

CWD-DDD-001

Date: 1994 March 22

Issue: 0

Rev.: 0

Page: i

## ISDAT Detailed Design

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

November 30, 1999



---

## 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	Definitions and acronyms . . . . .	1
1.4	Overview of the document . . . . .	1
<b>2</b>	<b>Standards</b>	<b>2</b>
2.1	Design Standards . . . . .	2
2.2	Documentation Standards . . . . .	2
2.3	Naming Conventions . . . . .	2
2.4	Programming Standards . . . . .	2
2.5	Software Development Tools . . . . .	2
<b>3</b>	<b>System Design Specifications</b>	<b>3</b>
3.1	ISDAT Directory Tree . . . . .	3
<b>4</b>	<b>Directory Db Component Specifications</b>	<b>4</b>
4.1	Sub-directory lib . . . . .	4
4.2	Sub-directory server . . . . .	4
4.3	Sub-directory include . . . . .	4
<b>5</b>	<b>config Component Specifications</b>	<b>5</b>
	<b>References</b>	<b>6</b>
<b>A</b>	<b>include files</b>	<b>7</b>
A.1	File Db.h . . . . .	7
A.2	File Dbdef.h . . . . .	14
A.3	File Gi.h . . . . .	18
A.4	File Instruments.h . . . . .	19
A.5	File Is.h . . . . .	20
A.6	File IsVersion.h . . . . .	25
A.7	File Isutil.h . . . . .	25
A.8	File Ph.h . . . . .	30
A.9	File Ts.h . . . . .	31
A.10	File Ui.h . . . . .	36
A.11	File Vmalloc.h . . . . .	42

## 1 Introduction

### 1.1 Purpose of the document

This document describes the detailed design of the ISDAT 1.0 software. An example of an project specific architectural design is given in [Ref. 4]. 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. 7]. The instrument specific software modules are described in separate documents. The general design of instrument modules is described in [Ref. 8]. The Freja F4 instrument detailed design is described in [Ref. 6].

### 1.2 Scope of the software

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

### 1.3 Definitions and acronyms

Acronym	Meaning
DBH	Data Base Handler
ISDAT	Interactive Science Data Analysis Tool
N/A	Not Applicable
TBW	To Be Written

### 1.4 Overview of the document

Section 1 and 2 contain the general background. The specific software descriptions is given in section 3. Section 3 is organised after the software components and is primarily intended to be a support for the maintenance of the software.

## **2 Standards**

### **2.1 Design Standards**

The design standards and assumptions are discussed in [Ref. 3] and [Ref. 7].

### **2.2 Documentation Standards**

The ISDAT documentation follows the ESA standards [Ref. 1] in a loose manner.

### **2.3 Naming Conventions**

The naming conventions follows the recommendations given in [Ref. 7].

### **2.4 Programming Standards**

The ISDAT core software is written in C, and follows the standards described in the ISDAT general guidelines [Ref. 7].

### **2.5 Software Development Tools**

The basic development platform is a SUN Sparc station II.

## 3 System Design Specifications

### 3.1 ISDAT Directory Tree

Each sub-directory containing source code should include an Imakefile as well as a Makefile with reference to the next level in order to facilitate the possibility to automatically build the complete ISDAT from the ISDAT root level. The ISDAT directory tree consist of the following main branches:

**Db** All source code related to the DBH, including source code for the DbLib (see [Ref. 5]). The components are described in section4.

**bin** TBW

**config** Configuration files for all supported platforms. The components are described in section 5.

**doc** All on-line manual pages source code. See [Ref. 5, 11].

**include** All system level include files. See Appendix A.

**crpe** IDL interface library. See [Ref. 10].

**eiscat** EISCAT instrument specific source code. Not included in the public distribution.

**filter** Filter source code. See [Ref. 9].

**fmi** Source code related to the TS scientific library. See [Ref. 11].

**matlab** Source code related to the MatLab interface. See [Ref. 2].

**motif** Motif related source code.

**sci** Scientific analysis and display client software.

**sys** TBW

**tech** Source code for clients used for system development.

**tools** TBW

**util** TBW

**xgks** Source code related to XGKS.

## **4 Directory Db Component Specifications**

### **4.1 Sub-directory lib**

Directory lib accommodates all source code for the DbLib [Ref. 5].

