

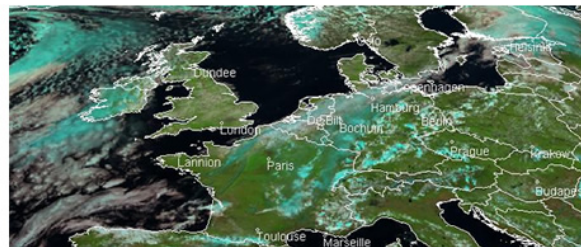
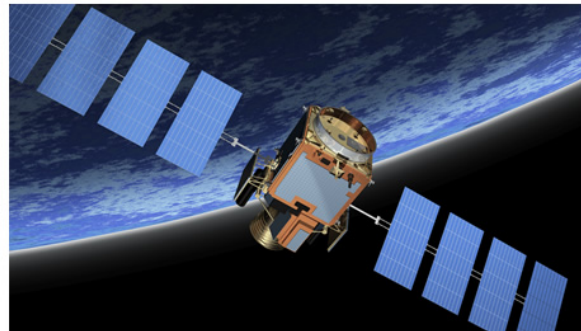
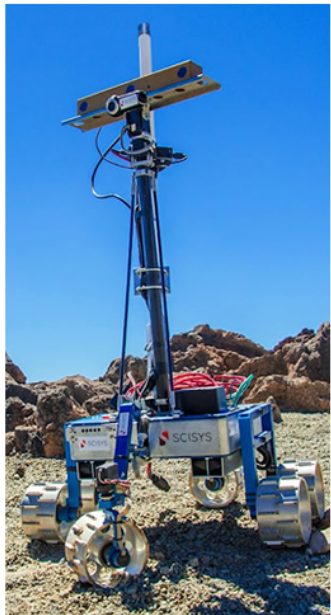


