

DS-IRF-TP-0001  
Date: 1995 November 23

Issue: 2  
Rev. : 2  
Page: i

# CSDS User Interface ISDAT Software Verification and Validation Plan

Swedish Institute of Space Physics, Uppsala Division  
S-755 91 Uppsala, Sweden

Change bars show changes introduced in issue 2, revisions 0, 1 and 2

Document Status Sheet			
1. Document Title: <b>CSDS-UI ISDAT Test Plan</b>			
2. Document Reference Number: <b>DS-IRF-TP-0001</b>			
3. Issue	4. Revision	5. Date	6. Reason for Change
Draft	0	94 Oct 28	New document.
0	1	94 Dec 13	Referring to document [Ref. 3] for master schedule dead-lines in section 2.2. Expanded the test overview in section 2.5. Replaced all acronyms CUI by CSDS UI. Included the CSDS module test file in Appendix ???. Included the <i>cuitm</i> unit test procedure in Appendix ???. Included the <i>cuigr</i> unit test procedure in Appendix ???. Included the <i>cuimeta</i> unit test procedure in Appendix ???. Included the <i>cuiestat</i> unit test procedure in Appendix ???. Modified the generic test report in section 4.2.3. Specified test file directories and modified list in section 4.3.1. Moved the reference list from a section to a sub-section. Referring also to the user manuals in section 5.1. Section 5.3 has been completely re-written (there are no change bars for the body of the chapter). Most of section 6.4 has been removed and reference is made to section 5.3. Referring to the ESRIN recommendations [Ref. 5] in section 3.2. Included the integration test procedure file as Appendix A. Included the acceptance test procedure file as Appendix A. Removed recommendation to use ESRIN and IRF-U as test sites in section ???. Added the NDC acceptance test into the table in section 2.2.
0	2	95 Apr 03	Changes related to ESRIN comments to version 0.0: Reference to [Ref. 2] for dead-lines in section 2.2. Removed the statement that AT are not covered by IRF-U resources in section 2.3, page 3. Clarified the test steps on section 2.5, page 3. Clarified that The unit test environment will be made available to ESRIN, section 4.1, page 4. Updated the schedule: ISDAT 2.1 IT week 15 instead of week 14, section 2.2. Changes related to ESRIN comments to version 0.1 document (see fax from P Donzelli 11 Jan 1995): Added "Compliance with interface documents" as an IT purpose in section 5.1, page 6.
			Continues on next page

Document Status Sheet			
1. Document Title: <b>CSDS-UI ISDAT Test Plan</b>			
2. Document Reference Number: <b>DS-IRF-TP-0001</b>			
3. Issue	4. Revision	5. Date	6. Reason for Change
1	0	95 Apr 12	All test procedures are updated. All test procedure file names are changed. The IT and AT test procedures have been merged, see Appendix A. The calculator unit test procedure has been added in Appendix ???. The search unit test procedure has been added in Appendix ???. Added the factory AT in the schedule in section 2.2, page 2. Removed Test platform IP number as a part of the test report in section 4.2.3 page 5. Removed the disclaimer that search is not included in version 2.0, section 4.2.4, page 5. Added the calculator unit test in section 4.2.4, page 5. Numerous changes in sections 4.3.1 and 5.3 to account for the changed file names. Removed the statement that network connections will be tested at the AT, section 6.2, page 7. As a consequence, section 6.3, page 7 has been changed accordingly.
1	1	95 May 24	Changes made after the factory AT: Removed all unit test procedures from the document. The AT procedure in Appendix A has been updated. The AT procedure is valid for CDF templates of version 1.7.
2	0	95 Sep 29	Changes related to ISDAT 2.2, CSDS UI Release 4: The AT procedure is now based on the AT distributed to the National Data centres by ESRIN for Release 3. Added schedule for R4 in section 2.2. Typo corrected in section 4.2.3. Complete integration test reports will only be given for ISDAT v2.1, section 4.2.3. Referring to Appendix A in section 4.3.1. For ISDAT 2.2 only ESRIN provided test files are used, section 4.3.1. Typo correction in section 5.3. A new compliance matrix has been included in the AT procedure file. The following test procedures have been added: ISDAT_32, ISDAT_33, ISDAT_34, ISDAT_35, ISDAT_36. The following test cases have been modified or added/deleted in already existing test procedures: ISDAT_1.9d, ISDAT_1.9e, ISDAT_7.2, ISDAT_9, ISDAT_9.7, ISDAT_20, ISDAT_21.10, ISDAT_21.11, ISDAT_21.12, ISDAT_21.13, ISDAT_30.
2	1	95 Oct 14	Final modifications for the ISDAT 2.2 (Release 4) delivery. The following test cases have been (slightly) modified: ISDAT_1.1, ISDAT_1.6, ISDAT_1.8, ISDAT_1.7, ISDAT_1.9, ISDAT_1.10, ISDAT_1.11, ISDAT_2.3, ISDAT_3.1, ISDAT_4.3, ISDAT_7.2, ISDAT_8.8, ISDAT_8.9, ISDAT_8.10, ISDAT_9.2-11, ISDAT_10.1, ISDAT_13, ISDAT_14
2	2	95 Nov 23	Updated the test procedure (Appendix A); Primarily test procedure ISDAT-08

---

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Purpose of the document . . . . .	1
1.2	Scope of the software . . . . .	1
1.3	Reference Documents . . . . .	1
1.4	Overview of the document . . . . .	2
<b>2</b>	<b>Test Plan Overview</b>	<b>2</b>
2.1	Organisation . . . . .	2
2.2	Master Schedule . . . . .	2
2.3	Resources Summary . . . . .	3
2.4	Responsibilities . . . . .	3
2.5	Overview of the tests . . . . .	3
<b>3</b>	<b>Administrative Procedures</b>	<b>3</b>
3.1	Anomaly reporting and resolution . . . . .	3
3.2	Task iteration policy . . . . .	3
3.3	Deviation policy . . . . .	4
3.4	Control procedures . . . . .	4
3.5	Standards, practices and conventions . . . . .	4
<b>4</b>	<b>Unit Test Plan</b>	<b>4</b>
4.1	Purpose of unit tests . . . . .	4
4.2	Unit test activities . . . . .	4
4.2.1	Code inspection . . . . .	4
4.2.2	Functionality test . . . . .	5
4.2.3	Test reports . . . . .	5
4.2.4	Units to be tested . . . . .	5
4.3	Test procedures . . . . .	6
4.3.1	Test prerequisites . . . . .	6
<b>5</b>	<b>Integration test Plan</b>	<b>6</b>
5.1	Purpose of the test . . . . .	6
5.2	Test set-up and prerequisites . . . . .	6
5.3	Test procedure . . . . .	6

---

<b>6</b>	<b>Acceptance Test Plan</b>	<b>6</b>
6.1	Purpose of the acceptance test . . . . .	6
6.2	Test activities . . . . .	7
6.3	Test set-up and prerequisites . . . . .	7
6.4	Test procedures . . . . .	7
<b>A</b>	<b>Integration/acceptance Test Procedure</b>	<b>8</b>

## 1 Introduction

### 1.1 Purpose of the document

This document describes the verification and validation plan for the CSDS User interface, data manipulation part (ISDAT sub-system). It is intended for use within the CSDS UI software development teams at IRF-U, ESRIN, ESTEC, and RAL. The document also contains the proposed acceptance test plan. This issue is valid for CSDS ISDAT 2.2 distributed as CSDS UI Release 4.

### 1.2 Scope of the software

The scope of the CSDS UI software is to provide tools for data manipulation and display of the CSDS data bases. The scope of the ISDAT sub-system is to provide tools for data manipulation and display. subsectionDefinitions, acronyms, and abbreviations Words or expressions typed in *italics* signify exact names of functions or files or exact commands.

The used acronyms and abbreviations are explained in Table 1.

Acronym	Meaning
AT	Acceptance test
CDF	Common Data Format
CD-ROM	Compact Disc Read Only Memory
CSDS	Cluster Science data System
ESA	European Space Agency
ESRIN	European Space Research Institute
ESTEC	European Space Technology Centre
GUI	Graphical User Interface
ID	Identification
IDL	Interactive Data Language
IRF-U	Institutet för Rymdfysik, Uppsalaavdelningen Swedish Inst. of Space Phys., Uppsala Division
ISDAT	Interactive Science Data Analysis Tool
IT	Integration Test
N/A	Not Applicable
NDC	National Data Centre
P-M-I	Project-Member-Instrument
PPD	Prime Parameter Data
RAL	Rutherford Appleton Laboratory
RAM	Random Access Memory
R2, R3, R4	Release 2, 3, 4
SPD	Summary Parameter Data
SPMP	Software Project Management Plan
SSL	Symbolic Selection Line
STL	Symbolic Time Line
TM	Time Manager
UI	User Interface
UR	User Requirement
UT	Unit Test

Table 1: Acronyms and abbreviations

### 1.3 Reference Documents

- [1] CSDS-UI internal interface control document. Technical Report DS-ESR-ID-0002, ESRIN, August 1994.
- [2] CSDS User Interface Software Project Management Plan. Technical Report DS-ESR-MP-0001, ESRIN, April 1995.

- [3] CSDS User Interface user requirements document. Technical Report DS-EST-RS-0003, ESTEC, February 1995.
- [4] CSDS-UI external interface control document. Technical Report DS-ESR-ID-0001, ESRIN, October 1994. Issue 1.1.
- [5] CSDS-UI Unit Test and Integration Test Guidelines. Technical Report DS-ESR-TN-0007, ESRIN, December 1994.
- [6] CSDS User Interface ISDAT Architectural Design Document. Technical Report DS-IRF-AD-0001, IRF-U, September 1995. Issue 1.0.
- [7] CSDS User Interface, ISDAT Detailed Design Document. Technical Report DS-IRF-DD-001, IRF-U, September 1995.
- [8] CSDS User Interface, ISDAT Installation Manual. Technical Report DS-IRF-IM-0001, IRF-U, September 1995.
- [9] CSDS User Interface, ISDAT User Manual. Technical Report DS-IRF-UM-0001, IRF-U, September 1995.

## 1.4 Overview of the document

This document covers test plans for three development phases, unit tests, integration tests, and acceptance tests. The first three sections give a background applicable for all phases in terms of introduction, overview and administrative processes. Sections four to six cover the three specific test plans.

## 2 Test Plan Overview

### 2.1 Organisation

The organisation of the CSDS User Interface software development is described in [Ref. 2]. We are here concerned with the sub-system under development by IRF-U, the ISDAT server and client package.

### 2.2 Master Schedule

The scheduled time periods<sup>1</sup> for the tests are the following:

Test	Scheduled period
Unit tests	Included in the unit coding tasks
ISDAT 2.0 integration test	week 50 1994
ISDAT 2.1 integration test	week 15 1995
ISDAT 2.1 factory AT	week 16 1995
ESRIN R3 Acceptance test	week 22 - 23 1995
NDC R3 Acceptance test	week 25 - 28 1995
ISDAT 2.2 integration test	week 39 1995
ISDAT 2.2 factory AT	week 40 1995
ESRIN R4 AT	week 42 1995

Dead-lines are given in [Ref. 2].