### **4.2 Sub-directory server**

Directory server should accommodate all instrument specific source code. The content may vary from one installation to another.

### **4.3 Sub-directory include**

Directory *include* accommodates all instrument specific include files.

## **5 config Component Specifications**

TBW



## References

- [1] Esa software engineering standards. Technical Report ESA PSS-05-0, ESA, February 1991. Issue 2.
- [2] G. Holmgren. ISDAT ghplot users manual. Technical Report CWD-SUM-006, IRF-U, October 1993.
- [3] G. Holmgren and A. Lundgren. ISDAT interactive scientific analysis tool. an introduction. Technical report, IRF-U, February 1994.
- [4] G. Holmgren and A. Lundgren. Wec detailed data analysis software. architectural design. Technical Report CWD-ADD-001, IRF-U, February 1994.
- [5] A. Lundgren. ISDAT reference. system subroutine library. Technical Report CWD-SUM-004, IRF-U, March 1994.
- [6] A. Lundgren, E. Dackborn, and G. Holmgren. ISDAT DBH instrument module detailed design Freja instrument F4. Technical Report FRE-DDD-001, IRF-U, March 1994.
- [7] A. Lundgren and G. Holmgren. ISDAT programmers guide. 1. overview and general guidelines. Technical Report CWD-IPG-001, IRF-U, February 1994.
- [8] A. Lundgren and G. Holmgren. ISDAT programmers guide. 2. dbh instrument modules. Technical Report CWD-IPG-002, IRF-U, February 1994.
- [9] A. Lundgren and G. Holmgren. ISDAT programmers guide. 4. filters. Technical Report CWD-IPG-004, IRF-U, February 1994.
- [10] A. Lundgren and G. Holmgren. ISDAT programmers guide. 6. external software packages. Technical Report CWD-IPG-006, IRF-U, March 1994.
- [11] A. Lundgren and A. Mälkki. ISDAT reference. scientific subroutine library. Technical Report CWD-SUM-005, IRF-U, March 1994.

## A include files

All instrument specific include files reside in the sub-directory *instrument*. They are not included in this appendix. The reader is referred to the relevant instrument specific detailed design documents.

