



CFDP Reference and Test Facility (CRTF)

Final Presentation

Stuart Fowell, Bruno Carvalho

10th December 2014



Overview

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

Consortium

- Avionics System Integration and Validation (ASIV) TRP project
 - » TERMA – Prime
 - » SCISYS – Subcontractor

- 3 main tasks:
 - » CANBus Reference and Test Facility (CBRTF)
 - » OBC to RTU Protocol Analysis
 - » **CFDP Reference and Test Facility (CRTF) – the subject of this Final Presentation**

User Requirements

- Future ESA missions planned to make use of file-based operations
 - » Baseline for this is to use CFDP as the protocol to transfer files to/from spacecraft,
 - » ... alongside the continuing use of PUS packets for spacecraft monitoring and control
 - » Uplink of files containing sets of commands, configuration data and software patches,
 - » together with down-linking of sets of telemetry are all planned.
- CFDP is also planned to be used for downlink of science data
 - » Files being a better packaging mechanism for atomic transfer of large data sets
 - » CFDP providing retransmission across noisy space links, e.g. the Ka-band for Euclid mission.
- In support of this, on-going research studies have been looking into appropriate configurations of CFDP for different mission scenarios and how best to make use of it
 - » Mission Control Centre, spacecraft and possibly Ground Stations will all need CFDP deployments (known as CFDP entities)
 - » These different environments will necessarily require different CFDP implementations
 - › Embedded implementation within a spacecraft's Mass Memory may consist of hardware and embedded software,
 - › Implementation integrated with SCOS-2000 at ESOC is better suited to implementation in a portable, standard programming language such as Java.
- There is a resulting need for a reference CFDP implementation
 - » To be used to test these different deployed CFDP entities during development and spacecraft AIV
 - » Different purpose to validation of software or hardware, instead testing:
 - › implementation of the protocol
 - › interoperability against other CFDP implementations
 - » to ensure the different components of a mission will integrate and interoperate
 - » It may well be, of course, that some components are provided by external agencies
 - › Necessitating inter-agency interoperability
 - › Strength of CFDP being an international standard
 - › Still the requirement to ensure conformance to the standard.
- Therefore ESA require the CFDP Reference and Test Facility (CRTF)
 - » Configurable to the CFDP options selected and configuration used for particular mission scenarios

User Requirements

- Research studies have already concluded that the following CFDP classes are appropriate:
 - » Class 1 (direct unreliable)
 - » Class 2 (direct reliable), though different reliability options are available (e.g. immediate or deferred NAK)
- Need for relaying by CFDP (Classes 3 and 4 or Store-and-Forward Overlay) is less proven as fewer missions require relaying (primarily planetary landers)
- Therefore CRTF should be configurable to support
 - » all the CFDP options associated with Classes 1 and 2
 - » together with the different user functions such as “message to user”
 - › “Message to user” can be used to contain PUS packets
 - › thus allowing CFDP to provide reliable transfer of PUS packets (or relayed if classes 3 and 4 or Store-and-Forward overlay are used)
- CCSDS has defined CFDP interoperability test plan
 - » Originally used to validate the protocol specification
 - » Suitable to test conformance of an implementation against the specification
 - » Therefore CRTF Test Suite should implement these tests.
- Each test requires configuration of source and destination CFDP entities before initiating the test
 - » Set up CFDP entities’ Management Information Bases (MIBs)
 - » Initiate CFDP operations from either source or destination
 - » Adaptable per CFDP entity under test

