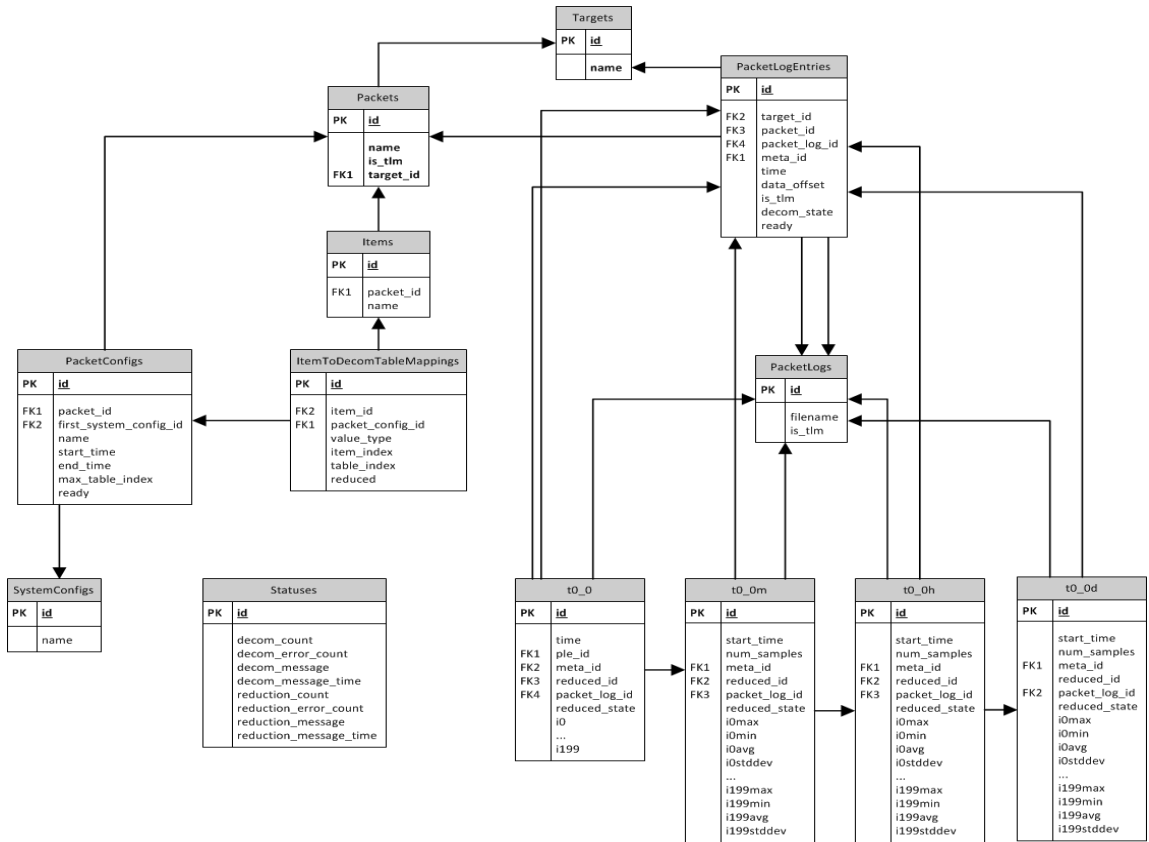
- **DART Automatically reduces decommutated data into minute, hour, and day data sets**
 - Integers and Floats Only
 - Reduced Data Sets Contain Min, Max, Average, and StdDev over time range
 - Supports Fast Summarized Queries
- **Samples**
 - Regular 1Hz (86,400/day, 31,557,600/year)
 - Minute (1440/day, 525,960/year)
 - Hour (24/day, 8766/year)
 - Day (1/day, 365.25/year)





Database Schema

- DART Schema is broken down into 13 table types
- Raw packets are stored in COSMOS Log Files
- Users do not need to interact with SQL



Data Output – Two APIs



- **Raw streaming API** provides raw packets over a specified time range
- **Decommutated/Reduced API** provides a JSON formatted set of data for one telemetry/command data point



Raw Packet Streaming API

- **DART listens on a TCP/IP Socket** by default on port 8777
- **JSON formatted request** received that includes time range, and optionally a limited set of requested packets
- **Requested packets stream out** until all sent or socket closed
- **Simple Example Request** (response is stream of binary packets)

```
{"start_time_sec": 1514764800, "start_time_usec": 0,  
"end_time_sec": 1514809815, "end_time_usec": 0}
```