### A.1 File Db.h

```
/*
 * Db.h - Header definition and support file for the C subroutine
 * interface library (DbLib) to the ISDAT Database protocol.
 * Structures and symbols starting with "_" are private to the library.
 */
#ifndef _DB_H_
#define _DB_H_

#ifdef USG
#ifndef __TYPES__
#include <sys/types.h> /* forgot to protect it... */
#define __TYPES__
#endif /* __TYPES__ */
#else
#include <sys/types.h>
#endif /* USG */

#include "Isutil.h"
#include "Dbdef.h"

#ifndef NeedFunctionPrototypes
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)
#define NeedFunctionPrototypes 1
#else
#define NeedFunctionPrototypes 0
#endif /* __STDC__ */
#endif /* NeedFunctionPrototypes */

#ifndef NeedWidePrototypes
#if defined(NARROWPROTO)
#define NeedWidePrototypes 0
#else
#define NeedWidePrototypes 1 /* default to make interopt. easier */
#endif
#endif

#ifdef __cplusplus /* do not leave open across includes */
extern "C" { /* for C++ V2.0 */
#endif

#define Bool int
#define Status int

#define DbConnectionNumber(db) ((db)->fd)
#define DbQLength(db) ((db)->qlen)
#define DbServerVendor(db) ((db)->vendor)
#define DbProtocolVersion(db) ((db)->proto_major_version)
#define DbProtocolRevision(db) ((db)->proto_minor_version)
```

```
#define DbVendorRelease(db) ((db)->release)
#define DbDatabaseString(db) ((db)->database_name)
#define DbNextRequest(db) ((db)->request + 1)
#define DbLastKnownRequestProcessed(db) ((db)->last_request_read)

/*
 * Data structure for host setting; getting routines.
 *
 */

typedef struct {
int family; /* for example AF_DNET */
int length; /* length of address, in bytes */
char *address; /* pointer to where to find the bytes */
} DbHostAddress;

/*
 * Database datatype maintaining database specific data.
 */
typedef struct _Database {
struct _Database *next; /* next open Database on list */
int fd; /* Network socket. */
int proto_major_version; /* maj. version of server's protocol */
int proto_minor_version; /* minor version of servers protocol */
char *vendor; /* vendor of the server hardware */
    long resource_base; /* resource ID base */
long resource_mask; /* resource ID mask bits */
long resource_id; /* allocator current ID */
int resource_shift; /* allocator shift to correct bits */
DbID (*resource_alloc)(); /* allocator function */
int vnumber; /* Dblib's protocol version number. */
int release; /* release of the server */
unsigned long last_request_read; /* seq number of last event read */
unsigned long request; /* sequence number of last request. */
char *last_req; /* beginning of last request, or dummy */
char *buffer; /* Output buffer starting address. */
char *bufptr; /* Output buffer index pointer. */
char *bufmax; /* Output buffer maximum+1 address. */
unsigned max_request_size; /* maximum number 32 bit words in request*/
int (*synchronizer)(); /* Synchronization handler */
char *database_name; /* "host:database" string used on this connect*/
char *scratch_buffer; /* place to hang scratch buffer */
unsigned long scratch_length; /* length of scratch buffer */
/*
 * additional connection info
 */
unsigned long flags; /* internal connection flags */
} Database;

typedef struct _DbTimePeriod {
    IsTime start; /* start of time period */
    IsTime interval; /* length of time period */
} DbTimePeriod;

typedef struct _DbDataSpec {
    int project;
```

```
    int member;
    int instrument;
    int sensor;
    int signal;
    int channel;
    int parameter;
} DbDataSpec;

typedef struct _DbDataInfo {
    int quantity[DbMAX_DIMS]; /* out */
    float minValue[DbMAX_DIMS]; /* out */
    float maxValue[DbMAX_DIMS]; /* out */
    int scaleType[DbMAX_DIMS]; /* out */
    float samplingFreq[DbMAX_DIMS]; /* out */
    char message[64]; /* out */
} DbDataInfo;

typedef struct _DbExtraInfo {
    int dummy; /* out */
} DbExtraInfo;

typedef struct _DbSpecName {
    char project[16];
    char member[16];
    char instrument[16];
    char sensor[16];
    char signal[16];
    char channel[16];
    char parameter[16];
} DbSpecName;

typedef struct _DbContentDesc {
    DbDataSpec spec; /* in */
    int sections; /* out */
} DbContentDesc;

typedef struct _DbContentSection {
    DbDataSpec spec;
    DbTimePeriod period;
    char message[32];
} DbContentSection;

typedef struct _DbDataDesc {
    IsTime start; /* in/out */
    IsTime interval; /* in/out */
    DbDataSpec spec; /* in */
    int segments; /* out */
    int dimension; /* out */
    int samples[DbMAX_DIMS]; /* in/out */
    int units[DbMAX_DIMS]; /* in/out */
    int skip[DbMAX_DIMS]; /* in/out */
    int reduction[DbMAX_DIMS]; /* in */
    int gapFill[DbMAX_DIMS]; /* in */
    DbDataInfo info; /* out */
    DbExtraInfo extraInfo; /* out */
    unsigned int warning; /* out */
}
```

```
} DbDataDesc;
```

```
typedef struct _DbPrepareDesc {  
    IsTime start; /* in/out */  
    IsTime interval; /* in/out */  
    DbDataSpec spec; /* in */  
} DbPrepareDesc;
```

```
typedef struct _DbSearchDesc {  
    IsTime start; /* in */  
    IsTime interval; /* in */  
    DbDataSpec spec; /* in */  
    unsigned int event; /* in */  
    int sections; /* out */  
} DbSearchDesc;
```

```
typedef struct _DbSearchSection {  
    DbDataSpec spec;  
    int items;  
    DbTimePeriod *period;  
    char message[32];  
} DbSearchSection;
```

```
typedef struct _DbDataSegment {  
    IsTime start;  
    IsTime interval;  
    int offset;  
    int samples;  
} DbDataSegment;
```

```
typedef struct _DbInfoCoord {  
    int valid;  
    float x;  
    float y;  
    float z;  
} DbInfoCoord;
```

```
typedef struct _DbInfoDesc {  
    DbDataSpec spec;  
} DbInfoDesc;
```

```
typedef struct _DbInfoData {  
    int category;  
    DbInfoCoord location;  
    DbInfoCoord direction;  
} DbInfoData;
```

```
typedef struct _DbQueryDesc {  
    int mode;  
    int level;  
    IsTime time;  
    DbDataSpec spec; /* note that parameter will be unused here */  
} DbQueryDesc;
```

```
typedef struct _DbQueryData {  
    int value;
```