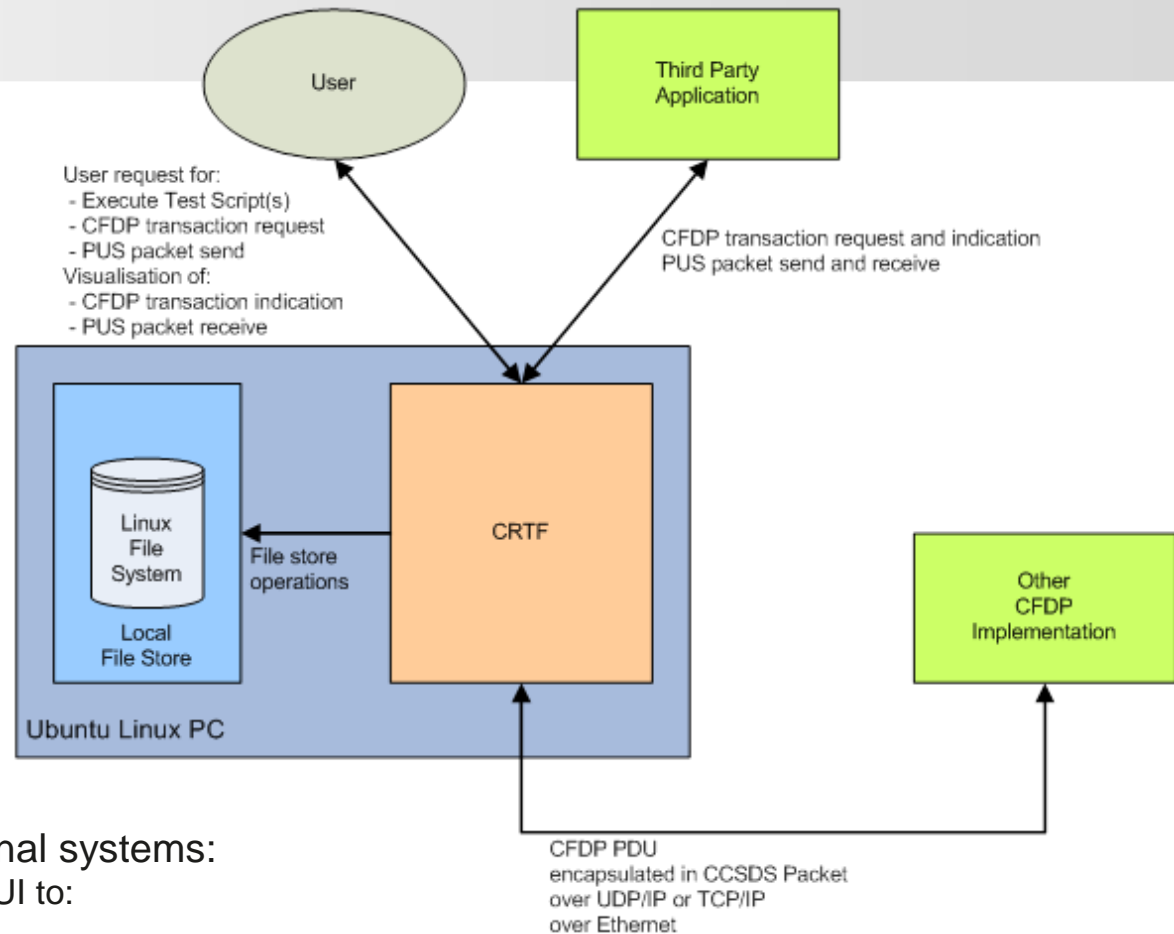
Objectives

- 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

CFDP Implementation

- Originally CFDP was developed and the standard validated using a Delphi implementation
 - » Old development environment no longer support
- Aa CFDP Java implementation (developed to category D) exists
 - » IPR is owned by ESA
- Implements the whole CFDP specification
 - » including all options except unbounded files
- Meets the core objectives of providing a reference implementation
 - » Indeed likely to provide CFDP implementation used by ESOC in future missions where file-based operations are required
 - » Has extensive, Linux bash script-based test suite
 - » Already tested against the (old) Delphi-based ESA Reference
 - › Indeed some 14 bugs were found in the reference!
 - » Has/is being used in a number of further projects
 - › thus providing additional debugging through multiple usage
- Meets all requirements except for the TCP connectivity in addition to UDP and the simulation of delays
 - » Both were added in the project.
- Therefore Java implementation and test suite selected for re-use