# CFDP Reference and Test Facility (CRTF)

## SAVOIR Onboard Mass Memory Day

Stuart Fowell, Matt Jordan

14<sup>th</sup> February 2014



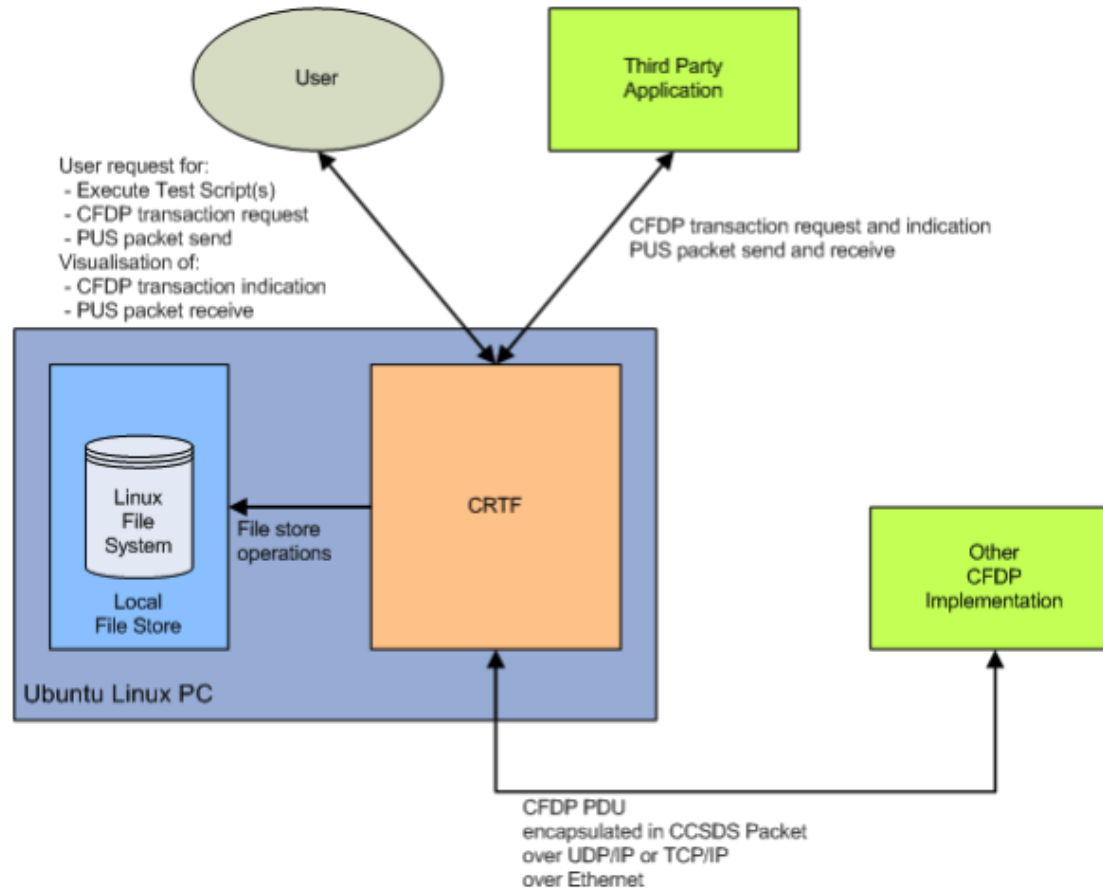
# Overview

- Objectives
- Overview
- Components of the CRTF
- Testing a Third-Party System's CFDP
- Conclusions

# Objectives

- Part of Avionics System Integration and Validation (ASIV) TRP project, primed by TERMA
- Objective is to provide new CFDP implementation suitable as a reference, test and demonstration environment, the CRTF
  - » Replacing old Delphi implementation
- Intended usage of the CRTF is:
  - » Reference CFDP implementation covering class 1 and 2 procedures and all options
  - » Test system against which other implementations can be verified using nominal and off-nominal conditions
  - » Support system used to emulate mission specific configurations
  - » Part of end-to-end test and development environment, connected to ground based implementation at ESOC, used to prepare for future file-based operations
  - » As a standalone Linux based system