<sup>1</sup>Note that some units should be tested both for ISDAT versions 2.0 and 2.1

## 2.3 Resources Summary

The unit tests and integration tests activities are performed by the IRF-U technical manager and IRF-U programming staff, see [Ref. 2]. The software development platforms will be used for the tests.

## 2.4 Responsibilities

The IRF-U programmers are responsible for the unit tests. The IRF-U technical manager is responsible for the integration tests. (see [Ref. 2]). ESRIN is responsible for the acceptance test.

## 2.5 Overview of the tests

This document relates to one sub-system of the CSDS User Interface software package. Other sub-systems are developed by ESRIN and RAL. The IRF-U CSDS UI (ISDAT) sub-system integration and acceptance by ESRIN will follow six steps:

1. **ISDAT SVVP/AT** (this document) delivery by IRF-U and acceptance by ESRIN.
2. **Unit tests** where the units are checked against the detailed design document, [Ref. 7], as described in section 4.
3. **Integration tests** where the ISDAT subsystem is tested by IRF-U against the architectural design, [Ref. 6], as described in section 5. As a final part of the integration test, an Acceptance test will be performed where ESRIN will be given the opportunity to witness the tests at IRF-U. Once ESRIN has agreed that the software behaves according to requirements, it will be put under the agreed configuration control with the support of IRF-U.
4. **Partial integration** After sub-system successful building, ESRIN will perform a partial integration (ESRIN + IRF-U) and validation. In order to move to next step, a second partial integration is required with RAL sub-system (ESRIN + RAL).
5. **Acceptance tests** where the ISDAT sub-system is tested by ESRIN against the user requirements [Ref. 3] as described in section 6. ISDAT functions requiring interfaces to ESRIN and RAL CSDS UI sub-systems will, at this stage be tested against a global CSDS-UI test set-up.
6. **NDC Acceptance** Execution of CSDS-UI acceptance tests at the national data centres.

System tests, normally performed to test against the software requirements, are not planned on ISDAT sub-system level, since no software requirements document has been developed on this level.

# 3 Administrative Procedures

## 3.1 Anomaly reporting and resolution

Anomalies related to unit tests are reported, discussed and normally resolved at bi-weekly IRF-U progress meetings. Any unresolved items will be reported to CSDS UI progress meetings (see [Ref. 2]).

## 3.2 Task iteration policy

Due to the short implementation time, no test harness, and very little of test software will be developed. Most of the tests will make use of other CSDS-UI ISDAT units for testing. The disadvantage is that the unit under test will not be "isolated" from the rest of the system. However, the strategy will be possible to use thanks to the fact that the development is based on re-use of existing units. Thus, all tests can be performed using old or new ISDAT components that are normal components in the system.



Where feasible, problems encountered during the unit and integration tests will be solved at the time of the test. However, where major problems have been encountered and solved, the test has to be re-done from the start of the test procedure, as described in [Ref. 5].

### **3.3 Deviation policy**

All deviations from the baseline will be reported to the CSDS UI progress meetings.

### **3.4 Control procedures**

Control procedures described in the CUI Software Project Management Plan [Ref. 2] will be followed.

### **3.5 Standards, practices and conventions**

Coding standards, practices and conventions are described in the general part of the CUI ISDAT Detailed design Document [Ref. 7].

## **4 Unit Test Plan**

### **4.1 Purpose of unit tests**

The purpose of the unit tests for the ISDAT sub-system are three-fold:

1. To test and verify the software at the end of the coding phase.
2. To provide tools for verification during the maintenance phase.
3. To provide tools to verify software ported from the origin UNIX platform to OPEN/VMS platforms.

The complete unit test environment will be provided by IRF-U with the software delivery to ESRIN.

### **4.2 Unit test activities**

#### **4.2.1 Code inspection**

On unit level, the software will be manually checked against the coding standards given in the general part of the detailed design document [Ref. 7]. This is normally done during the coding phase by means of source code inspection by the technical manager. The following verifications will be performed where applicable:

- Functionality
- Performance
- Recovery from invalid input
- Coding standards
- Resource consumption
- Ease of integration

#### 4.2.2 Functionality test

The functionality will be verified on the units by executing the code where such verification is relevant (see section 4.2.4). The unit verifications will all be using test data files provided within the CSDS-UI ISDAT Server software package. The test procedures and expected results will be described in very precise ASCII files iprovided with the source code package. Where possible, the expected results will given by reference to on-line result files. Else the expected result will be quantified in writing.

The following verifications will be tested by executing code where applicable:

- Functionality
- Performance
- Recovery from invalid input
- Resource consumption

#### 4.2.3 Test reports

The inspection part of the unit verification will not be formally documented. The functionality tests for ISDAT 2.1 will be documented in short reports stating:

1. Unit ID
2. Input data file identification
3. Supporting ISDAT version
4. Date of test
5. Remarks and non-conformance
6. Signature of responsible person
7. Operative system
8. Motif version number
9. CDF version

#### 4.2.4 Units to be tested

All new or substantially modified source code will be inspected according to section 4.2.1. Re-used code will not be modified to conform with new coding conventions according to [Ref. 7]. The unit functionality verification according to section 4.2.2 will be performed on the following units (see [Ref. 6] for unit definitions):

**Unit 1.2 CSDS module**

**Unit 2.1 cuitm**

**Unit 2.2 cuiqr**

**Unit 2.2.5 calculator**

**Unit 2.3 cuimeta**

**Unit 2.4 search**

**Unit 2.7 cuistat**

## 4.3 Test procedures

### 4.3.1 Test prerequisites

Test requisites are listed in Appendix A. For ISDAT v2.2 only ESRIN provided test data files will be used.

## 5 Integration test Plan

### 5.1 Purpose of the test

The purpose of the integration test is to verify the CSDS-UI ISDAT sub-system conformance with the architectural design [Ref. 6]. This test will also validate some software installation [Ref. 8] and user manuals [Ref. 9].

Interfaces to the ESRIN software will not be verified on this level. Only UNIX to UNIX connections will be tested.

The integration test will comprise the following steps:

1. Unit level tests completeness. Applicable documents are unit test protocols (see section 4).
2. CSDS-UI ISDAT System installation with [Ref. 8] as reference.
3. Compliance with the software manuals [Ref. 9].
4. Verification of the software sub-system compliance with the architectural design [Ref. 6].
5. Compliance with the external interface document [Ref. 4] and the internal interface document [Ref. 1].

### 5.2 Test set-up and prerequisites

The integration tests will be performed with the same requisites as for the unit tests, described in section 4.3.1, page 6.

### 5.3 Test procedure

This Integration test procedure is available as an ASCII file:

```
~csds/isdat_test/IT-AT/it_proc.txt
```

The file is included as Appendix A in this document. For the actual test it is recommended to make a copy of the file in the test directory, follow the procedure and make notes directly in the file in a terminal window during the test.

## 6 Acceptance Test Plan

### 6.1 Purpose of the acceptance test

The purpose of the acceptance test is to validate the delivered CSDS-UI ISDAT subsystem with respect to the user requirements [Ref. 3].

## **6.2 Test activities**

The test activity will consist of a functionality test of the CSDS-UI functions in the integrated CSDS User Interface system.

The acceptance tests will be analogous to the CSDS-UI ISDAT integration functionality test described in section 4.2.2, page 5.

## **6.3 Test set-up and prerequisites**

The acceptance test set-up is analogous to the integration test set-up described in section 5.2.

## **6.4 Test procedures**

The Acceptance test procedure is identical to the Integration test procedure, see section 5.3. The AT procedure is included in Appendix A, page 8.

---

## A Integration/acceptance Test Procedure

This AT is based on:

```
#-----  
#  
# European Space Agency - ESRIN  
# Cluster Science Data System - User Interface (CSDS-UI)  
#  
# Description :  
#     ISDAT Acceptance test procedures description.  
# $Id: isdat_tp.txt,v 1.19 1995/08/31 13:16:15 messeri Exp $  
#-----  
The file was updated 23 Nov 1995 by Gunnar Holmgren  
Updated lines are marked with "->" in the left margin.
```

User requirements vs AT compliance matrix:

UR.03        Inherent in the design - not tested on sub-system level.  
UR.07        ISDAT\_30  
UR.08 a/     ISDAT\_34  
UR.08 b/     ISDAT\_35  
UR.10c       ISDAT\_17  
UR.15        ISDAT\_9  
UR.15b       ISDAT\_9  
UR.27        ISDAT\_1  
UR.28        ISDAT\_9  
UR.29        ISDAT\_3  
UR.30        ISDAT\_2  
UR.31        ISDAT\_3  
UR.32        ISDAT\_4  
UR.32b       ISDAT\_20  
UR.33a       ISDAT\_5  
UR.33b       ISDAT\_6  
UR.33c       ISDAT\_7  
UR.33d       ISDAT\_8  
UR.34        ISDAT\_1  
UR.35        ISDAT\_10  
UR.35b       ISDAT\_20  
UR.36a/      ISDAT\_1  
UR.36b/      ISDAT\_2  
UR.36c/      ISDAT\_11  
UR.36d/      ISDAT\_1  
UR.36e/      ISDAT\_2  
UR.36f/      ISDAT\_13  
UR.36g/      ISDAT\_14  
UR.39        Not applicable to testing on sub-system level  
UR.40        Inherent in the design, not applicable to testing on sub-system level.  
UR.42b       ISDAT\_32, ISDAT\_33  
UR.43a       Inherent in the design, not applicable to testing on sub-system level  
UR.44        ISDAT\_15  
UR.45a/      ISDAT\_18  
UR.45a       ISDAT\_36  
UR.45g       ISDAT\_19  
UR.45i       ISDAT\_12  
UR.46        Inherent in the design, not applicable to testing on sub-system level  
UR.50        ISDAT\_16  
UR.51        SPMP do not makes this UR applicable to IRF-U  
             ISDAT 2.0 provides online help from the clients. (no to be ATed)  
UR.53        Not applicable to testing.  
UR.54a/      ANSI C flag (-Xc) setup in ISDAT makefiles  
UR.54b/      Inherent in the design, not applicable for testing.  
UR.54c/      Inherent in the design, not applicable for testing.

Test design ISDAT

Test Item: Interactive CDF data hadling (ISDAT)

General description

This group of tests is aimed to verify the capability of the CSDS-UI software to

HW/SW Environment

Test cases are executed on the following hw/sw configurations:

snn  
ann

Test pre-requisites:

- 1/ CSDS-UI Server and Client shall be installed and configured.
- 2/ The CSDS-UI System Manager (csdsadm) account shall:
  - have C-Shell as login shell
  - be installed and configured as CSDS-UI System Manager
- 3/ A Client account, csds, shall be created and shall:
  - have C-Shell as login shell
  - be installed and configured as a CSDS-UI Client

Test environment setup procedure.