Decommutated/Reduced API

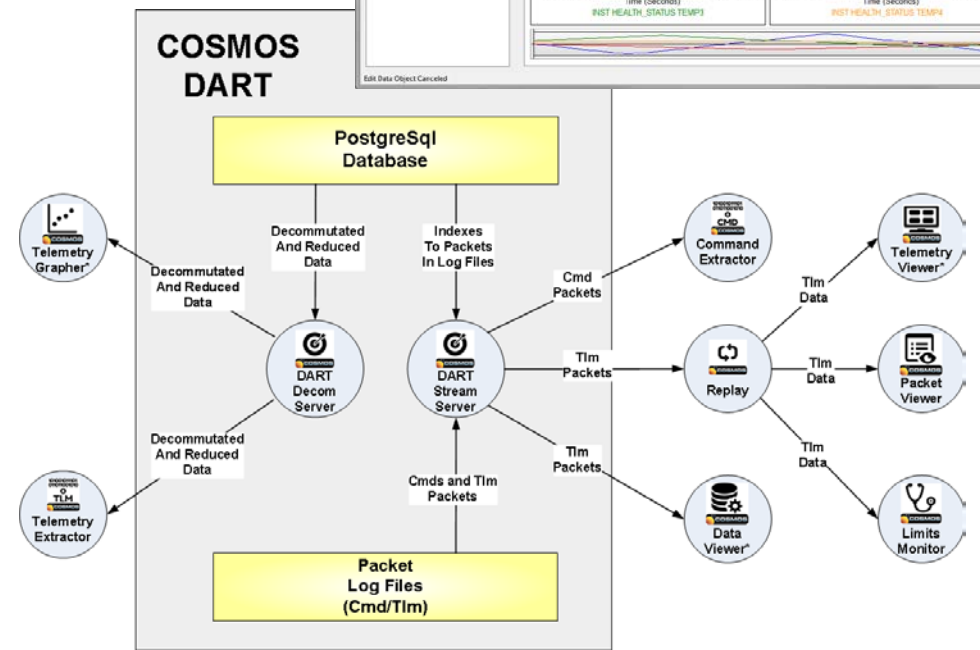
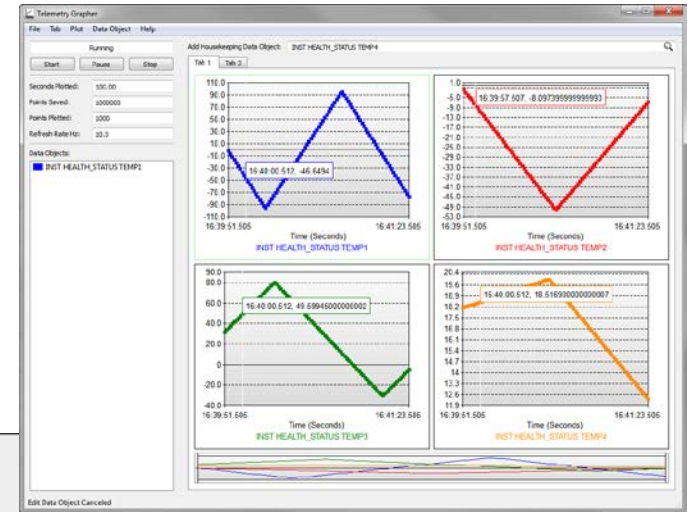
- **DART runs a HTTP server** by default on port 8779
- **JSON-RPC formatted request** received that includes time range, item name, reduction, raw/converted, etc.
- **Requested data returned as a JSON structure** with up to 10,000 data points at a time
- Simple Example Request and Response:

```
→ {"jsonrpc": "2.0", "method": "query", "params":  
  [{"start_time_sec": 1514764800, "start_time_usec": 0,  
    "end_time_sec": 415000000, "end_time_usec": 0, "item":  
    ["INST", "HEALTH_STATUS", "TEMP1"], "reduction":  
    "NONE", "value_type": "CONVERTED"}], "id": 1}
```

```
<-- {"jsonrpc": "2.0", "result": [[10.3, 1514764800, 0,  
1, 1], [15.6, 1514764801, 340, 1, 1]], "id": 1}
```

Integrated Directly with COSMOS Tools

- COSMOS Tools that previously supported log file analysis now also support directly querying DART
 - **Telemetry Grapher**
 - (Decom/Reduced API)
 - **Telemetry Extractor**
 - (Decom/Reduced API)
 - **Command Extractor**
 - (Raw Streaming API)
 - **Data Viewer**
 - (Raw Streaming API)
 - **Replay**
 - (Raw Streaming API)



Other Details



- DART can adapt to changing packet definitions in realtime without having to restart
- DART understands COSMOS SYSTEM META packet allowing queries to filter on meta data such as serial number, mission name, etc, to easily support multiple missions/builds/constellations

Considerations for Production

- **DART expands raw data size by approximately 4x –**
 - Constant overhead per packet so large packets will have lower percent overhead, smaller packets will have higher
- **Plan Backups**
 - Need to backup PostgreSQL database and COSMOS raw log files
- **Running DART on a Separate Server from CmdTImServer is recommended**





SELECTED UPDATES SINCE SMALLSAT 2017

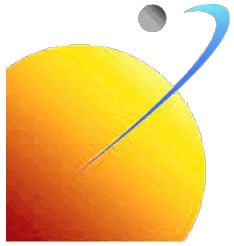
- **8 Releases:** 4.0.1, 4.0.2, 4.0.3, 4.1.0, 4.1.1, 4.2.2, 4.2.3, 4.2.4
- **121 Tickets** Incorporated
- **COSMOS DART** added in 4.2.2
- **COSMOS API changed from a custom TCP protocol to HTTP.** Makes integrating the API from other languages even easier. (Python module is available)
- **Supports Ruby 2.4/2.5**
- **Bug Fixes**
- **COSMOS 4.3 will have built in support for differentiating realtime and stored telemetry**

Questions?



**Installation Instructions at
cosmosrb.com**

**See the Paper for lots of details
on DART implementation**



COSMOS



BALL'S OPEN SOURCE COMMAND AND CONTROL SYSTEM

- **A full C2 system** providing command and telemetry parsing, realtime and offline graphing, test and operations scripting, individual commanding, offline data analysis, command sequence generation, limits monitoring, telemetry screens, and more
- **Downloaded over 83,000 times**
- **Cross Platform** – Windows, Linux, Mac OSX
- **ITAR approved**
- Used for test on over **50 programs** at Ball
- **Flown** as user interface for airborne LIDAR programs
- **Operating satellites** at Blue Canyon and the Naval Post Graduate School (and likely others)
- **In use across the industry** at NASA Goddard, Raytheon, Harris, LASP, Lockheed, SDL USU, Blue Canyon, and many others
- **Game changing technology for universities** providing an affordable solution
- **Reaching outside the industry** (Used to test “Chomp” from Battlebots)
- More info and documentation at **cosmosrb.com**