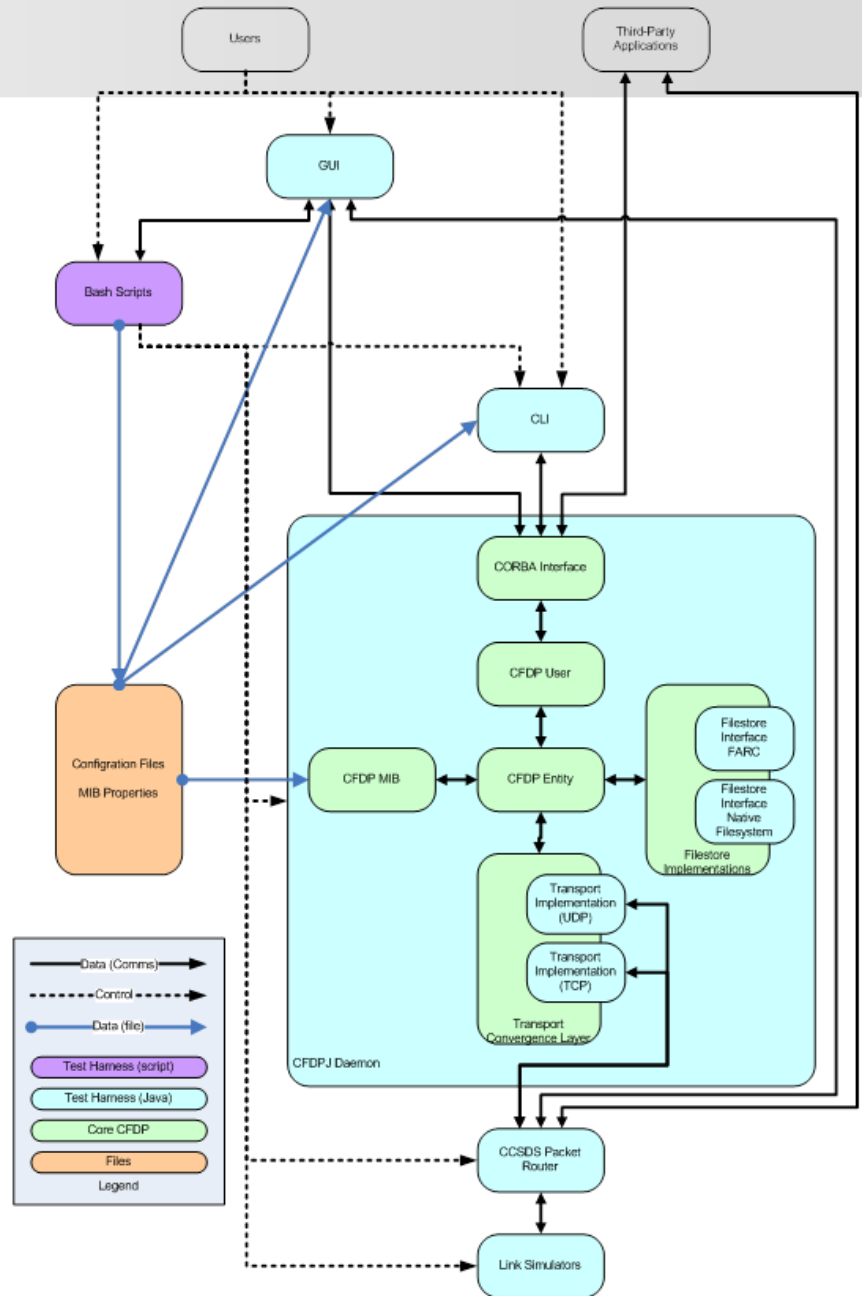
Overview



- Following exchanges with external systems:
 - » User interactions with CRTF GUI to:
 - › Execute test script(s)
 - › Initiate and monitor CFDP supported File transfer and remote file store operation requests
 - › Send and receive PUS packets, including file transfer and filestore management requests and responses
 - » Third party application exchanges with the CRTF:
 - › CFDP supported File transfer and remote file store operation requests
 - › PUS packet transmission and reception
 - » CRTF exchanges with other CFDP implementations:
 - › Interoperability of CFDP class 1 and class 2 procedures, including all associated operations
 - › Interoperability of CFDP message to user capability