Important Note 1:

In this document, the term "Run" refers, for both Solaris and OpenVMS platforms, to the execution of a command from a terminal window. Commands which may require more system knowledge or which are very platform dependent are written in full.

Important note 2:

This AT is written for the integrated CSDS User Interface Environment. It is, however, also used for the ISDAT sub-system AT. In the latter case, all instructions to:

"From the CSDS-UI Session Manager, start ISDAT"

Replace it by:

"From the terminal window start cuitm".

A. In an integrated CSDS User Interface environment:

This is the original ESRIN environment - it is not used neither tested at IRF-U

1/ From a terminal window, if not already done:

- on the machine where the NDC software is installed,  
login as csdsadm
- execute the server Acceptance Test set-up.
  - On Solaris: source \$CUI\_PRD\_ROOT/./CUI\_AT/bin/at\_setup
  - On OpenVMS: move to csdsprd sys\$login directory  
move to [.cui\_at] directory  
execute the following command: @[.bin]at\_setup  
move back to original working directory
- Run isdat\_test\_setup\_adm

2/ From a terminal window, if not already done:

- on the machine where the Client software is installed,  
login as a client user
- execute the server Acceptance Test set-up.
  - On Solaris: source \$CUI\_AT\_ROOT/bin/at\_setup
  - On OpenVMS: at\_setup
- Run isdat\_test\_setup\_cl
- Run cuism and log in as csds01a or csds02a
- From CSDS-UI Session Manager, press ISDAT Client button  
(see note 2 under Test environment setup procedure)

3/ From this point onward, unless otherwise specified, any action requested from the Tester shall be performed from the client user account.

B. In an ISDAT sub-system environment:

This is the test environment used at IRF-U for the in-factory AT.

- login as user csdsadm
- Run ". atenv"
- Run irfu\_test\_setup\_adm.csh
- On a separate window or workstation:
  - Run dbhstart
- login as user csds
- Run isdat\_test\_setup\_cl.csh
- Run cuitm



Test Procedure ISDAT\_00

Test Procedure objective:

Test whether ISDAT server is alive (pingdbh)

Reference UR:

None

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_00\_0 Test whether ISDAT server is alive (pingdbh)

Test Case execution:

- From csdsadm account, run pingdbh
- Expected result: Message: database <db address> is alive
- Actual result:

---

Test Procedure ISDAT\_01

Test Procedure objective:  
Subsetting CDF files in time interval  
Plot parameter vs time  
Save as flat files  
Write postscript for hard copy.

Reference UR:

UR.27  
UR.36a  
UR.34  
UR.36d

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_01\_1 Sub-set and plot a (real) parameter vs time

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-1.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-1.igr
- Expected result: A visible plot for the time interval shown on the cuitm panel (30 seconds of data). The plot contains, on the Y axis, 8 data points ranging from approximately 17 to 9.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile  
Press 0, Apply, and specify file  
AT01-1.flat in directory \$CUI\_USR\_ROOT
- Press OK
- Compare the file AT01-1.flat with the reference file \$USR\_ROOT/ref/AT01-1.flat
- Expected result: Two lines differ: the creation dates.
- Actual result:
- Optionally, check the exact values against the relevant time interval and parameter in the original CDF file C1\_PP\_EFW\_19960101\_V01.cdf by using the cdfedit tool and inspecting the reference file \$USR\_ROOT/ref/AT01-1.flat using a text editor.
- Actual result:

ISDAT\_01\_2 Test Case not used

ISDAT\_01\_3 Write as a Postscript formatted file.

Verify that the Postscript file is printable

Test Case execution:

- From cuigr, select File->Print, press button file, specify the file AT01-3.ps and press OK
- Expected result: The Postscript file AT01-3.ps is created
- Actual result:

Test Case verification:

- On a Postscript printer, print the file AT01-3.ps and the reference file \$USR\_ROOT/ref/AT01-3.ps
- Expected result: Two hard-copies are produced, differing only production dates but otherwise identical.
- Put the two hard copies on top of each other with a background

light source and compare the plots.

- Actual result:

ISDAT\_01\_4 Test Case not used.

ISDAT\_01\_5 Verify that hard copies can be produced directly from cuigr.

Test Case execution:

- From cuigr, select File->Print, press button Printer, select the appropriate local printer name, press OK

Test Case verification:

- Expected result: A hard copy identical to the one produced under ISDAT\_01\_3 is produced  
- Actual result:

ISDAT\_01\_6 Sub-set and plot a (real\*4) parameter vs time

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-6.ps to get a hard copy.  
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-6.ctm  
- If cuigr is not running: Select Clients->general->cuigr  
- From cuigr, select File->Config->Load and select the reference \$USR\_ROOT/ref/AT01-6.igr  
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resemping the hard copy created above.  
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-6.flat  
- Compare the file AT01-6.flat with the reference file \$USR\_ROOT/ref/AT01-6.flat  
- Expected result: Two lines differ - The creation dates  
- Actual result:

ISDAT\_1\_7 Objective: Plot a (int\*4) parameter vs a (int\*4) parameter.

This test case is not used for Release 4 since handling of (int\*4) is not implemented in the cuigr.

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-7.ctm  
- If cuigr is not running: Select Clients->general->cuigr  
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-7.igr  
- Expected result: Error message: "Plot 0 calculator error: Unsupported data type 9: q0"  
- Actual result:

ISDAT\_1\_8 Sub-set and plot a (int\*2) parameter vs time

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-8.ps to get a hard copy.  
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-8.ctm  
- If cuigr is not running: Select Clients->general->cuigr  
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-8.igr  
- Expected result: A visible plot corresponding to the time interval

specified on the cuitm and resemping the hard copy created above.

- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-8.flat
- Compare the file AT01-8.flat with the reference file \$USR\_ROOT/ref/AT01-8.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_8a Plot a (int\*2) parameter vs a (int\*2) parameter

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-8a.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-8a.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-8a.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resemping the hard copy created above.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-8a.flat
- Compare the file AT01-8a.flat with the reference file \$USR\_ROOT/ref/AT01-8a.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_9 Sub-set and plot a (real\*4 vector) parameter vs time

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-9.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resemping the hard copy created above. (The norm of the vector is plotted in the upper panel, and the three vector components are plotted in the lower panel)
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-9.flat and plot p0 on cuigr, button 0 on flat file dialog, p4, button 4, p5, button 5, p6, button 6. Press Apply
- Compare the file AT01-9.flat with the reference file \$USR\_ROOT/ref/AT01-9.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_9a Plot a (real\*4) vector vs a (real\*4) vector.  
Both the norm of the vector and single components are tested.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-9a.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9a.ctm

- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9a.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resampling the hard copy created above.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-9a.flat and plots p0
- Compare the file AT01-9a.flat with the reference file \$USR\_ROOT/ref/AT01-9a.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_9b Plot a (real\*4) vector vs a (real\*4) scalar.  
Both the norm of the vector and single components are tested.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-9b.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9b.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9b.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resampling the hard copy created above.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile  
press the 0 button and specify file AT01-9b.flat and plots p4
- Compare the file AT01-9b.flat with the reference file \$USR\_ROOT/ref/AT01-9b.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_9c Plot a (real\*4) vector vs a (int\*4) scalar.  
Both the norm of the vector and single components are tested.  
Not used for Release 4. (int\*4) is not implemented.

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9c.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9c.igr
- Expected result: Error message: "Plot 0 calculator error:  
Unsupported data type 9: q0"
- Actual result:

ISDAT\_1\_9d Plot a (real\*4) vector vs time.  
Verify that fill values in one vector component is properly treated, Prime parameters.  
Both the norm of the vector and single components are tested.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-9d.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9d.ctm
- If cuigr is not running: Select Clients->general->cuigr

- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9d.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resempling the hard copy created above.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-9d.flat
- Compare the file AT01-9d.flat with the reference file \$USR\_ROOT/ref/AT01-9d.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_9e Plot a (real\*4) vector vs time.

Verify that fill values in one vector component is properly treated, Summary parameters.

Both the norm of the vector and single components are tested.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-9e.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9e.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-9e.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resempling the hard copy created above.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-9e.flat
- Compare the file AT01-9e.flat with the reference file \$USR\_ROOT/ref/AT01-9e.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_1\_10 Sub-set and plot a parameter vs a parameter with different sampling rates.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-10.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-10.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-10.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resempling the hard copy created above. The time resolution corresponds to the parameter with lower resolution.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-10.flat
- Compare the file AT01-10.flat with the reference file \$USR\_ROOT/ref/AT01-10.flat
- Expected result: Two lines differ - The creation dates

- Actual result:

ISDAT\_1\_11 Sub-set and plot a derived parameter vs a parameter with different sampling rates.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT01-11.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-11.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01-11.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm and resampling the hard copy created above. The time resolution corresponds to the parameter with lower resolution.  
The y-parameter is 100 units bigger than in previous test case.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT01-11.flat
- Compare the file AT01-11.flat with the reference file \$USR\_ROOT/ref/AT01-11.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

---

Test Procedure ISDAT\_2

Test Procedure objective:

Merge CDF files retaining independent time lines.  
Join CDF files into common time line.  
Support multi-panel plots.  
User exchange of plot designs.

Reference UR:

UR.30  
UR.31  
UR.36b  
UR.36e

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_2\_1 Verify that SP and PP data can be plotted along the same time line.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT02-1.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT02-1.ctm
- If cuigr is not running: Select Clients->general->cuigr on cuitm.
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT02-1.igr
- Expected result: Two curves plotted with data of two different time resolutions but the same parameter (PP and SP).
- Actual result:

Test Case verification:

- The plot shall resemble the reference hard copy just plotted.

ISDAT\_2\_2 Test Case not used

ISDAT\_2\_3 Verify that data sets of two different time resolutions can be "joined" to the same time line.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT02-3.ps to get a hard copy.
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT02-3.igr
- Expected Result: In top panels PP plots with 4 sec resolution. In bottom panels one SP plot one with 60 sec resolution and one SP plot with 4 sec resolution. The latter should have a stepwise appearance.
- Actual result:

Test Case verification:

- From cuigr, select cuigr: File->Print, press Printer, select a printer name and press OK
- Compare the hard copies just produced. They should differ only with respect to production date.
- Check exact values for the "joined" data sets:
- From cuigr, press button "44", selecting the plot in the bottom right corner
- From cuigr, select File->SaveData->FlatFile and specify the file AT02-3.flat



- Compare the file AT02-3.flat with the reference file \$USR\_ROOT/ref/AT02-3.flat
- Expected result: Only creation date differ
- Actual result:

ISDAT\_2\_4 Test Case not used

Test Procedure ISDAT\_3

Test Procedure objective:  
Merge CDF files across midnight

Reference UR:  
UR.29

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_3\_1 Verify that data from two CDF files can be merged  
into one plot.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT03-1.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT03-1.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT03-1.igr
- Expected result:  
A visible plot corresponding to the time interval specified on the cuitm.  
The plot starts with a data gap, starts before midnight and continues across midnight with a dis-continuity at mid-night due to the test data behaviour. The plot resembles the reference hard copy just made.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify the file AT03-1.flat
- Compare the file AT03-1.flat with the reference file \$USR\_ROOT/ref/AT03-1.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

Test Procedure ISDAT\_4

Test Procedure objective:  
Results of data manipulation to be written as a new CDF file.