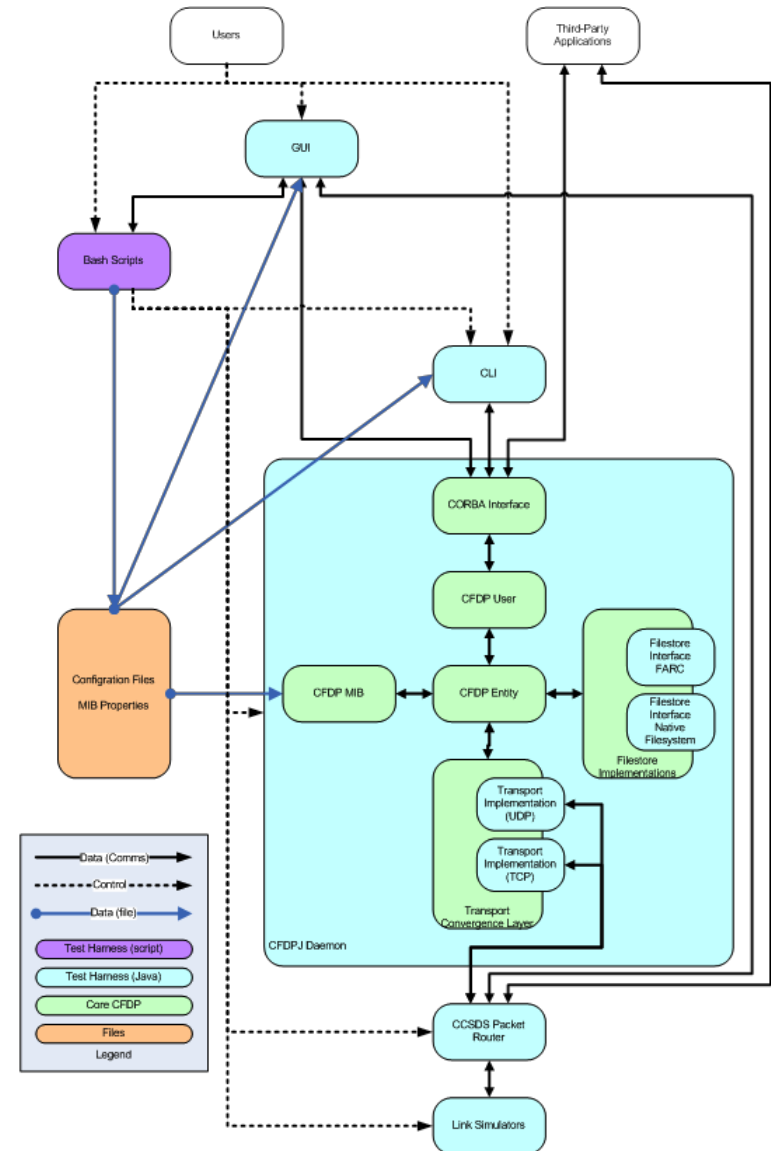
# Overview



- CRTF provides testing of different configurations of CFDP and implementations of CFDP in Third Party Systems, driven by Scripts, Users or Third-Party Applications
- Also supports exchange of PUS Packets, including proposed PUS Services in support of File-Based Operations

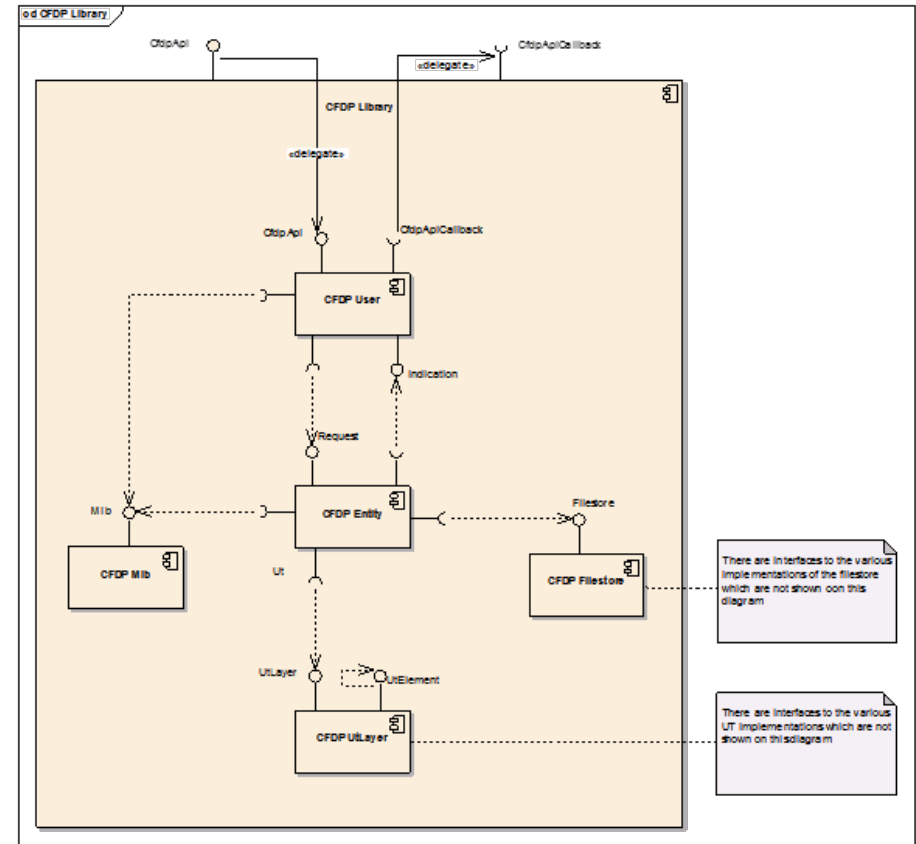
# Components of the CRTF

- CFDP Entity
- Bridge & Link Simulator
- GUIs:
  - » CFDP Test Client GUI
  - » PUS TM/TC Display
- Command Line Interface and CORBA Interfaces to Third-Party Applications
- Test Script Execution Environment
- CFDP Interoperability Test Suite



# CFDP Entity

- ESOC's Java implementation of CFDP
  - » Classes 1, 2, 3 and 4
  - » Store-and-Forward Overlay
  - » Full MIB
  - » Flexible UnitData Transports (UTs) supported
    - › Including TCP and UDP transfer of CFDP PDUs encapsulated in CCSDS Packets
  - » Filestore mapped onto local Linux filestore
    - › FARC also supported
- Updates incorporated
  - » Fixes to support maximum-sized CCSDS Packets (65542 octets including headers)
  - » Fixes to support maximum-supported files (4Gbyte)
  - » Support for Priority on transactions defined in Flow Label parameter
  - » Fix for bug on queuing incoming CFDP PDUs