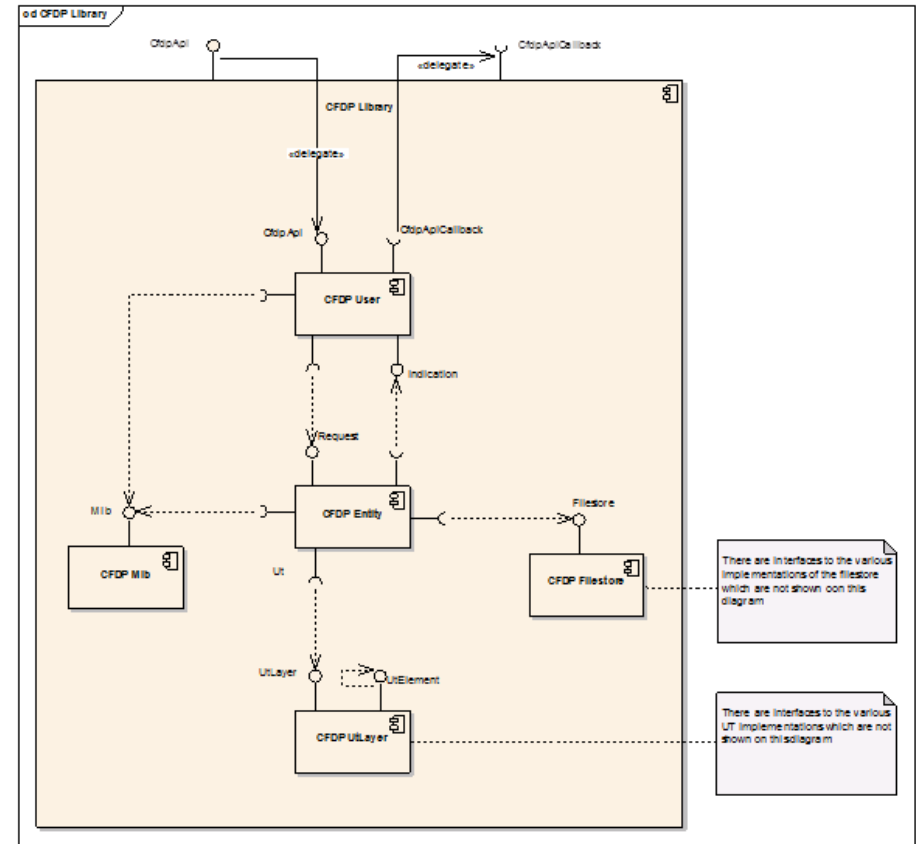
Components of the CRTF

- CFDP Java Implementation
- 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
- Example test CRTF Configurations
 - » (old) Delphi-based ESA Reference
 - » Second CRTF
- Installed on Ubuntu Linux PC delivered to ESA



CFDP Java Implementation

- 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 (new functionality)
 - » Multiplexing/demultiplexing CCSDS Packets onto space link
 - » Routing to applications based on registered APIDs
- Simulation of space links from CCSDS Packet perspective (new functionality)
- 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 (former new functionality)
 - » 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 CFDP Test Client GUI. The main window is titled '01' and contains two primary panels: 'Transfers' and 'Log'.

Transfers Panel:

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

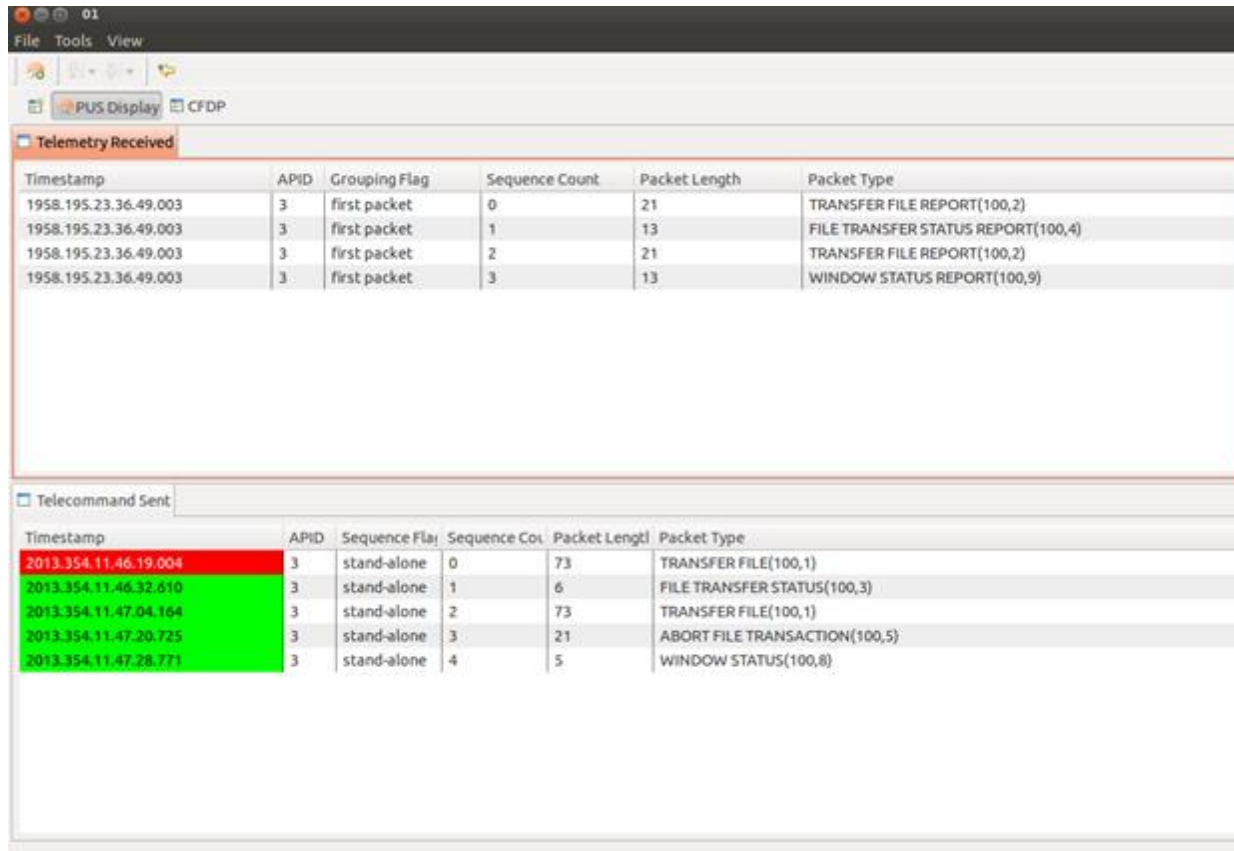
Log Panel:

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 S...
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.