Reference UR:  
UR.32

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_4\_1 Verify that a real\*4 can be written to a CDF file

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-1.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-1.igr
- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- In the Selection test widget, replace "EXTN" in the proposed file name by "AT04-1"
- In the CDF window: press OK
- Expected result: A non-empty file has been created.  
In \$CUI\_USR\_ROOT, the sub directory cdfdb, check that the file CL\_SM\_MIX\_20000101\_V01\_AT04-1.cdf exists
- Actual result:

Test Case verification:

- Compare the file CL\_SM\_MIX\_20000101\_V01\_AT04-1.cdf with the reference file CL\_SM\_MIX\_20000101\_V01\_AT04-1.cdf by:  
cdfcmp \$CUI\_USR\_ROOT/ref/CL\_SM\_MIX\_20000101\_V01\_AT04-1.cdf CL\_SM\_MIX\_20000101\_V01\_AT04-1.cdf
- Expected result: Values of the 11 samples are identical.  
Generation dates differ
- Actual result:

ISDAT\_4\_2 Test Case not used

ISDAT\_4\_3 Verify that a int\*4 can be written to a CDF file  
Test case not used for Release 4 since (int\*4) handling is not implemented in cuigr.

ISDAT\_4\_4 Test Case not used

ISDAT\_4\_5 Verify that a int\*2 can be written to a CDF file

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-5.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-5.igr
- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- Replace "EXTN" in the proposed file name by "AT04-5"
- In the CDF window: press OK
- In \$CUI\_USR\_ROOT, the sub directory cdfdb, check that the file CL\_SM\_MIX20000101\_V01\_AT04-5.cdf exists
- Expected result: A non-empty file has been created.

- Actual result:

Test Case verification:

- Compare the file CL\_SM\_MIX20000101\_V01\_AT04-5.cdf with the reference file CL\_SM\_MIX20000101\_V01\_AT04-5.cdf
- Expected result: Values of the 11 samples are identical. Generation dates differ.
- Actual result:

ISDAT\_4\_6 Test Case not used

ISDAT\_4\_7 Verify that 4 parameters can be written to a CDF file

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-7.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-7.igr
- From cuigr, select File->SaveData->CDF
- Set current plot 0 by pressing button 0 in cuigr
- Press button 0 on the "Save CDF" dialog.
- Set current plot 4 by pressing button 1 in cuigr
- Press button 1 on the "Save CDF" dialog.
- Set current plot 8 by pressing button 2 in cuigr
- Press button 2 on the "Save CDF" dialog.
- Set current plot 12 by pressing button 3 in cuigr
- Press button 3 on the "Save CDF" dialog.
- Press button "Apply" on the "Save CDF" dialog
- Replace "EXTN" in the proposed file name by "AT04-7"
- In the CDF window: press OK
- In \$CUI\_USR\_ROOT, the sub directory cdfdb, check that the file CL\_SM\_MIX\_20000101\_V01\_AT04-7.cdf exists
- Expected result: A non-empty file has been created.
- Actual result:

Test Case verification:

- Compare the file CL\_SM\_MIX20000101\_V01\_AT04-7.cdf with the reference file CL\_SM\_MIX20000101\_V01\_AT04-7.cdf
- Expected result: Values of the 11 samples are identical. Generation dates differ.
- Actual result:

ISDAT\_4\_8 Test Case not used

ISDAT\_4\_9 Verify that long intervals can be written to a CDF file

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-9.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT04-9.igr
- From cuigr, select File->SaveData->CDF
- Set current plot 0 by pressing button 0 in cuigr
- Press button 0 on the "Save CDF" dialog.
- Set current plot 4 by pressing button 1 in cuigr
- Press button 1 on the "Save CDF" dialog.
- Set current plot 8 by pressing button 2 in cuigr
- Press button 2 on the "Save CDF" dialog.
- Set current plot 12 by pressing button 3 in cuigr

- Press button 3 on the "Save CDF" dialog.
- Press button "Apply" on the "Save CDF" dialog
- Replace "EXTN" in the proposed file name by "AT04-9"
- In the CDF window: press OK
- In \$CUI\_USR\_ROOT, the sub directory cdfdb, check that the file CL\_PM\_MIX20000101\_V01\_AT04-9.cdf exists
- Expected result: A non-empty file has been created.
- Actual result:

Test Case verification:

- Compare the file CL\_PM\_MIX20000101\_V01\_AT04-9.cdf with the reference file CL\_PM\_MIX20000101\_V01\_AT04-9.cdf
- Expected result: Values of the 11 samples are identical. Generation dates differ.
- Actual result:

---

Test Procedure ISDAT\_5

Test Procedure objective:  
Interactive time interval selection.

Reference UR:  
UR.33a

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_5\_1 Interactive time interval selection

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT05.ctm
- From cuitm, press button "content"
- Mark the first interval and press OK
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT05.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT05.igr
- Expected result: A visible plot corresponding to the time interval specified on the cuitm.
- Actual result:

ISDAT\_5\_2 Specify time by editing the Start time

Test Case execution:

- On cuitm, the edit fields show:  
Start "1968-10-18 08:00:00.0"  
Stop "1968-10-18 08:30:00.0"  
Interval: "1800"  
Put the cursor behind 08 in the Start field; do "Back space" and type in "9" instead of "8"; move cursor out of the edit field.
- Expected result:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:30:00.0"  
Interval: "1800"
- Actual result:  
- On cuitm, press "Update"
- Expected result: cuigr displays the new interval (data gaps in data)
- Actual result:

ISDAT\_5\_3 Specify time by editing the Stop time

Test Case execution:

- On cuitm, the edit fields show:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:30:00.0"  
Interval: "1800"  
Put the cursor behind 30 in the Stop field; do "Back space" twice and type in "10" instead of "30"; move cursor out of the edit field.
- Expected result:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:10:00.0"  
Interval: "600"  
Note that "Interval" has changed.
- Actual result:  
- On cuitm, press "Update"

- Expected result: cuigr displays the new interval
- Actual result:

#### ISDAT\_5.3 Specify time by editing the Interval time

##### Test Case execution:

- On cuitm: The edit fields show:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:10:00.0"  
Interval: "600"  
Put the cursor behind 6 in the Interval field; do "Back space"  
and type in "3" instead of "6"; move cursor out of the edit field.
- Expected result:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:05:00.0"  
Interval: "300"  
Note that "Stop" has changed.
- Actual result:  
- On cuitm: press "Update"
- Expected result: cuigr displays the new interval
- Actual result:

#### ISDAT\_5.4 Increment time by utilizing the forward arrow

##### Test Case execution:

- On cuitm: The edit fields show:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:05:00.0"  
Interval: "300"
- Press the "Forward arrow" to the right of the "Update" button.
- Expected result: cuigr displays the new interval.  
Start "1968-10-18 09:05:00.0"  
Stop "1968-10-18 09:10:00.0"  
Interval: "300"  
Note that "Stop" -> "Start".
- Actual result:

#### ISDAT\_5.5 Decrement time by utilizing the backward arrow

##### Test Case execution:

- On cuitm: The edit fields show:  
Start "1968-10-18 09:05:00.0"  
Stop "1968-10-18 09:10:00.0"  
Interval: "300"
- Press the "Backward arrow" to the left of the "Update" button.
- Expected result: cuigr displays the new interval.  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:05:00.0"  
Interval: "300"  
Note that "Stop" -> "Start".
- Actual result:

#### ISDAT\_5.6 Increment time with specified step by utilizing the forward arrow

##### Test Case execution:

- On cuitm: The edit fields show:  
Start "1968-10-18 09:00:00.0"  
Stop "1968-10-18 09:05:00.0"  
Interval: "300"
- From cuitm, select Options->Step

- Expected result: The cuitm window expands and shows an editable "Step" field.
- Edit the Step to "60.0"
- Press the "Forward arrow" to the right of the "Update" button.
- Expected result: cuigr displays the new interval.  
Start "1968-10-18 09:01:00.0"  
Stop "1968-10-18 09:06:00.0"  
Interval: "300"
- Note that "Start" and "Stop" have been incremented by 1 min while "Interval" is still 5 minutes.
- Actual result:
- Reset the cuitm window by selecting Options->Step again
- Re-set the original window size by dragging the right window border.
- Expected result: cuitm resumes its small size, including the "Interactive Time Selection".
- Actual result:

ISDAT\_5\_7 Specify time by dragging the graphical time bar.

Test Case execution:

- On cuitm "Interactive Time Selection" field:  
Move the time bar on cuitm by dragging with the middle mouse button and press update.
- Expected result: A visible plot corresponding to the new time interval specified above and updated "Start", "Stop" fields.
- Actual result:

ISDAT\_5\_8 Set the time and interval on the time bar

Test Case execution:

- On cuitm "Interactive Time Selection" field:  
Press the left mouse button anywhere along the black bar;  
keep the mouse button pressed and move towards right;  
Release the left mouse button.
- On cuitm: press update
- Expected result: "Start", "Stop", and "Interval" fields updated.  
cuigr displays the new interval.
- Actual result:

ISDAT\_5\_9 Specify time interactively on the plot panel (zoom in time)

Test Case execution:

- Zoom in time on cuigr by using the middle mouse button.
- Expected result:
  1. A visible plot corresponding to the time interval specified above.
  2. The times on the cuitm should follow the change on cuigr.
- Actual result:
- From cuigr, select Edit->Undo->Timescale"
- Expected result: The time interval before zooming should be re-established.
- Actual result:



Test Procedure ISDAT\_6

Test Procedure objective:  
Retrieval request of PPD base on SPD.