# Bridge & Link Simulator

- CCSDS Packet Router
  - » Multiplexing/demultiplexing CCSDS Packets onto space link
  - » Routing to applications based on registered APIDs
- Simulation of space links from CCSDS Packet perspective, each configured with
  - » Bandwidth
  - » Latency
  - » Link Availability
  - » Dropped CCSDS Packets rate
  - » Encapsulated CFDP PDU manipulation (based on configured APID)
    - › Dropped, delayed (hence re-ordering), corrupted, or duplicated
- Bridging to Third-Party systems
  - » Exchanging CCSDS Packets over TCP or UDP
  - » Using configured TCP or UDP port number and IP address
    - › TCP or UDP matching selected CFDP UT
- Configured using a MIB

# CFDP Test Client GUI

The screenshot displays the Eclipse-based GUI for the CFDP Test Client. The main window is titled 'CFDP' and contains two primary panels: 'Transfers' and 'Log'.

**Transfers Panel:** A table showing the status of file transfers.

Txn. ID	Dest. ID	Source File	Dest. File	Progress	Status	Condition Code	Start Time	Completion Time
01_00	07	T03101-01	T03101-01	17 B	COMPLETED	NO_ERROR	2013-12-05 14:37:38.4	2013-12-05 14:37:38.6

**Log Panel:** A log viewer showing system events. The log is filtered by 'Indications' and 'ALL'.

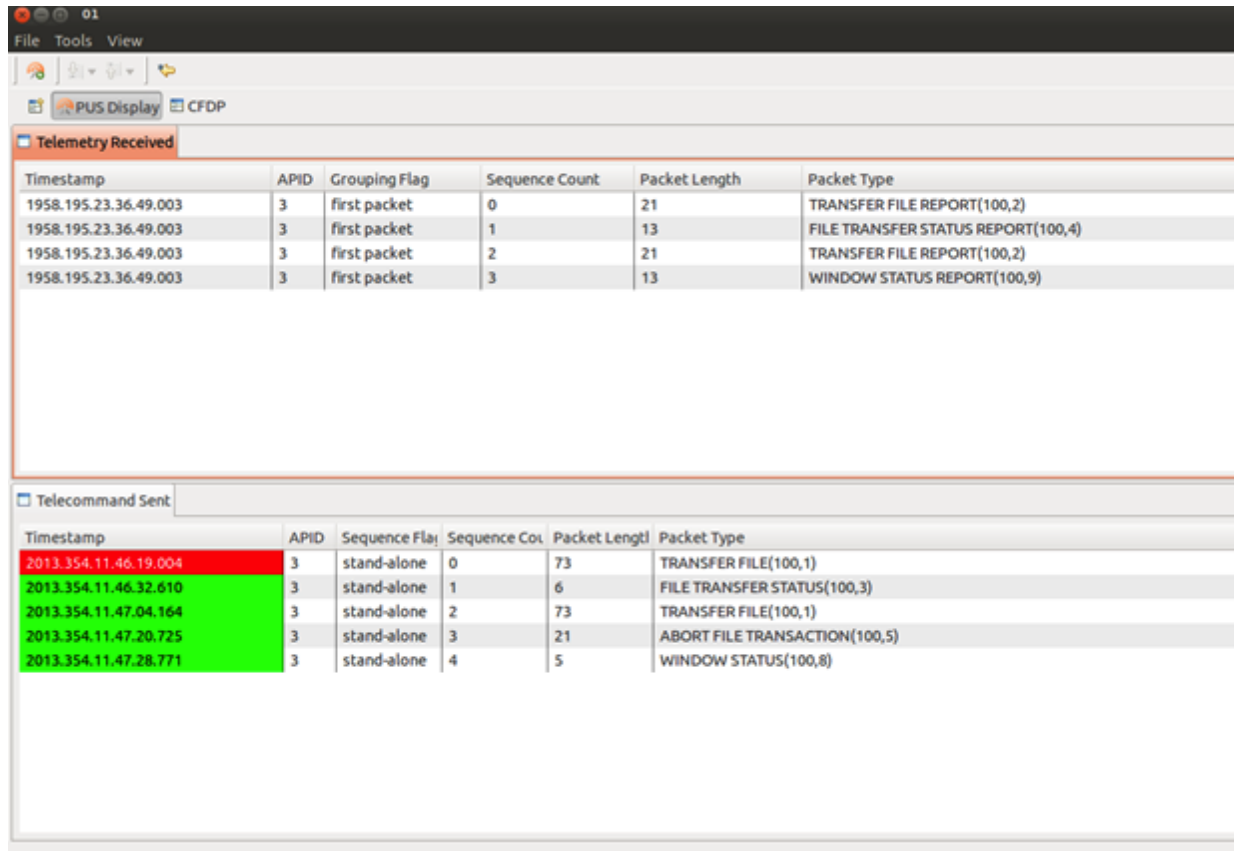
Severity	Timestamp	Source	Text
INFO	2013-12-05 14:37:38.661	esa.egos.cfdp.entity.ProactiveTransactionThread.run	01_00 Transaction shutdown, Condition Code NO_ERROR(INCOMPLETE)
INDICATION	2013-12-05 14:37:38.648	01_00	01_00 TransactionFinishedIndication [ConditionCode=NO_ERROR, DeliveryCode=COMPLETE, FileStatus=...
INDICATION	2013-12-05 14:37:38.338	01_00	01_00 TransactionIndication
INFO	2013-12-05 14:37:38.325	esa.egos.cfdp.entity.Entity.processRequest	01_00 Processing: PutRequest[Destination 07 Mode default SourceFile T03101-01 DestFile T03101-01 5
INFO	2013-12-05 14:37:38.325	esa.egos.cfdp.entity.Transaction.start	01_00 Starting transaction
INFO	2013-12-05 14:37:38.108	esa.egos.cfdp.daemon.CfdpClientApiImpl.connect	User test (Role Administrator) logged in to session [1] from host crtf.