The right-hand side of the GUI features a 'Filestore Browser' panel showing a tree view of the filestore structure, including 'Entity 01 filestore/' and 'T03101-01'.

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

PUS TM/TC Display



The screenshot shows the PUS Display GUI with two tables. The 'Telemetry Received' table has 6 columns: Timestamp, APID, Grouping Flag, Sequence Count, Packet Length, and Packet Type. The 'Telecommand Sent' table has 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)

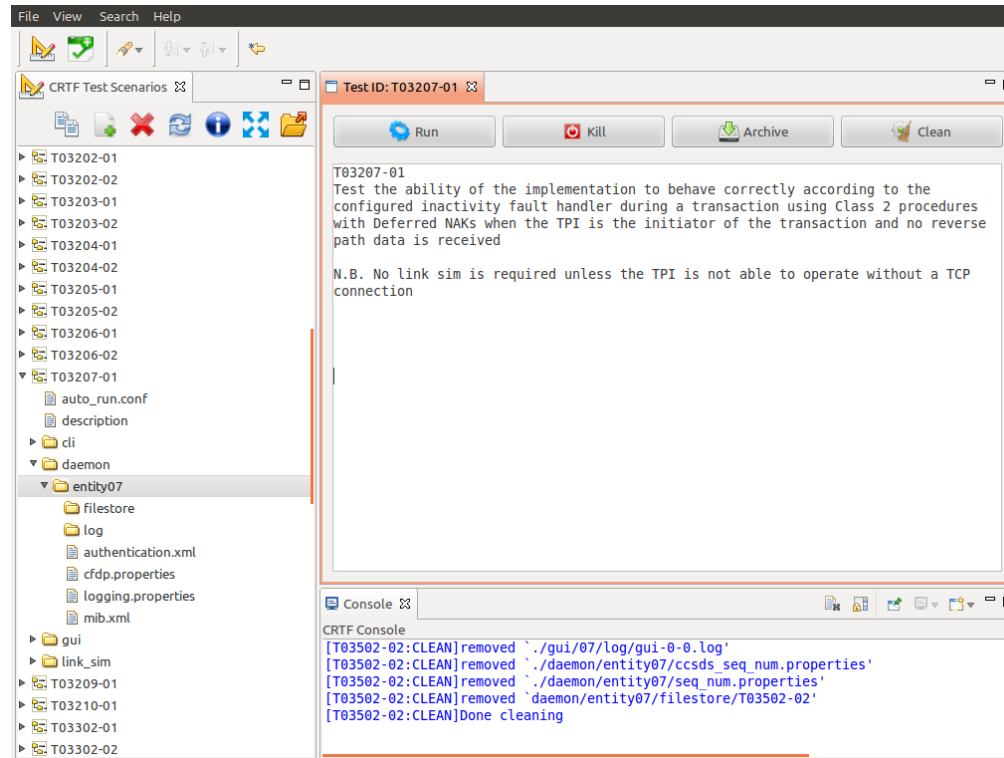
Timestamp	APID	Sequence Flag	Sequence Count	Packet Length	Packet Type
2013.354.11.46.19.004	3	stand-alone	0	73	TRANSFER FILE(100,1)
2013.354.11.46.32.610	3	stand-alone	1	6	FILE TRANSFER STATUS(100,3)
2013.354.11.47.04.164	3	stand-alone	2	73	TRANSFER FILE(100,1)
2013.354.11.47.20.725	3	stand-alone	3	21	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 (new functionality)
 - » 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 (new functionality)
 - › 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