Reference UR:  
UR.33b

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_6\_1 Retrieval request of PPD base on SPD.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT06.ctm
- On cuitm, press content
- Mark the first interval and press OK
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT06.ctm again
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT06.igr
- Expected result: A visible plot showing summary data for the 10 hours time interval specified on the cuitm.
- Actual result:
- With the middle mouse button mark a smaller time interval of interest on the panel of cuigr. Note the Start and Interval selected (bottom line of igr panel plot).
- From cuigr, select Inst->CSDS\_PP->C1->FGM
- From cuigr, select Param->B\_xyz\_gse
- On cuitm, press Update
- Expected result: PP data for the selected interval is plotted.
- Actual result:

Test Procedure ISDAT\_7

Test Procedure objective:  
Coordinate transformations

Reference UR:  
UR.33c

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_7.1 Verify the possibility to multiply a vector by a matrix.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT07.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT07.igr
- Expected result: Two identical plot panels showing three components of a vector in red, green and blue.
- Actual result:
- From cuigr, select Select->Calculator
- From Calculator window, select Option->Make\_Tensor
- From Calculator Tensor window, select File->Read\_Tensor and select the reference file \$USR\_ROOT/ref/AT07.igt
- Expected result: The tensor Matrix becomes:  
0 0 1  
1 0 0  
0 1 0
- Actual result:
- From Calculator Tensor window, press Define and Close buttons
- On the igr calculator edit:  
p4y={T0\*q0}[0] and Apply  
p5y={T0\*q0}[1] and press Apply  
p6y={T0\*q0}[2] and press Apply
- From Calculator window, press Close
- From cuitm, press Update
- Expected result: Three curves RGB representing the three components of the B\_xyz\_gse vector - now with the vector rotated, manifested by red->green, green->blue, blue->red in the lower panel.
- Actual result:

Test Case verification:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT07-1.ps to get a hard copy.
- From cuigr, select File->Print
- Press "printer" and "Portrait" radio buttons and press OK
- Put the two hard copie on top of each other; shift the sheets so that plot 1 on one sheet covers plot 0 on the other sheet.
- Expected result: Identical curves but different line types.
- Actual result:

ISDAT\_7.2 Verify Cartesian to polar transformations.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT07-2.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference

file \$USR\_ROOT/ref/AT07-2.igr

- Expected result: Two plot panels, the upper showing three components of a vector in red, green and blue, the lower showing the polar components R, PHI and THETA in red, green and blue.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile and specify file AT07-2.flat and plots p0 p1 p2 p4 p5 p6
- Compare the file AT07-2.flat with the reference file \$USR\_ROOT/ref/AT07-2.flat
- Expected result: Two lines differ - The creation dates
- Actual result:

---

Test Procedure ISDAT\_8

Test Procedure objective:  
Arithmetic operations.

Reference UR:  
UR.33d

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_8\_1 Add two variables

Test Case execution:

- On a Postscript printer,  
print the reference file \$USR\_ROOT/ref/AT08-1.ps to  
get a hard copy.
- From cuitm, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08.ctm
- From cuitm, press Content, select the first interval and press OK
- From cuitm, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08.ctm again
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-1.igr
- Expected result: Three curves the lowest is the sum of the two upper.
- Actual result:

Test Case verification :

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On cuitm load file AT08-1b.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog
  - > - Set current plot 4 and press button 1 in the "flat file" dialog
  - > - Set current plot 8 and press button 2 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file AT08-1b.flat
  - > - Inspect the created file and verify that column 4 is the sum  
of columns 2 and 3. (a reference file is also available)
  - > Note that only the first 6 decimals are expected to agree  
due to the 32-bit finite precision in the operation.

ISDAT\_8\_1a Plot a manipulated quantity vs another manipulated qty.

Test Case execution:

- On a Postscript printer,  
print the reference file \$USR\_ROOT/ref/AT08-1a.ps to  
get a hard copy.
- From cuitm, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-1a.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-1a.igr
- Expected result: A visible plot corresponding to the time interval  
specified on the cuitm and resembling the hard copy created above.
- Actual result:
  - > - On cuitm load file AT08-1b.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog

- > - Set current plot 4 and press button 1 in the "flat file" dialog
- > - Press apply in the "flat file dialog and save as file AT08-1a.flat
- > - Inspect the created file and verify that column 1 and 2 are
- > identical and equal to the 4:th
- > column of file AT08-1b.flat (a reference file is also available)

#### ISDAT\_8\_2 Subtract two variables

##### Test Case execution:

- On a Postscript printer,  
print the reference file \$USR\_ROOT/ref/AT08-2.ps to  
get a hard copy.
- > - From cuitm, select File->Config->Load and select the reference
- > file \$USR\_ROOT/ref/AT08-1a.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-2.igr
- Expected result: Three curves the lowest is the difference  
between the two upper.
- Actual result:

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
- > - On cuitm load file AT08-1b.ctm and press "update"
- > - On cuigr select File->SaveData->flatFile
- > - Set current plot 0 and press button 0 in the "flat file" dialog
- > - Set current plot 4 and press button 1 in the "flat file" dialog
- > - Set current plot 8 and press button 1 in the "flat file" dialog
- > - Press apply in the "flat file dialog and save as file AT08-2.flat
- > - Inspect the created file and verify that column 4 is the difference
- > between columns 3 and 2. (a reference file is also available)
- > Note that only the first 6 decimals are expected to agree
- > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_3 Multiply two variables

##### Test Case execution:

- On a Postscript printer,  
print the reference file \$USR\_ROOT/ref/AT08-3.ps to  
get a hard copy.
- > - From cuitm, select File->Config->Load and select the reference
- > file \$USR\_ROOT/ref/AT08-1a.ctm
- > - If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-3.igr
- Expected result: Three curves the lowest is the product of  
the two upper.
- Actual result:

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
- > - On cuitm load file AT08-1b.ctm and press "update"
- > - On cuigr select File->SaveData->flatFile
- > - Set current plot 0 and press button 0 in the "flat file" dialog
- > - Set current plot 4 and press button 1 in the "flat file" dialog

- > - Set current plot 8 and press button 1 in the "flat file" dialog
- > - Press apply in the "flat file dialog and save as file AT08-3.flat
- > - Inspect the created file and verify that column 4 is the product of columns 3 and 2. (a reference file is also available)
- > Note that only the first 6 decimals are expected to agree
- > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_4 Divide two variables

##### Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT08-4.ps to get a hard copy.
- > - From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-1a.ctm
- > - If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-4.igr
- Expected result: Three curves the lowest is the quota of the two upper.  
Note the division by zero in the first samples
- Actual result:

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
- > - On cuitm load file AT08-1b.ctm and press "update"
- > - On cuigr select File->SaveData->flatFile
- > - Set current plot 0 and press button 0 in the "flat file" dialog
- > - Set current plot 4 and press button 1 in the "flat file" dialog
- > - Set current plot 8 and press button 1 in the "flat file" dialog
- > - Press apply in the "flat file dialog and save as file AT08-4.flat
- > - Inspect the created file and verify that column 4 is the quotient of columns 3 and 2. (a reference file is also available)
- > Note that only the first 6 decimals are expected to agree
- > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_5 Override precedence rules in arithmetic operations

##### Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT08-5.ps to get a hard copy.
- > - From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-1a.ctm
- > - If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-5.igr
- Expected result: Three curves the lowest is the arithmetic operation.
- Actual result:

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
- > - On cuitm load file AT08-1b.ctm and press "update"
- > - On cuigr select File->SaveData->flatFile
- > - Set current plot 0 and press button 0 in the "flat file" dialog

- > - Set current plot 4 and press button 1 in the "flat file" dialog
- > - Set current plot 8 and press button 1 in the "flat file" dialog
- > - Press apply in the "flat file dialog and save as file AT08-5.flat
- > - Inspect the created file and verify that column 4 is
- > in agreement with the expression  $q_0 * (q_4 - 5)$
- > where  $q_0$  is printed in column 2 and  $q_4$  in column 3 of the file.
- > (a reference file is also available)
- > Note that only the first 6 decimals are expected to agree
- > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_6 Scalar multiplication of two vectors

##### Test Case execution:

- On a Postscript printer,  
print the reference file `$USR_ROOT/ref/AT08-6.ps` to  
get a hard copy.
- From `cuitm`, select File->Config->Load and select the reference  
file `$USR_ROOT/ref/AT08-6.ctm`
- From `cuigr`, select File->Config->Load and select the reference  
file `$USR_ROOT/ref/AT08-6.igr`
- Expected result: Three panels:  
The upper left is the norm of the vector;  
The upper right show the vector components  
The lowest is the scalar product of the vector with itself.  
The upper left should be the square root of the lower curve.
- Actual result:

##### Test Case verification:

- On `cuigr`: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On `cuitm` load file `AT08-6b.ctm` and press "update"
  - > - On `cuigr` select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog
  - > - Set current plot 4 and press button 1 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file `AT08-4.flat`
  - > - Inspect the created file and verify that column 3 is the square  
of columns 2. (a reference file is also available)
  - > Note that only the first 5 decimals are expected to agree
  - > due to the 32-bit finite precision in the operation.

#### ISDAT\_8-7 Vectorial multiplication of two aligned vectors

##### Test Case execution:

- On a Postscript printer,  
print the reference file `$USR_ROOT/ref/AT08-7.ps` to  
get a hard copy.
- > - From `cuitm`, select File->Config->Load and select the reference  
file `$USR_ROOT/ref/AT08-6.ctm`
- > - If `cuigr` is not running: Select Clients->general->`cuigr`
- From `cuigr`, select File->Config->Load and select the reference  
file `$USR_ROOT/ref/AT08-7.igr`
- Expected result: Three panels:  
The upper left is the norm of the vector;  
The upper right show the vector components  
The lowest is the vector product of the vector with itself.  
It should thus be zero.

##### Test Case verification:

- On `cuigr`: File->Print Printer Portrait OK

- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On cuitm load file AT08-7.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog
  - > - Set current plot 4 and press button 1 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file AT08-7.flat
  - > - Inspect the created file and verify that column 3 is 0.
  - > (a reference file is also available)

#### ISDAT\_8\_8 Vectorial multiplication of two perpendicular vectors

##### Test Case execution:

- On a Postscript printer,
  - print the reference file \$USR\_ROOT/ref/AT08-8.ps to
  - get a hard copy.
- > - From cuitm, select File->Config->Load and select the reference
- > file \$USR\_ROOT/ref/AT08-6.ctm
- > - If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference
- file \$USR\_ROOT/ref/AT08-8.igr
- From cuigr, select Select->Calculator
- From Calculator window, select Option->Make\_Tensor
- From Calculator Tensor window, select Tensor: T0
- From Calculator Tensor window, select File->Read\_Tensor and
- select the reference file \$USR\_ROOT/ref/AT08-0.igt
- From Calculator Tensor window, press Define
- From Calculator Tensor window, select Tensor: T1
- From Calculator Tensor window, select File->Read Tensor and
- select the reference file \$USR\_ROOT/ref/AT08-1.igt
- From Calculator Tensor window, press Define and Close
- From cuigr calculator window, press Close
- From cuitm, press Update
- Expected result: Three panels:
  - The upper left is the norm of the vector;
  - The upper right show the vector components
  - The lowest is the vector product of the vector x-component
  - with the y-component.