**Filestore Panel:** A file browser showing the local filestore structure. The root is 'Entity 01 filestore/' and it contains a file named 'T03101-01'.

- Eclipse-based GUI to local CFDP Entity
- Initiate, visualise and log status of CFDP Transactions
- Visualisation of CFDP Local Filestore
- Configuration Editor for CFDP Entity and Link Simulator MIB



# PUS TM/TC Display



The screenshot shows the PUS Display GUI with two main sections: 'Telemetry Received' and 'Telecommand Sent'. The 'Telemetry Received' section contains a table with 6 columns: Timestamp, APID, Grouping Flag, Sequence Count, Packet Length, and Packet Type. The 'Telecommand Sent' section contains a table with 6 columns: Timestamp, APID, Sequence Flag, Sequence Count, Packet Length, and Packet Type. The first row of the 'Telecommand Sent' table is highlighted in red, and the next four rows are highlighted in green.

Timestamp	APID	Grouping Flag	Sequence Count	Packet Length	Packet Type
1958.195.23.36.49.003	3	first packet	0	21	TRANSFER FILE REPORT(100,2)
1958.195.23.36.49.003	3	first packet	1	13	FILE TRANSFER STATUS REPORT(100,4)
1958.195.23.36.49.003	3	first packet	2	21	TRANSFER FILE REPORT(100,2)
1958.195.23.36.49.003	3	first packet	3	13	WINDOW STATUS REPORT(100,9)