---

```
    int groupId;
    unsigned int event;
    char *name;
} DbQueryData;

typedef struct _DbRawStatusDesc {
    IsTime start; /* in/out */
    IsTime interval; /* in/out */
    DbDataSpec spec; /* in */
    int samples[2]; /* in/out */
    int reduction; /* in */
    unsigned int warning; /* out */
} DbRawStatusDesc;

typedef struct _DbControlDesc {
    int function;
    DbDataSpec spec;
} DbControlDesc;

typedef struct _DbLoadDesc {
    DbDataSpec spec;
    int type;
    int size;
} DbLoadDesc;

#define DbAllocID(db) ((*db)->resource_alloc)((db))

/*
 * Db function declarations.
 */
extern DbClose(
#ifdef NeedFunctionPrototypes
    Database* /* database */
#endif
);

extern DbControl(
#ifdef NeedFunctionPrototypes
    Database* /* database */,
    DbControlDesc* /* desc */,
    int /* value */
#endif
);

extern char *DbErrorString(
#ifdef NeedFunctionPrototypes
    int /* code */
#endif
);

extern DbFlush(
#ifdef NeedFunctionPrototypes
    Database* /* database */
#endif
);
```

```
extern void DbFree(  
#if NeedFunctionPrototypes  
    void* /* ptr */  
#endif  
);  
  
extern DbGetContent(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbContentDesc* /* desc */,  
    DbContentSection** /* section */  
#endif  
);  
  
extern DbGetData(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbDataDesc* /* desc */,  
    void** /* data */  
#endif  
);  
  
extern DbGetInfo(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbInfoDesc* /* desc */,  
    DbInfoData** /* info */  
#endif  
);  
  
extern DbUpload(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbLoadDesc* /* desc */,  
    unsigned char** /* buffer */  
#endif  
);  
  
extern DbGetRawStatus(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbRawStatusDesc* /* desc */,  
    unsigned char*** /* status */  
#endif  
);  
  
extern DbGetSegmentedData(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbDataDesc* /* desc */,  
    DbDataSegment** /* seg */,  
    void** /* data */  
#endif  
);  
  
extern DbGetTimeTaggedData(  
#if NeedFunctionPrototypes  
    Database* /* database */,  
    DbDataDesc* /* desc */,  
    DbDataSegment** /* seg */,  
    void** /* data */  
#endif  
);
```

```
#if NeedFunctionPrototypes
    Database* /* database */,
    DbDataDesc* /* desc */,
    DbDataSegment** /* seg */,
    void** /* data */,
    IsTime** /* time */
#endif
);

extern char *DbName(
#if NeedFunctionPrototypes
    const char* /* string */
#endif
);

extern int DbName2Spec(
#if NeedFunctionPrototypes
    Database* /* database */,
    DbSpecName* /* name */,
    DbDataSpec* /* spec */
#endif
);

extern Database *DbOpen(
#if NeedFunctionPrototypes
    const char* /* database_name */,
    int /* argc */,
    char** /* argv */
#endif
);

extern DbPrepareData(
#if NeedFunctionPrototypes
    Database* /* database */,
    DbPrepareDesc* /* desc */
#endif
);

extern DbSearch(
#if NeedFunctionPrototypes
    Database* /* database */,
    DbSearchDesc* /* desc */,
    DbSearchSection** /* section */
#endif
);

extern DbSetProtoFmt(
#if NeedFunctionPrototypes
    Database* /* database */,
    unsigned char* /* buffer */,
    int /* size */
#endif
);

extern char *DbQuantityString(
#if NeedFunctionPrototypes
```



```
        int /* quantity */
#endif
);

extern int DbQuery(
#if NeedFunctionPrototypes
    Database* /* database */,
    DbQueryDesc* /* desc */,
    DbQueryData** /* qdata */
#endif
);

extern int DbSpec2Name(
#if NeedFunctionPrototypes
    Database* /* database */,
    DbDataSpec* /* spec */,
    DbSpecName* /* name */
#endif
);

extern char *DbUnitString(
#if NeedFunctionPrototypes
    int /* unit */
#endif
);

extern char *DbWarningString(
#if NeedFunctionPrototypes
    int /* mask */
#endif
);

#ifdef __cplusplus
} /* for C++ V2.0 */
#endif

#endif /* _DB_H_ */
```

