Cluster Science Data System
Detailed Design
for the Scandinavian Data Centre
Part 1 General Description

Editor: Bengt H Nilsson
Alfvén Laboratory
Royal Institute of Technology
Stockholm

January 18, 2000

Document Status Sheet			
1. Document Title: **SDC DDD Part 1**			
2. Document Reference Number: **DS-SDC-DDD-0001**			
3. Issue	4. Revision	5. Date	6. Reason for Change
Draft	0	94 Aug 2	New document

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to describe the detailed design for the Scandinavian Data Centre (SDC) for the detailed design development phase. This document constitutes part 1, General Description. The component design specifications are described in part 2. The document is intended for the software developer.

1.2 Scope

The software will constitute one component of the CSDS system intended for production and access to Cluster databases. A detailed description of the scope of the SDC software packege is given in the SDC interface control document [Ref. 2].

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. Interface control document for the Scandinavian data centre. Technical Report DS-SDC-ID-0001, Swedish Institute of Space Physics, Uppsala Division, May 1994.

1.5 Overview

The document follows the recommendations for ESA Software projects as described in [Ref. 1]. Part 1, General descriptions explains the backgrounds and sets the general standards. Part 2, printed in a separate volume, follows the software source code organisation and describes each unit in a standardised manner.

2 Project Standards, Conventions and Procedures

2.1 Design standards

The SDC package will be coded in ANSI C and in FORTRAN 77. C is essentially used for ISDAT related software and external software, like the CSDS User Interface package. The coding conventions will be described separately for C and FORTRAN. The potential C/FORTRAN interface problems will be explicitely mentioned in connection with the individual units in part 2 of this document.

2.2 Documentation standards

2.2.1 C code

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 and descriptions of revisions, starting after the date of the SDC technical manager verification.
- Date of SDC technical manager approval.

2.2.2 FORTRAN code

TBD

2.3 Naming conventions

2.3.1 C code

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**.

2.3.2 FORTRAN code

TBD

2.4 Programming standards

2.4.1 C code

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.

2.4.2 FORTRAN code

TBD

2.5 Software development tools

2.5.1 C code

TBD

2.5.2 FORTRAN code

TBD