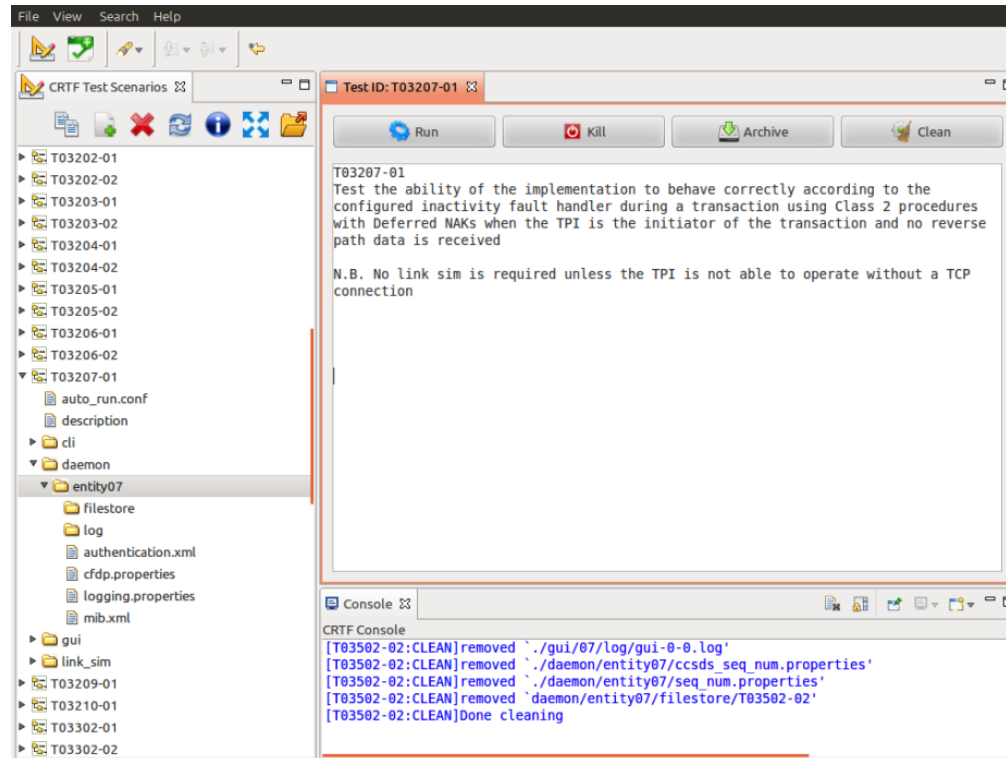
Timestamp	APID	Sequence Flag	Sequence Count	Packet Length	Packet Type
2013.354.11.46.19.004	3	stand-alone	0	73	TRANSFER FILE(100,1)
2013.354.11.46.32.610	3	stand-alone	1	6	FILE TRANSFER STATUS(100,3)
2013.354.11.47.04.164	3	stand-alone	2	73	TRANSFER FILE(100,1)
2013.354.11.47.20.725	3	stand-alone	3	21	ABORT FILE TRANSACTION(100,5)
2013.354.11.47.28.771	3	stand-alone	4	5	WINDOW STATUS(100,8)

- Eclipse-based GUI for sending PUS TCs and receiving TMs
  - » Sending of (proposed) FBO file transfer and onboard filestore management, and user defined raw PUS telecommands
  - » Visualisation of received (proposed) FBO file transfer and onboard filestore management, and raw PUS telemetry packets
- Multiplexed onto simulated space links by CCSDS Packet Router

# Command Line Interface and CORBA Interfaces

- CORBA Interfaces
  - » CFDP Entity CORBA Interface
    - › Allows user applications to instigate and control CFDP transactions for local CFDP Entity
    - › and read and modify local CFDP Entity's MIB
  - » PUS CORBA Interface
    - › Allows user applications to send PUS TCs and receive PUS TM packets
    - › Multiplexed onto simulated space links by CCSDS Packet Router
- Command Line Interface (CLI) to initiate CFDP transactions with local CFDP Entity
  - » Wrapper to CFDP Entity CORBA interface
  - » Allows testing to be scripted and CFDP Entity to be monitored