## A.2 File Dbdef.h

```
#ifndef DBDEF_H
#define DBDEF_H

#define True 1
#define False 0

#define DbPROTOCOL 1 /* current protocol version */
#define DbPROTOCOL_REVISION 0 /* current minor version */

/* Resources */

typedef unsigned long DbID;

typedef DbID DbClient;
```

```
typedef unsigned long DbMask;

/* protocol families */

#define DbFamilyInternet 0
#define DbFamilyDECnet 1
#define DbFamilyChaos 2

/*****
 * WINDOW DEFINITIONS
 *****/

/* Used in ChangeCloseDownMode */

#define DbDestroyAll          0
#define DbRetainPermanent    1
#define DbRetainTemporary    2

/*****
 * HOSTS AND CONNECTIONS
 *****/

/* for ChangeHosts */

#define DbHostInsert 0
#define DbHostDelete 1

/* for ChangeAccessControl */

#define DbEnableAccess 1
#define DbDisableAccess 0

#define DbFamilyLocal (256)

/*****
 * Database control functions - DbControl()
 *****/
#define DbCONTROL_MODE 1
#define DbCONTROL_SEQ 2
#define DbCONTROL_RELEASE 3

/* control mode bit mask values - DbCONTROL_MODE */
#define DbMODE_REALTIME 1 /* koko, remove */
#define DbMODE_BLOCK 2
#define DbMODE_SEQUENTIAL 4

/* control values - DbCONTROL_SEQ */
#define DbSEQ_NOP 0
#define DbSEQ_FIRST 1
#define DbSEQ_LAST 2
#define DbSEQ_HOLD 3
#define DbSEQ_CONT 4

/*****
 * Database load functions - DbDownload() / DbUpload()
 *****/
```

```
#define DbLOAD_TM_MAP 1

/*****
 * Database stuff
 *****/

/* maximum number of dimensions */
/* changing the constant is not enough, a lot of changes needs to be
   done in Dbproto.h as well */
#define DbMAX_DIMS 5

/* valid for type & filter */
#define DbUNDEF 0
#define DbUNUSED 0

/* project */
#define DbCLUSTER 1
#define DbCRRES 2
#define DbEISCAT 3
#define DbFAST 4
#define DbFREJA 5
#define DbGALILEO 6
#define DbGEOTAIL 7
#define DbISEE 8
#define DbMAGNETOMETER 9
#define DbPOLAR 10
#define DbPROTO 11
#define DbRIOMETER 12
#define DbSIMCLU 13
#define DbVIKING 14
#define DbTEST 15 /* to get test data */
#define DbNUM_PROJECTS 16 /* count zero even if it's not used */

/* sensor */
#define DbSTATUS 100 /* instrument specific status */
#define DbATTITUDE 101 /* instrument specific attitude */

/* units */
/* general units */
#define DbUN_TM 1
#define DbUN_CORR 2
#define DbUN_PHYS 3 /* must be last of the general units */
/* specific units (must have higher numbers than the general units) */
#define DbUN_METER 4 /* (m) */
#define DbUN_KILOMETER 5 /* (km) */
#define DbUN_M_PER_S 6 /* (m/s) */
#define DbUN_VOLT 7 /* (V) */
#define DbUN_V_PER_M 8 /* (V/m) */
#define DbUN_MV_PER_M 9 /* (mV/m) */
#define DbUN_V_PER_M_SQR_PER_HZ 10 /* (V/m)2/Hz */
#define DbUN_MV_PER_M_SQR_PER_HZ 11 /* (mV/m)2/Hz */
#define DbUN_AMPERE 12 /* (A) */
#define DbUN_MICRO_AMP 13 /* (uA) */
#define DbUN_TESLA 14 /* (T) */
#define DbUN_NANO_TESLA 15 /* (nT) */
#define DbUN_KELVIN 16 /* (K) */
```

```
#define DbUN_HZ 17 /* (Hz) */
#define DbUN_PROCENT 18 /* (%) */
