WEC Detailed Data Analysis Software
Detailed Design
General Description

Anders Lundgren
Swedish Institute of Space Physics, Uppsala

December 10, 1999

<table>
<tr><td colspan="4">Document Status Sheet</td></tr>
<tr><td colspan="4">1. Document Title: <b>CWD DDD General Description</b></td></tr>
<tr><td colspan="4">2. Document Reference Number: <b>CWD-DDD-0001</b></td></tr>
<tr><td>3. Issue</td><td>4. Revision</td><td>5. Date</td><td>6. Reason for Change</td></tr>
<tr><td>Draft</td><td>0</td><td>94 March 21</td><td>New document</td></tr>
<tr><td></td><td>1</td><td>94 July 29</td><td>Only the general description included</td></tr>
</table>

Contents

1 Introduction

1.1 Purpose

This document describes the detailed design of the ISDAT 2.0 software. A general introduction to the ISDAT is given in [Ref. 3]. An overview of documents related to coding and general guidelines are given in [Ref. 4]. The instrument specific software modules are described in separate documents. The general design of instrument modules is described in [Ref. 5].

1.2 Scope

The scope of the ISDAT software is to provide a general tool for scientific data analysis of space science projects.

1.3 Acronyms

Acronym	Meaning
C	A programming language
CDDS	Cluster Data Disposition System
CDDS file	New notation for SHF
CDF	Common Data Format
CD-ROM	Compact Disc Read Only Memory
CoI	Co-investigator
CSDS	Cluster Science Data System
DAT	Digital Audio Tape
DB	Data Base
DEC	Digital Equipment Corporation
EFW	Electric Field and Wave Experiment
FGM	Flux Gate Magnetometer
FORTRAN	FORmula TRANslator
IRF-U	Institutet för Rymdfysik, Uppsalaavdelningen Swedish Inst. of Space Phys., Uppsala Division
ISDAT	Interactive Science Data Analysis Tool
KTH	Kungliga Tekniska Högskolan Royal Institute of Technology
NDC	National Data Centre
PI	Principal Investigator
PP	Prime Parameter
PPDB	Prime Parameter Data Base
RDM	Raw Data Medium
RFA	Request for Action
SDC	Scandinavian Data Centre
SPDB	Summary Parameter Data Base
SPL	Summary Plot
TBD	To be defined
TBW	To be written

Table 1: Acronyms

References

[1] Guide to the software detailed design and production phase. Technical Report ESA PSS-05-05, ESA, May 1992.

[2] G. Holmgren. WEC detailed data analysis software, documentation plan. Technical Report CWD-SDP-001, July 1994.

[3] G. Holmgren and A. Lundgren. ISDAT interactive scientific analysis tool. an introduction. Technical report, IRF-U, February 1994.

[4] A. Lundgren and G. Holmgren. ISDAT programmers guide. 1. overview and general guidelines. Technical Report CWD-IPG-001, IRF-U, February 1994.

[5] A. Lundgren and G. Holmgren. ISDAT programmers guide. 2. dbh instrument modules. Technical Report CWD-IPG-002, IRF-U, February 1994.

1.5 Overview

The document follows the recommendations for ESA Software projects as described in [Ref. 1]. This volume, General descriptions explains the backgrounds and sets the general standards. Separate volumes describe the specific design. An overview of available design documents is found in the documentation plan [Ref. 2].

2 Project Standards, Conventions and Procedures

2.1 Design standards

TBW

2.2 Documentation standards

All C library calls and C executable processes shall have an associated on-line man-page following UNIX standards. All other C units shall be written in a self-documentary style including a standard ASCII header with the following minimum information:

- argument specifications
- author
- Date of the verification by the technical manager.
- date and descriptions of revisions, starting after the date of the technical manager verification.

2.3 Naming conventions

The following naming conventions shall be used:

- Begin variable names by lower case, e.g. **variable**
- Indicate multi-word variables by upper case, e.g. **secondVariable**
- Begin functions by upper case, e.g. **ComputeAverage()**
- Use all-upper-case for define, e.g. **#define PI 3.14159** and underscore for multi-word names e.g. **#define PI_HALF 1.57**.

2.4 Programming standards

The ISDAT package will be coded in ANSI C..

The following standards and rules should be applied:

- All units should follow ANSI C standard
- Group families of variables into *structures* to avoid long argument lists.
- Avoid long functions. Use max one A4 page as a rule. Otherwise split up into several functions.
- Propagate errors to the top level. Never print out errors in the low level functions.
- Favour readable and logic code before the fastest possible code.
- Never use hard coded paths in the code.

2.5 Software development tools

The *imake* tool will be used for building the system.