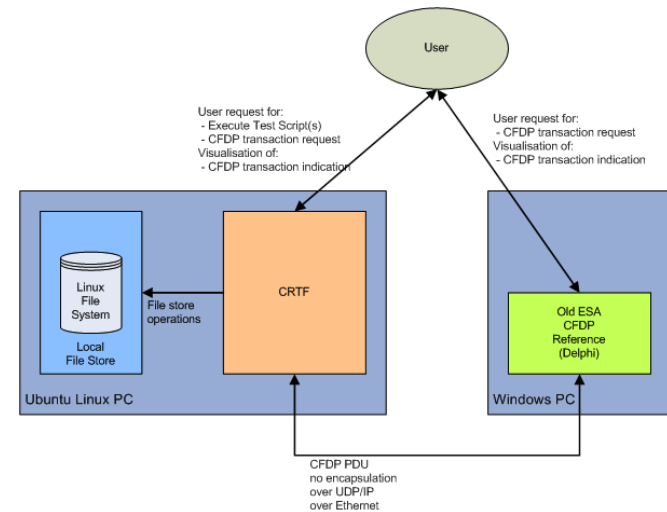
# Test Script Execution Environment



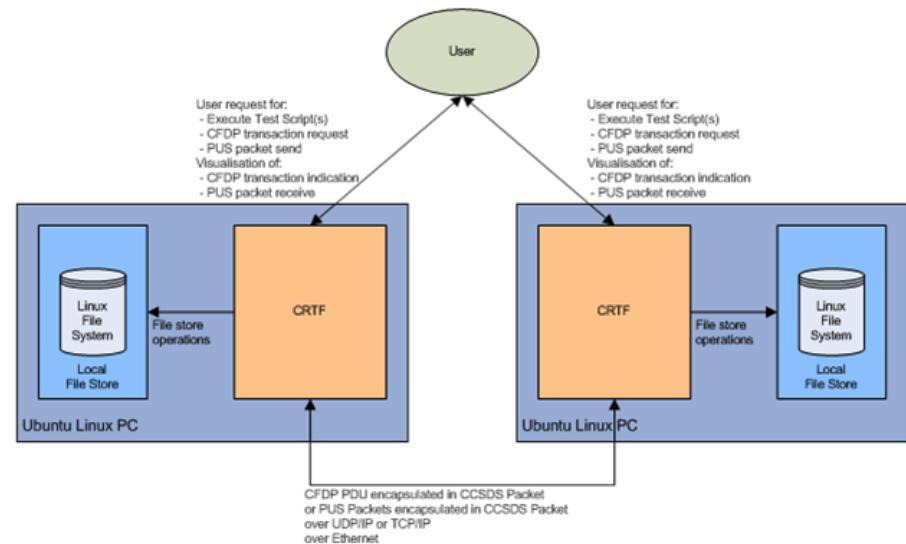
- Facility to (partially-) automate initiating and monitoring of CFDP transactions
- Can be used for verification of Third Party System's CFDP implementations using nominal and off-nominal conditions
- Based on Linux bash scripts using the CFDP CORBA API together with manipulation of the local CFDP Entity and Link Simulator MIBs
- Editor includes short-cut for creating new test scripts and provided together with a basic editor.

# CFDP Interoperability Test Suite

- CRTF comes with test scripts for CFDP Interoperability Tests
  - » “Notebook Of Common Inter-Agency Tests for Core Procedures”, CCSDS 720.4-Y-1, Issue 1, September 2007
- Requires steps to manipulate Third-Party System’s CFDP implementation
  - » Hence need adapting for testing interoperability of each Third-Party System’s CFDP implementation
- Tailor to selected set of options for Third-Party System’s CFDP implementation



Testing against Old Delphi Implementation



Testing against second copy of CRTF

# Testing Third-Party System's CFDP

- Bridge to Third-Party System's CFDP Implementation:
  1. Configure CFDP Entity MIB to include Third-Party System's CFDP Implementation
  2. Configure CFDP Entity MIB to select TCP or UDP UT
  3. Bridge Link Simulators to Third-Party System to exchange CCSDS Packets
  4. Configure Link Simulator MIB to desired bandwidth, latency, link availability and dropped packets rate of the simulated space links in each direction
- Three approaches to testing:
  1. Use Test Client and PUS TM/TC Display GUIs for ad hoc Testing
  2. Bridge to Third-Party Applications for integration with e.g. MCS
  3. Perform CFDP Interoperability Testing
    - › Create copy and adapt CFDP Interoperability test scripts

# Conclusions

- CRTF provides a flexible CFDP reference and test facility to facilitate:
  - » investigation of CFDP characteristics and performance for mission specific configurations
  - » testing of support for CFDP by other implementations using nominal and off-nominal conditions
    - › Development test tool for implementers
    - › Validation and acceptance testing