#define DbUN_DECIBELL 19 /* (dB) */
#define DbUN_CELCIUS 20 /* (degC) */

/* quantity */
#define DbQTY_AMPLITUDE 1
#define DbQTY_FREQUENCY 2
#define DbQTY_POWER 3
#define DbQTY_COUNTS 4
#define DbQTY_ENERGY 5
#define DbQTY_ANGLE 6

/* reduction */
#define DbRED_NONE 0
#define DbRED_AVERAGE 1
#define DbRED_RESAMPLE 2
#define DbRED_SKIP 3
#define DbRED_MIN 4
#define DbRED_MAX 5

/* gap fill strategy (DbGetData() only) */
#define DbGAP_NAN 0
#define DbGAP_INTERPOL 1
#define DbGAP_ZERO 2
#define DbGAP_MERGE 3

/* scale values */
#define DbSCALE_LIN 0
#define DbSCALE_LOG 1

/* pack values (for internal use) */
#define DbPACK_TIMETAG 1
#define DbPACK_SEGMENT 2

/* DbGetContent() event values */
#define DbEVENT_SWEEP 1
#define DbEVENT_CALIBRATION 2
#define DbEVENT_SOUNDER 3

/* DbQuery() mode values */
#define DbQUERY_ALL 1
#define DbQUERY_ONLINE 2

/* DbGetContent(), DbGetInfo() and DbQuery() level values */
#define DbLEVEL_PROJECT 0
#define DbLEVEL_MEMBER 1
#define DbLEVEL_INSTRUMENT 2
#define DbLEVEL_SENSOR 3
#define DbLEVEL_SIGNAL 4
#define DbLEVEL_CHANNEL 5
#define DbLEVEL_PARAMETER 6

/*****
 * Database ERROR CODES
 *****/
```

```
#define DbSUCCESS          0 /* everything's okay */
#define DbBAD_DROP        1 /* must be same as DbWARN_DROP */
#define DbBAD_TIME        2 /* requested time not on disc */
#define DbBAD_PROJECT     3
#define DbBAD_MEMBER      4
#define DbBAD_EOF         5 /* must be same as DbWARN_EOF */
#define DbBAD_INSTRUMENT  6
#define DbBAD_SENSOR      7 /* mux1 or mux2 doesn't correspond to
                             requested sensor */
#define DbBAD_SIGNAL      8
#define DbBAD_CHANNEL     9
#define DbBAD_PARAMETER  10
#define DbBAD_UNITS       11
#define DbBAD_REDUCTION   12
#define DbBAD_GAPFILL     13
#define DbBAD_ALLOC       14
#define DbBAD_INTERNAL    15
#define DbBAD_GAP         16 /* must be same as DbWARN_GAP */
#define DbBAD_ZONE        17 /* requested interval is between two samples */
#define DbSUSPEND         20 /* for internal server use only */
#define DbBAD_REQUEST     21 /* bad request code */
#define DbBAD_VALUE       22 /* int parameter out of range */
#define DbBAD_ACCESS      23 /* depending on context:
    - attempt to modify the access control
    list from other than the local host.
*/
#define DbBAD_LENGTH      24 /* Request length incorrect */
#define DbNOT_IMPLEMENTED 25 /* request not implemented for specified project */
#define DbBAD_INTERVAL    26 /* invalid interval, eg. negative */

/*****
 * Database warning codes, note that we use a bitmask here.
 *****/

#define DbWARN_DROP 1 /* koko do not change!, see above */
#define DbWARN_PART 2 /* mux1 or mux2 corresponds to requested
                       sensor only some part of the time */
#define DbWARN_EOF  4 /* end of file, koko do not change!, see above */
#define DbWARN_GAP  16 /* koko do not change!, see above */

#endif /* DBDEF_H */
```