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On cuitm load file AT08-8b.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 40 and press button 0 in the "flat file" dialog
  - > - Set current plot 41 and press button 1 in the "flat file" dialog
  - > - Set current plot 4 and press button 1 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file AT08-8.flat
  - > - Inspect the created file and verify that column 4 is the product
  - > of columns 2 and 3. (a reference file is also available)
  - > Note that only the first 6 decimals are expected to agree
  - > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_9 Manual input of arithmetic expression

##### Test Case execution:

- From cuitm, select File->Config->Load and select the reference
- file \$USR\_ROOT/ref/AT08.ctm



- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-9.igr
- From cuigr, select Select->Calculator
- On Calculator window, type in an arbitrary arithmetic expression with the limitations
  1. Start it with "p8y="
  2. Use only quantities "q0" and "q4" in the expression for quantities.
- Press "Apply"
- Expected result: Error message or accepted expression
- Actual result:
- Correct the expression until it is accepted by the calculator
- From cuitm, press Update
- Expected result: The resulting plot shown in the lowest panel of cuigr
- Actual result:

#### ISDAT\_8\_10 Verify Tensor operations with non-integer elements

##### Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT08-10.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-10.ctm
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT08-10.igr
- From cuigr, select Select->Calculator
- From Calculator window, select Option->Make\_Tensor
- From Calculator Tensor window, select Tensor: T0
- From Calculator Tensor window, select File->Read\_Tensor and select the reference file \$USR\_ROOT/ref/AT08-10.igt
- From Calculator Tensor window, press Define and Close
- From the igr calculator window, press Close
- From cuitm, press Update
- Expected result: Two panels:
  - The upper is the original vector components
  - The lowest is the tensor product of the vector

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On cuitm load file AT08-10b.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog
  - > - Set current plot 1 and press button 1 in the "flat file" dialog
  - > - Set current plot 2 and press button 1 in the "flat file" dialog
  - > - Set current plot 4 and press button 0 in the "flat file" dialog
  - > - Set current plot 5 and press button 1 in the "flat file" dialog
  - > - Set current plot 6 and press button 1 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file AT08-10.flat
  - > - Inspect the created file and verify that column 7 is in agreement with the tensor multiplication. For example for the first line:
  - >
  - > | 0.4 0.0 0.0 | | 43 | | 17.20 |
  - > | 0.0 0.1 0.0 | \* | -43 | = | -43.00 |
  - > | 0.0 0.0 -0.6 | | 13 | | -7.8 |
  - >
  - > where the tensor is specified in the calculator, the

- > left hand column matrix is shown in the flat file columns 2-4,
- > the resulting column matrix is given by the flat file
- > columns 5-7. (a reference file is also available)
- > Note that only the first 6 decimals are expected to agree
- > due to the 32-bit finite precision in the operation.

#### ISDAT\_8\_11 Verify combination of all operators

##### Test Case execution:

- On a Postscript printer,  
print the reference file \$USR\_ROOT/ref/AT08-11.ps to  
get a hard copy.
- From cuitm, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-11.ctm
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT08-11.igr
- From cuitm, press Update
- Expected result: Three panels
- Actual result:

##### Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
  - > - On cuitm load file AT08-11b.ctm and press "update"
  - > - On cuigr select File->SaveData->flatFile
  - > - Set current plot 0 and press button 0 in the "flat file" dialog
  - > - Set current plot 4 and press button 1 in the "flat file" dialog
  - > - Set current plot 8 and press button 1 in the "flat file" dialog
  - > - Press apply in the "flat file dialog and save as file AT08-11.flat
  - > - Inspect the created file and verify that column 4 is the expression:
    - >
    - >  $(-1) * (q0 * (q4 - 2.0)) / 2.5$
    - >
  - > where q0 is printed in column 2 of the flat file and
  - > q4 is printed in column 3.
  - > (a reference file is also available)
  - > Note that only the first 6 decimals are expected to agree
  - > due to the 32-bit finite precision in the operation.

---

Test Procedure ISDAT\_9

Test Procedure objective:  
Subsetting CDF files on variables and search criteria.  
Search files according to science criteria.

Reference UR:

UR.28  
UR.15  
UR.15b

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_9\_1 Subset files using CSDS-UI Catalogue browser search result

Test Case execution:

- From cuitm, press the Catalog button
- Expected result: A list of intervals pops up.
- Actual result:
- Select the first interval and press OK
- Expected result: The time period of the selected interval is propagated to cuitm
- Actual result:
- In the directory \$CUI\_QUERY\_RESULT, delete the file query\_result.res
- Expected result: In the cuitm the Catalog button is insensitive.
- Actual result:

ISDAT\_9\_2 Search in science data using "<" operator.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-2.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-2.igr
- Expected result: One sinusoidal curve.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search window, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-2.sea
- From Search window, press Search
- Expected result: 3 matching intervals are found and listed  
The intervals correspond to the time intervals where the cuigr plot shows negative values.
- Actual result:
- Mark the first interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0 and file AT09-2.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-2.flat with the reference file \$USR\_ROOT/ref/AT09-2.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_3 Search in science data using ">" operator.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-3.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-3.igr
- Expected result: One sinusoidal curve.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-3.sea
- From Search, press Search
- Expected result: 4 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0 and file AT09-3.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-3.flat with the reference file \$USR\_ROOT/ref/AT09-3.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_4 Search in science data using ">" operator with two variables.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-4.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-4.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-4.sea
- From Search, press Search
- Expected result: 3 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-4.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-4.flat with the reference file \$USR\_ROOT/ref/AT09-4.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_5 Search in science data using ">" and "&&" operator.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-5.ctm

- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-5.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-5.sea
- From Search, press Search
- Expected result: 2 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-5.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-5.flat with the reference file \$USR\_ROOT/ref/AT09-5.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_6 Search in science data using ">" and "||" operators.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-6.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-6.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-6.sea
- From Search, press Search
- Expected result: 4 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-6.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-6.flat with the reference file \$USR\_ROOT/ref/AT09-6.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_7 Search in science data using "!" operator.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-7.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference

- file \$USR\_ROOT/ref/AT09-7.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-7.sea
- From Search, press Search
- Expected result: 4 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-7.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-7.flat with the reference file \$USR\_ROOT/ref/AT09-7.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_8 Search in science data using ">" and "+" operators.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-8.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-8.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-8.sea
- From Search, press Search
- Expected result: 2 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-8.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-8.flat with the reference file \$USR\_ROOT/ref/AT09-8.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_9 Search in science data using ">" and "-" operators.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-9.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-9.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:

- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-9.sea
- From Search, press Search
- Expected result: 2 matching intervals are found and listed
- Actual result:
- Mark the second interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-9.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-9.flat with the reference file \$USR\_ROOT/ref/AT09-9.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_10 Search in science data using ">" and "\*" operators.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-10.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-10.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-10.sea
- From Search, press Search
- Expected result: 4 matching intervals are found and listed
- Actual result:
- Mark the third interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-10.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-10.flat with the reference file \$USR\_ROOT/ref/AT09-10.flat
- Expected result: Only date differ
- Actual result:

ISDAT\_9\_11 Search in science data using ">" and "/" operators.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-11.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-11.igr
- Expected result: Two sinusoidal curves, on a logarithmic scale.
- Actual result:
- From cuitm, select Clients->General->Search
- From Search, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT09-11.sea

- From Search, press Search
- Expected result: 4 matching intervals are found and listed
- Actual result:
- Mark the third interval in the Matching interval list
- Press Update tm;
- Expected result: cuigr shows only the marked interval
- Actual result:
- From cuigr, select File->SaveData->FlatFile.
- Specify p0, p1 and file AT09-11.flat
- From Search, select File->Exit

Test Case verification:

- Compare the file AT09-11.flat with the reference file \$USR\_ROOT/ref/AT09-11.flat
- Expected result: Only date differ
- Actual result:



Test Procedure ISDAT\_10

Test Procedure objective:  
Interface for user defined modules

Reference UR:  
UR.35

Prerequisites:

The test environment setup procedure has been executed.  
Please note that this Test Case requires access to the source code.

ISDAT\_10\_1 Verify that it is possible to remove and add clients  
This test case is not used in Release 4 at IRF-U since the cuiexx  
client will not be included in the executable code.

Test Case execution:

- From cuitm, select Client
- Expected result: 2 sub-menu entries appear: csds and general
- Actual result:
- In a other terminal window, login as csdsprd and move under the \$CUI\_PRD\_ROOT directory
- In the directory bin, rename the file cuiexx.cli\_demo to cuiexx.cli
- From cuitm, select File->Exit
- From the CSDS-UI Session Manager, start ISDAT
- From cuitm, select Client->demo
- Expected result: cuiexx appears in the menu.

NB:

cuiexx is a demo C client that calculate the maximum value of parameter E\_DUSK within the specified interval.

- Actual result:
- From cuitm, select File->Config->Load and select the reference file AT10.ctm
- From cuitm, select Clients->demo->cuiexx
- From cuitm, press 3 times forward arrow
- Expected result: in the directory where CSDS-UI Session Manager was started, the file cuiexx.res exists and contains 3 lines (one for each time the forward arrow was pressed)
- Actual result:

Post test case execution tasks:

- From cuitm, select File->Exit (stop all running ISDAT clients)
- From the terminal window where you are logged in as csdsprd, rename the file cuiexx.cli to cuiexx.cli\_demo
- From the CSDS-UI Session Manager, start ISDAT

Test Procedure ISDAT\_11

Test Procedure objective:  
Handle meta data.

Reference UR:  
UR.36c

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_11\_1 Verify that meta data can be displayed

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT11.ctm
- From cuitm, select Clients->cads->cuimeta.
- Expected result: The cuimeta window appears.
- Actual result:
- From cuimeta, press Variable, select Parameter->E\_dusk
- Expected result: A list of meta data should appear.
- Actual result:

Test Case verification:

- From cuimeta, select File->Save and specify the file AT11.mta
- Compare the file AT11.mta with the reference file \$USR\_ROOT/ref/AT11.mta
- Expected result: No difference are detected
- Actual result:

Test Procedure ISDAT\_12

Test Procedure objective:  
Status and progress info.

Reference UR:  
UR.45i

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_12\_1 Progress info is displayed when PP or SP database is too long

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT12.ctm
- From cuitm, select Clients->csds->cuistat
- From cuistat, select Param->State\_wec
- Expected result: On the cuistat window the progress is continuously reported until the data is displayed
- Actual result:

ISDAT\_12\_2 Progress info is displayed when ISDAT server is stopped while clients where connected.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT12-2.ctm
- From cuitm, select Client->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT12-2.igr
- From a terminal window, suspend ISDAT server
- Login as csdsadm
- on OpenVMS: run the command: 'posix'
- run the command: kill -STOP <dbh process id>
- From cuitm, press Update
- Expected result: After 5 sec. "cuigr progress" window appears
- From the previous opened terminal window, resume ISDAT server
- run the command: kill -CONT <dbh process id>
- From cuitm, press Update
- Expected result: "cuigr progress" window disappears, cuigr plot is updated.
- Actual result:

Test Procedure ISDAT\_13

Test Procedure objective:  
Pre-defined set of basic parameters.

Reference UR:  
UR.36f

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_13\_1 Pre-defined set of basic parameters.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT13.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Predefined and select the reference file sp\_fgm.igr
- Expected result: A pre-defined plot with three panels appears.
- Actual result:

Test Case verification:

- From cuigr, select File->SaveData->FlatFile, specify p0 as plot parameter and specify file AT13.flat
- Compare the file AT13.flat with the reference file \$USR\_ROOT/ref/AT13.flat
- Expected result: Only creation date should differ
- Actual result:

Test Procedure ISDAT\_14

Test Procedure objective:

Plot derived parameters vs. time or other parameter.