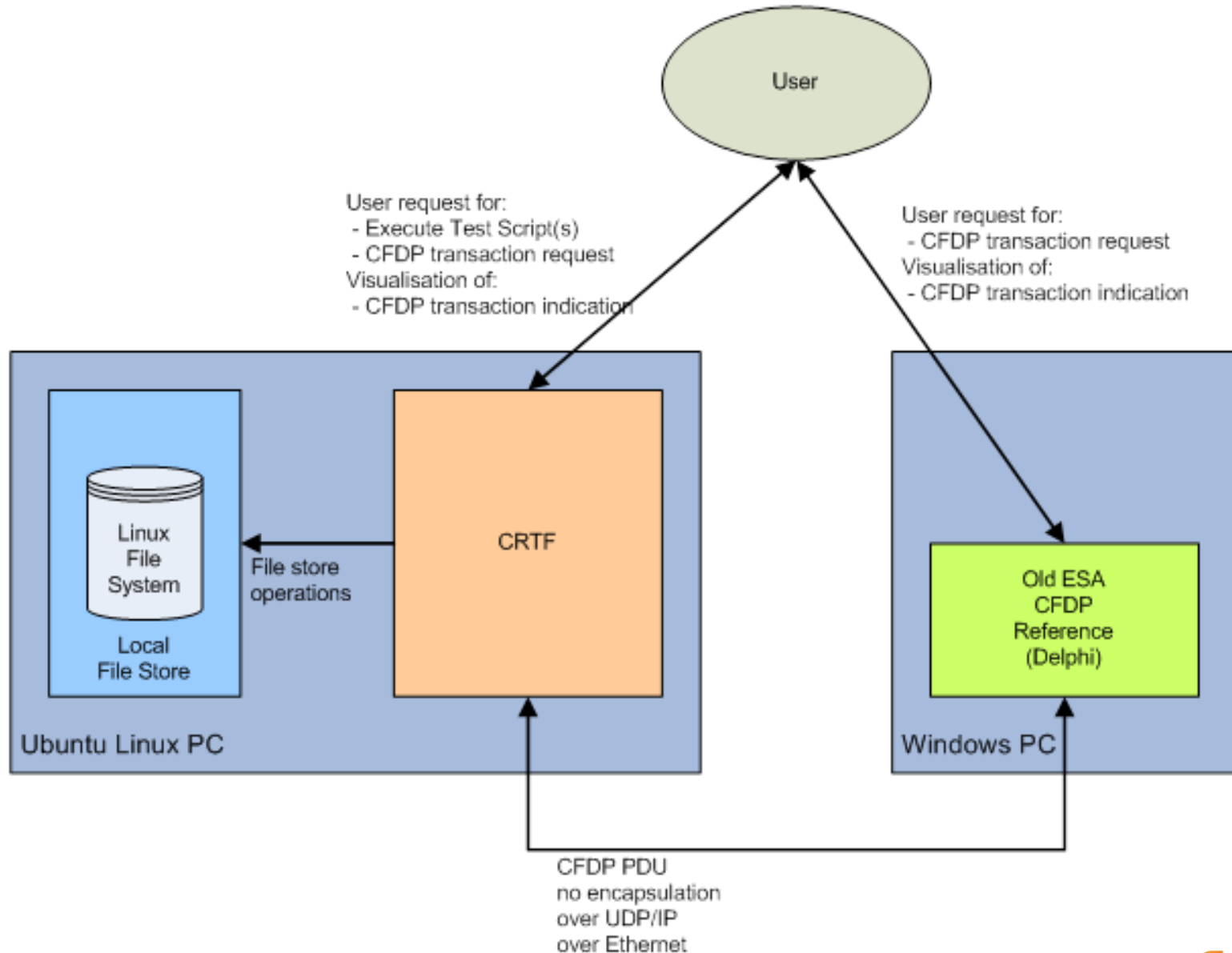


- Facility to (partially-) automate initiating and monitoring of CFDP transactions (improved functionality)
- Can be used for verification of Third Party System's CFDP implementations using nominal and off-nominal conditions
- Based on Linux bash scripts using the CFDP CORBA API together with manipulation 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 (new functionality)