Reference UR:

UR.36g

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_14\_1 Verify that parameters can be plotted vs time or vs another parameter.

Test Case execution:

- On a Postscript printer, print the reference file \$USR\_ROOT/ref/AT14.ps to get a hard copy.
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT14.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT14.igr
- Expected result: Two panels: A hodogram representing the two first components of the B\_xyz\_gse vector and a time series of the norm of the same vector.

Test Case verification:

- On cuigr: File->Print Printer Portrait OK
- Compare the hardcopy with the reference PS file just printed.
- Expected result: Only date of creation differ
- Actual result:
- From cuigr, select File->SaveData->FlatFile, specify p0 as plot parameter and specify file AT14.flat
- Compare the file AT14.flat with the reference file \$USR\_ROOT/ref/AT14.flat
- Expected result: Only creation date should differ
- Actual result:

Test Procedure ISDAT\_15

Test Procedure objective:  
Interface towards IDL.

Reference UR:  
UR.44

Prerequisites:  
IDL 3.6.1 installed

The test environment setup procedure has been executed.

ISDAT\_15\_1 Verify that the IDL demo client can be accessed  
and that CSDS data can be plotted

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT15.ctm
- From cuitm, press Content, select the first interval and press OK
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT15.ctm again
- Run cuit
- Expected result: A IDL window with a sinusoidal plot appears.  
A non-conformance might depend on IDL licence etc.
- Actual result:

Test Case verification:

- From cuitm, press several times right arrow
- Expected result: The IDL plot reacts.
- Actual result:
- On the terminal window, stop the IDL demo client by pressing CTL-C. When IDL prompt appears, type exit followed by RETURN.
- Expected result: The IDL window and the IDL prompter disappear.
- Actual result:

Test Procedure ISDAT\_16

Test Procedure objective:  
User configurable system.

Reference UR:  
UR.50

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_16\_1 Modify default settings of cuitm

Test Case execution:

- From cuitm, select File->Exit
- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.client as follows:
  - \*.tm.project: CSDS\_SP
  - \*.tm.instrument: ASPOC
  - \*.tm.member: CL
  - \*.tm.interval: 600
- From the CSDS-UI Session Manager, start ISDAT (see note 2 under Test environment setup procedure)
- Expected result:
  1. A cuitm window appears
  2. Default of above edited parameters agree with the edited values.
- Actual result:

Post test case execution tasks:

- From cuitm, select File->Exit
- In \$CUI\_USR\_ROOT, sub-directory cfg, delete the file isdat.client
- From the CSDS-UI Session Manager, start ISDAT (see note 2 under Test environment setup procedure)
- Expected result: A cuitm window with the "old" default values.
- Actual result:

Test Procedure ISDAT\_17

Test Procedure objective:  
Log info for abnormal situations.

Reference UR:  
UR.10c

Prerequisites:  
    The test environment setup procedure has been executed.

Test Case execution:

- In \$CUI\_USR\_ROOT, sub-directory log, delete all files present if any
- From the csdsadm account:
- On Solaris: run dbhstart
- On OpenVMS: run the command 'dbh'
- Wait about 20 seconds until the prompter returns
- Expected result: The message "dbh is already running" is displayed
- Actual result:



Test Procedure ISDAT\_18

Test Procedure objective:  
Immediate reaction.

Reference UR:  
UR.45a

Prerequisites:  
    The test environment setup procedure has been executed.

Test Case execution:  
- Clear immediate reaction from ISDAT GUI should be summarized by the  
  tester after having running all previous test cases  
- Expected result: "Satisfactory" reaction to commands from ISDAT  
- Actual result:

Test Procedure ISDAT\_19 Not used

Test Procedure ISDAT\_20

Test Procedure objective:  
Limit number of concurrent users.

Reference UR:  
UR.32b  
UR.35b

Prerequisites:

The test environment setup procedure has been executed.

Test Case execution:

- From the csdsadm account, do the following:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'
- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.server as follows:  
\*.common.serverLimit: 2
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'

Test Case verification:

- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
  - Expected result: An error message "Cannot connect to ISDAT server".
- Actual result:
- From the csdsadm account:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'

Post test case execution tasks:

- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.server as follows:  
\*.common.serverLimit: 0
- From the csdsadm account:
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'

## Test Procedure ISDAT\_21

Test Procedure objective:  
Robustness, recovery and error handling

Reference UR:  
None

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_21\_1 Verify controlled behaviour if time outside data-base is specified.

### Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21.ctm
- From cuitm, press Content, select all 3 intervals and press OK
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21.igr
- From cuitm, press the backward arrow twice (outside database)
- Expected result: Plot 0 database error: "no data" on the cuigr panel.
- Actual result:

ISDAT\_21\_2 Verify that ISDAT can handle times partially outside the data base

### Test Case execution:

- From cuitm, press the forward arrow until the data gap is reached
- Expected result: On edge no error message but data only partially
- Actual result:
- From cuitm, press the forward arrow once more (in the gap)
- Expected result: Plot 0 database error: "no data" on the cuigr panel.
- Actual result:

ISDAT\_21\_3 Verify that much data can be handled (10 days, 58299 samples)

### Test Case execution:

- From cuitm, press Content, select all 3 intervals and press OK
- On cuitm, press Update
- Expected result: 58299 samples plotted with gaps
- Actual result:

ISDAT\_21\_4 Test Case not used

ISDAT\_21\_5 Verify cuigr behaviour when no server is running

### Test Case execution:

- From the csdsadm account:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'
- From cuitm, press the forward arrow
- Expected result: A system error dialog box pops up.

- Press OK.
- cuigr dies.
- Actual result:
- From cuitm, select Clients->general->cuigr
- Expected result: Error window: "Cannot connect to database <dbh address>, terminate igr"
- On Error message window, press OK
- Actual result:

ISDAT\_21\_6 Verify cuitm behaviour when no server is running

Test Case execution:

- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
- Expected result: Error window: "Cannot connect to database <dbh address>, terminate cuitm"
- Actual result:
- On Error message window, press OK
- Run dbhstart

ISDAT\_21\_7 Test Case not used.

---

ISDAT\_21\_8 Verify behaviour at erroneous file name

Test Case execution:

- From cuitm, select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT01.ctm
- Expected result: Not accepted, Error message "AT01.ctm Not a configuration file"
- On Error message window, press Cancel
- Actual result:

ISDAT\_21\_9 Load and stress.

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT22.ctm
- From cuitm, start 4 concurrent:
  - cuigr, select File->general->cuigr
  - cuigr, select File->csds->cuistat
  - cuigr, select File->csds->cuimeta
- From each cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT22.igr
- From each cuistat, select Param->state\_wec
- On cuitm, press Update
- Expected result: All clients are refreshed. The reponse is slower (but no crash).
- Actual result:

ISDAT\_21\_10 Request data from an empty CDF file

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21-10.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From igr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21-10.igr
- Expected result: Plot 0 database error: "no data"
- Actual result:

ISDAT\_21\_11 Request data from a CDF file containing fill values in every second record

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21-11.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From igr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT21-11.cuigr
- From igr, select File->SaveData->FlatFile
- Press button 0 and press button Apply
- Specify AT21-110.flat and press OK

Test Case verification:

- Compare the file AT21-111.flat with the reference file \$USR\_ROOT/ref/AT21-110.flat (On the terminal window do > diff \$USR\_ROOT/ref/AT21-110.flat AT21-110.flat)
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_21\_12 Request data from a CDF file not containing the requested variable

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT21-12.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From igr, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT21-12.cuigr
- Expected result: Plot 0 database error: "no data"
- Actual result:

ISDAT\_21\_13 Request data from a CDF file with missing SCALEMIN  
SCALEMAX, LABLAXIS and LABL\_PTR\_1

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT21-13.ctm
- If igr is not running: Select Clients->general->cuigr
- From igr, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT21-13.igr

Test Case verification:

- check that cuigr reverts from preset to auto scaling.
- Expected result:
  - The title in panel 0 shows the scale range [8.1,9.9 u\_r4\_3].
  - The title in panel 1 shows the scale range [4.005,144.559 u\_r4\_v3].
- Actual result:



Test Procedure ISDAT\_22 Not used.

Test Procedure ISDAT\_23 Not used.

---

Test Procedure ISDAT\_24

Test Procedure objective:  
Plotting facilities

Reference UR:  
None

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_24\_1 Moving borders between panels

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT24.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT24.igr
- Point with the cursor just below a horizontal panel border in the middle of the window;
- Press the left mouse button and drag the border up or down.
- Expected result: The border should be moved accordingly and the plot re-scaled
- Actual result:
- Place the cursor to the right of a vertical border and do the corresponding dragging
- Expected result: Vertical bordered moved and plot re-sized.
- Actual result:

ISDAT\_24\_2 Changing colours

Test Case execution:

- From cuigr, press p0 buttons and select Graphics->Plot->Color
- Press a colour
- Expected result: The plotted data change colours accordingly
- Actual result:

ISDAT\_24\_3 Specifying plot scales

Test Case execution:

- From cuigr, press button p0 and select Graphics->Plot->Control
  - push the Scale Manual button
- drag the Ordinate sliders
- press Update
- Expected result: The plot y-scale of the upper left plot is changed accordingly.
- Actual result:

ISDAT\_24\_4 Verify that Scales can be changed by means of editing

Test Case execution:

- From Plot Control Dialog,
- push the Scale Manual button
- edit new limits fields
- press Update
- Expected result: The plot y-scale of the upper left upper plot is changed accordingly.
- Actual result:

ISDAT\_24\_5 Changing curve type

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT29.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT29.igr
- From cuigr, select Graphics->Plot->Control
- Expected result: A line plot
- Actual result:
- From Plot Control Dialog,
- press Type Scatter
- press Update
- Expected result: "x" to mark the plot on cuigr instead of a line plot
- Actual result:
- From Plot Control Dialog,
- press Type Hodogram
- press Update
- Expected result: Arrows between the points on cuigr
- Actual result:

Test Procedure ISDAT\_30

Test Procedure objective:

Provide data security and access control consistent with the CSDS agreed data restrictions.

Reference UR:

UR.07

Prerequisites:

The test environment setup procedure has been executed.

This test procedure is only applicable to the integrated CSDS UI.

ISDAT\_30\_1

Test Case execution:

- From cuitm, press the project button, keeping the mouse button pressed
- Expected result: CSDS\_PP and CSDS\_SP appear in the popup list.
- Actual result:
- From cuitm, select File->Exit
- From the CSDS-UI Session Manager, select File->Exit
- Run cuism
- Login as csds02a (Science User)
- From the CSDS-UI Session Manager, start ISDAT
- From cuitm, press the project button, keeping the mouse button pressed
- Expected result: Only CSDS\_SP appear in the popup list (i.e. Science Users are not allowed to access PP CDF files).
- Actual result:

ISDAT\_31\_1 Verify status word in hexadecimal format

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT31-1.ctm
- From cuitm, select Clients->cstds->cuistat
- From cuistat, select Parameter->Status
- From cuitm, press Update
- From cuistat, select File->Save and specify the file AT31-1.sta
- Compare the file AT31-1.sta with the reference file \$USR\_ROOT/ref/AT31-1.sta
- Expected result: No difference between the files
- Actual result:

ISDAT\_31\_2 Verify status word in hexadecimal format.