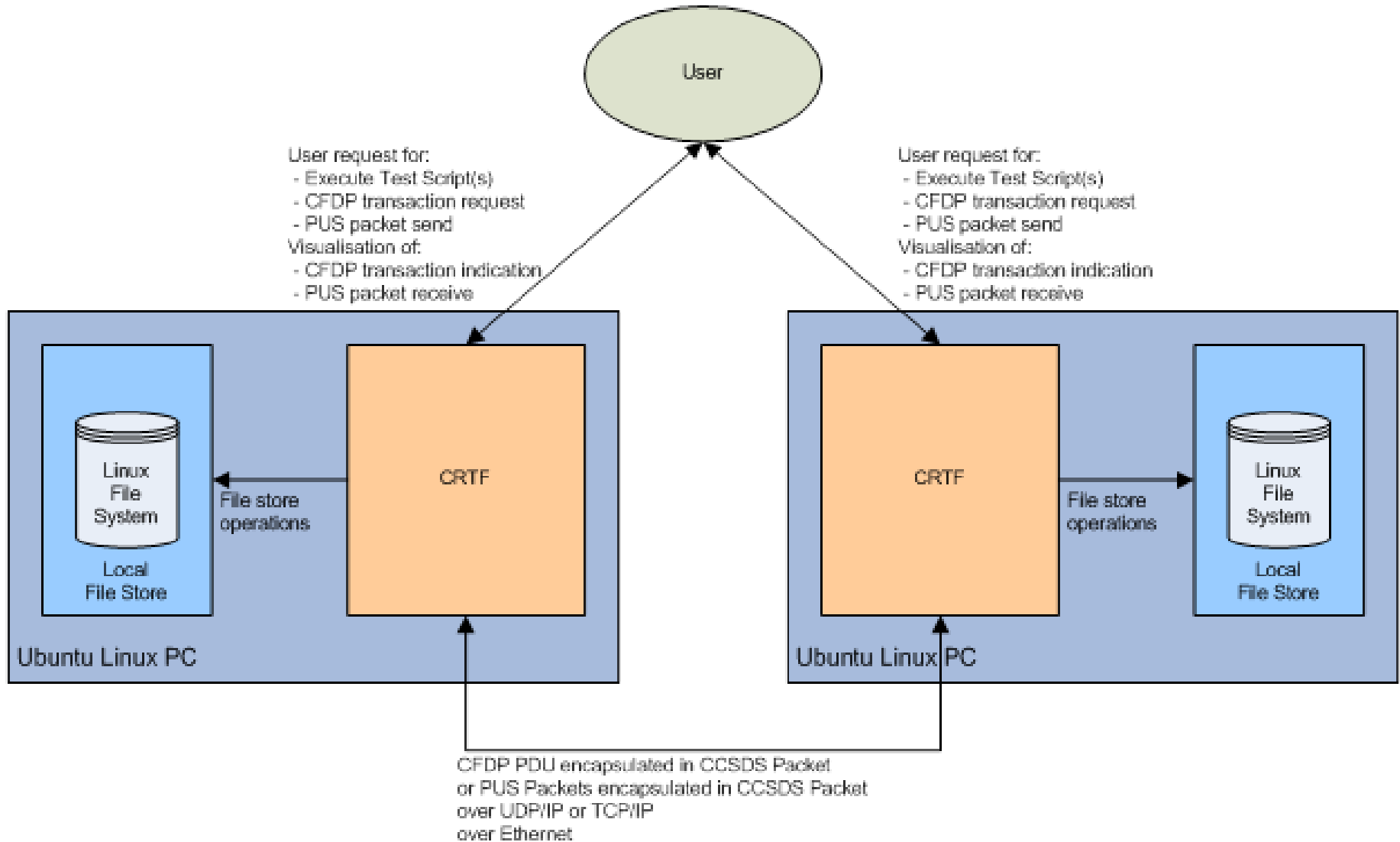
Steps to test Third-Party System's CFDP

- 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

Tested against (old) Delphi ESA Reference



Tested against second CRTF



CFDP Interoperability Test Suite

- CRTF comes with test scripts for CFDP Interoperability Tests (new functionality)
 - » “Notebook Of Common Inter-Agency Tests for Core Procedures”, CCSDS 720.4-Y-1, Issue 1, September 2007
 - › Series 1: Confidence Baseline
 - › Series 2: Acknowledged Mode
 - › Series 3: Reserved MTU Functionality
 - › Series 4: Other Functionality
 - › Series 5: Proxy Operations
- These scripts typically require steps to manipulate Third-Party System’s CFDP implementation
 - » Hence there is a need to adapt for testing interoperability of each Third-Party System’s CFDP implementation
- Also need to tailor which tests are executed based upon selected set of options provided by Third-Party System’s CFDP implementation

Complementary activities

- CFDP undergoing 5 year revision at CCSDS
- Small number of revisions proposed, e.g.
 - » Extension to maximum file size and segment size
 - » Deletion of classes 3 and 4 (Store and Forward Overlay retained)
- SCISYS has supported ESA in prototyping these revisions through updates to the CRTF

Possible follow-up activities

- ESA's RASTA Test Facility at ESTEC
 - » Set of flight-equivalent avionic hardware and software components
 - » Allows results of research studies to be installed at ESTEC without the additional cost of extra, often duplicated hardware
 - » For example, two different implementations of CFDP (from SpaceBel and SCISYS) have been accepted running on ESA's RASTA Test Facility and integrated with its Mass Memory solution
- CRTF is required in future to be integrated with the RASTA system so as to allow a standard reference for
 - » testing CFDP embedded software implementations running on RASTA
 - » integration with other CFDP implementations e.g. stand-alone Mass Memories supporting CFDP
 - » Test system against which other implementations can be verified using nominal and off-nominal conditions

Conclusions

- CRTF successfully developed, installed at ESTEC and accepted
 - » Linux laptop installed with
 - › CRTF Tool, source code and documentation
 - › Interoperability test scripts
 - › There is a final report being put together by TERMA, which include our contribution on CRTF
- CRTF now provides a flexible CFDP reference and test facility to enable:
 - » investigation of CFDP characteristics and performance for mission specific configurations
 - » End-to-end testing of third party CFDP implementations using nominal and off-nominal conditions
 - › Development test tool for implementers
 - › Validation and acceptance testing

File Store Browser Display (SpaceZilla)

The screenshot displays the File Browser Display interface with two main panels: Ground File Store and Onboard File Store. The Ground File Store shows a directory structure under FARC:// with files like config, config_global, onboard, file.500k, and file.10k. The Onboard File Store shows a directory structure under SSMM://RIO/ with files like Tom, dir.list, file.500k, and filea.500k. A yellow callout points to the Onboard File Store Model. Below these panels is the Manual Requests Stack and File Transactions sections.

Ground File Store

Filename	Filesize	Filetype	Last Modified	State
config	0	DIR	1970.001.00.00.00.000	CURRENT
config_global	0	DIR	1970.001.00.00.00.000	CURRENT
onboard	0	DIR	1970.001.00.00.00.000	CURRENT
file.500k	0	FILE	1970.001.00.00.00.000	CURRENT
file.10k	0	FILE	1970.001.00.00.00.000	CURRENT

Onboard File Store

Filename	Filesize	Filetype	Last Modified	State
Tom	0	FILE	1970.001.00.00.00.000	CURRENT
dir.list	0	FILE	1970.001.00.00.00.000	CURRENT
file.500k	0	FILE	1970.001.00.00.00.000	CURRENT
filea.500k	0	FILE	1970.001.00.00.00.000	ASSUMED

Manual Requests Stack

Request Type	File Store	Src Rep Path	Src File Name	Dest Rep Path	Dest File Name	Priority	Execution Time
DELETE_FILE_REQUEST	ONBOARD	-	-	/RIO/	Tom	MEDIUM	IMMEDIATE

File Transactions

Trans ID	Dest ID	Request Type	State	Source File	Destination File	Execution Time	Priority	Start Time
FBD_3	2	SEND_FILE	SUCCESSFUL	/file.500k	/RIO/file.500k	IMMEDIATE	HIGH	2014.324.15.29.21.00
FBD_4	2	GET_FILE	SUCCESSFUL	/file.10k	/file.10k	IMMEDIATE	HIGH	2014.324.15.50.36.00
FBD_4	2	1	SUCCESSFUL	/file.10k	/file.10k	IMMEDIATE	HIGH	2014.324.15.50.36.00
FBD_5	2	SEND_FILE	ACTIVE	/file.500k	/RIO/filea.500k	IMMEDIATE	HIGH	2014.324.15.54.30.00

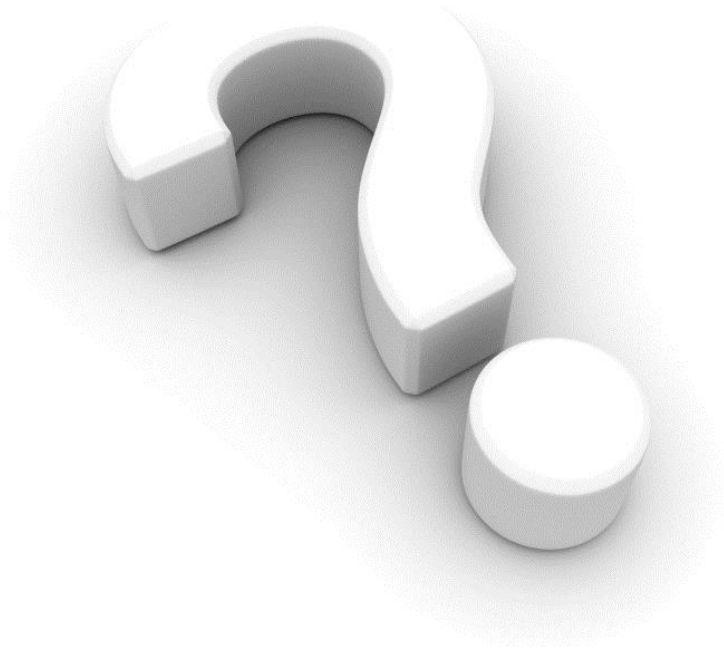
MCS Window: Tx Unknown Rx Unknown **S/C Window:** Tx Unknown Rx Unknown

Ground File Store

Onboard File Store Model

Transaction history and progress

Thank you for your attention



	Bruno Carvalho Business Development Manager Space Division
	SCISYS UK Ltd Clothier Road, Bristol BS4 5SS, UK Direct: +44 1249 466 337 Mobile: +44 7584 349174 bruno.carvalho@scisys.co.uk www.scisys.co.uk