Do not interpret first byte

Test Case execution:

- From cuistat, select Parameter->Status
- From cuistat, select Options->Format
- On Format window: Release button "interpret first byte"
- From cuitm, press Update
- From cuistat, select File->Save and specify the file AT31-2.sta
- Compare the file AT31-2.sta with the reference file \$USR\_ROOT/ref/AT31-2.sta
- Expected result: No difference between the files
- Actual result:

ISDAT\_31\_3 Verify status word in decimal format.

Do not interpret first byte

Test Case execution:

- From cuistat, select Parameter->Status
- From cuistat, select Options->Format
- On Format window: Press button "decimal"
- From cuitm, press Update
- From cuistat, select File->Save and specify the file AT31-3.sta
- Compare the file AT31-3.sta with the reference file \$USR\_ROOT/ref/AT31-3.sta
- Expected result: No difference between the files
- Actual result:

ISDAT\_31\_4 Verify status word in ASCII format.

Do not interpret first byte

Test Case execution:

- From cuistat, select Parameter->Status
- From cuistat, select Options->Format
- On Format window: Press button "ascii"
- From cuitm, press Update
- From cuistat, select File->Save and specify the file AT31-4.sta
- Compare the file AT31-4.sta with the reference file \$USR\_ROOT/ref/AT31-4.sta
- Expected result: No difference between the files
- Actual result:

---

Test Procedure ISDAT\_32

Test Procedure objective:  
Verify local server functionality.

Reference UR:  
UR 42b

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_32\_1 Overlapping local CDF files

Test Case execution:

- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-11.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-11.igr
- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- Replace "EXTN" in the proposed file name by "AT32-11"
- In the CDF window: press OK
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-12.ctm
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-12.igr
- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- Replace "EXTN" in the proposed file name by "AT32-12"
- In the CDF window: press OK
- In a terminal window, as user csds, type:  
export CUI\_ISDAT\_DB=unix:1
- As user csds Run "dbh :1"
- On cuitm do File->New manager and data base
- Verify that the popup window specifies unix:1
- Press OK
- A new manager with a differnt colour (brown) should appear.
- Run cuimgen
- From the brown cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-13.ctm
- From the brown cuitm start a brown cuigr
- From the brown cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-13.igr
- From the brown cuigr, select File->SaveData->FlatFile
- Press button 0 and press button Apply
- Specify AT32-11.flat and press OK

Test Case verification:

- Compare the file AT32-11.flat with the reference file \$USR\_ROOT/ref/AT32-11.flat  
(On the terminal window do > diff \$USR\_ROOT/ref/AT32-11.flat AT32-11.flat)
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_32\_2 Non contiguous local CDF files

Test Case execution:

- From the grey cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT32-21.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file

AT32-21.igr

- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- Replace "EXTN" in the proposed file name by "AT32-21"
- In the CDF window: press OK
- From cuitm, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-22.ctm
- From cuigr, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-22.igr
- From cuigr, select File->SaveData->CDF
- Press button 1 and press button Apply
- Replace "EXTN" in the proposed file name by "AT32-22"
- In the CDF window: press OK
- From the terminal window, do > cuimgen
- From the brown cuitm, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-23.ctm
- From the brown cuigr, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-23.igr
- From the brown cuigr, select File->SaveData->FlatFile
- Press button 0 and press button Apply
- Specify AT32-21.flat and press OK

Test Case verification:

- Compare the file AT32-21.flat with the reference file \$USR\_ROOT/ref/AT32-21.flat  
(On the terminal window do > diff \$USR\_ROOT/ref/AT32-21.flat AT32-21.flat)
- Expected result: Two lines differ - The creation dates
- Actual result:

ISDAT\_32\_3 Single local CDF file

Test Case execution:

- From the brown cuitm, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-31.ctm
- If cuigr is not running: Select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file  
\$USR\_ROOT/ref/AT32-31.igr
- From cuigr, select File->SaveData->FlatFile
- Press button 0 and press button Apply
- Specify AT32-31.flat and press OK

Test Case verification:

- Compare the file AT32-31.flat with the reference file \$USR\_ROOT/ref/AT32-31.flat  
(On the terminal window do > diff \$USR\_ROOT/ref/AT32-32.flat AT32-31.flat)
- Expected result: Two lines differ - The creation dates
- Actual result:



Test Procedure ISDAT\_33

Test Procedure objective:  
Verify local server access control.

Reference UR:  
UR 42b

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_33\_1 Access violation

Test Case execution:

- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.server as follows:  
\*.common.hosts: \$CUI\_USR\_ROOT/ishosts
- Create an empty ishosts file with the command:  
touch \$CUI\_USR\_ROOT/ishosts
- Start a local server:
  - On Solaris: run dbhstart :1
  - On OpenVMS: run the command: 'dbh :1'
- Switch to another computer named <host>.
- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
  - Expected result: A cuitm window appears on the screen.
- Actual result:
  - From cuitm, select File->new manager & database
  - Fill in <host>:1 in the database text field.
  - Press "OK".
  - Expected result: A new cuitm with a cuitm connect dialog pops up with the message:  
'Cannot connect to database <host>:1, terminate cuitm'
- Actual result:
  - Press "OK" in the cuitm connect dialog to terminate cuitm.

Post test case execution tasks:

- Switch to the computer where the local server was started.
- On Solaris: run kill <dbh process id>
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'

ISDAT\_33\_2 Access based on IP address

Test Case execution:

- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.server as follows:  
\*.common.hosts: \$CUI\_USR\_ROOT/ishosts
- Edit the file \$CUI\_USR\_ROOT/ishosts to contain the line:  
<host>
- Start a local server:
  - On Solaris: run dbhstart :1
  - On OpenVMS: run the command: 'dbh :1'
- Switch to another computer named <host>.
- From the CSDS-UI Session Manager, start ISDAT  
(see note 2 under Test environment setup procedure)
  - Expected result: A cuitm window appears on the screen.
- Actual result:
  - From cuitm, select File->new manager & database

- Fill in <host>:1 in the database text field.
- Press "OK".
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select File->Exit

Post test case execution tasks:

- Switch to the computer where the local server was started.
- On Solaris: run kill <dbh process id>
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'

ISDAT\_33\_3 Access based on IP address and user name

Test Case execution:

- In \$CUI\_USR\_ROOT, the sub directory cfg, edit the file isdat.server as follows:
  - \*.common.hosts: \$CUI\_USR\_ROOT/ishosts
- Edit the file \$CUI\_USR\_ROOT/ishosts to contain the line:
  - <host> <user>
- Start a local server:
  - On Solaris: run dbhstart :1
  - On OpenVMS: run the command: 'dbh :1'
- Switch to another computer named <host>.
- From the CSDS-UI Session Manager, start ISDAT (see note 2 under Test environment setup procedure)
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select File->new manager & database
- Fill in <host>:1 in the database text field.
- Press "OK".
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select File->Exit

Post test case execution tasks:

- Switch to the computer where the local server was started.
- On Solaris: run kill <dbh process id>
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'

Test Procedure ISDAT\_34

Test Procedure objective:  
Verify NDC server access control.

Reference UR:  
UR 08 a/

Prerequisites:  
The test environment setup procedure has been executed.

ISDAT\_34\_1 Acceptance for unauthorized access

Test Case execution:

- From the csdsadm account, do the following:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'
- Copy the file AT34-1.stub to \$CUI\_USR\_ROOT/aca\_rights.
- Source the shell script AT34-srv.sh by typing:
  - . AT34-srv.sh
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'
- Source the shell script AT34-cli.sh by typing:
  - . AT34-cli.sh
- Start cuitm from the command line.
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select the 'Project' option menu
  - Expected result: Only the CSDS\_SP is available.
- Actual result:

Post test case execution tasks:

- From the csdsadm account:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'
- In \$CUI\_USR\_ROOT, delete the file aca\_rights by typing:
  - rm aca\_rights
- From the csdsadm account:
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'

ISDAT\_34\_2 Acceptance for authorized access

Test Case execution:

- From the csdsadm account, do the following:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'
- Copy the file AT34-2.stub to \$CUI\_USR\_ROOT/aca\_rights.
- Source the shell script AT34-srv.sh by typing:
  - . AT34-srv.sh
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'
- Source the shell script AT34-cli.sh by typing:
  - . AT34-cli.sh

- Start cuitm from the command line.
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select the 'Project' option menu
  - Expected result: Both CSDS\_SP and CSDS\_PP are available.
- Actual result:
- From cuitm, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT34-2.ctm
- From cuitm, select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference file \$USR\_ROOT/ref/AT34-2.igr
- On cuitm, press "Update"
- Expected result: A plot is shown in the cuigr panel.
- Actual result:

Post test case execution tasks:

- From the csdsadm account:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'
  - run the command: kill -STOP <dbh process id>
  - run the command 'exit'
- In \$CUI\_USR\_ROOT, delete the file aca\_rights by typing:
  - rm aca\_rights
- From the csdsadm account:
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'

Test Procedure ISDAT\_35

Test Procedure objective:

Verify NDC server access control on instrument level  
and time interval.

Reference UR:

UR 08 b/

Prerequisites:

The test environment setup procedure has been executed.

ISDAT\_35\_1 Acceptance for authorized/unauthorized access

Test Case execution:

- From the csdsadm account, do the following:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'
- Copy the file AT35-1.stub to \$CUI\_USR\_ROOT/aca\_rights.
- Source the shell script AT35-srv.sh by typing:  
. AT35-srv.sh
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'
- Source the shell script AT35-cli.sh by typing:  
. AT35-cli.sh
- Start cuitm from the command line.
  - Expected result: A cuitm window appears on the screen.
- Actual result:
- From cuitm, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT35-1.ctm
- From cuitm, select Clients->general->cuigr
- From cuigr, select File->Config->Load and select the reference  
file \$USR\_ROOT/ref/AT35-1.igr
- On cuitm, press "Update"
- Expected result: A plot is shown in the cuigr panel.
- Actual result:
- On cuitm, press the right arrow button.
- Expected result: Plot 0 database error: "access error" on  
the cuigr panel.
- Actual result:

Post test case execution tasks:

- From the csdsadm account:
- On Solaris: run dbhstop
- On OpenVMS: run the command: 'posix'  
run the command: kill -STOP <dbh process id>  
run the command 'exit'
- In \$CUI\_USR\_ROOT, delete the file aca\_rights by typing:  
rm aca\_rights
- From the csdsadm account:
- On Solaris: run dbhstart
- On OpenVMS: run the command: 'dbh'