### A.3 File Gi.h

```
#ifndef GI_H
#define GI_H

#ifndef NeedFunctionPrototypes
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)
#define NeedFunctionPrototypes 1
#else
#define NeedFunctionPrototypes 0
#endif /* __STDC__ */
```

```
#endif /* NeedFunctionPrototypes */

#ifndef NeedWidePrototypes
#if defined(NARROWPROTO)
#define NeedWidePrototypes 0
#else
#define NeedWidePrototypes 1 /* default to make interprot. easier */
#endif
#endif

#ifdef __cplusplus /* do not leave open across includes */
extern "C" { /* for C++ V2.0 */
#endif

/* scale values, one of GiFIX_SCALE, GiAUTO_SCALE and GiTIME_SCALE
   must be chosen, the others can be ored in as needed */
#define GiFIX_SCALE 1
#define GiAUTO_SCALE 2
#define GiTIME_SCALE 3
#define GiMASK_SCALE 0x1f
#define GiLOG_SCALE 32

typedef struct _GiGraphAxis {
    char *label;
    char *info;
    int scale;
    float align;
    float min;
    float max;
    float *data;
    IsTime start;
    IsTime interval;
    IsTime minTime;
    IsTime maxTime;
    int drop;
} GiGraphAxis;

typedef struct _GiGraphDesc {
    char *label;
    int points;
    GiGraphAxis x;
    GiGraphAxis y;
    GiGraphAxis z;
} GiGraphDesc;

#ifdef __cplusplus
} /* for C++ V2.0 */
#endif

#endif /* GI_H_ */
```

#### A.4 File Instruments.h

```
/* Do not edit !!!*/
/* This file is generated by EditImake */
```

```
#ifndef INSTRUMENTS_H
#define INSTRUMENTS_H
#define DbHasClu
#define DbHasEis
#define DbHasFreja4
#define DbHasProto
#define DbHasRioMag
#define DbHasTest
#define DbHasVik3
#define DbHasVik4
#endif /* INSTRUMENTS_H */
```

## A.5 File Is.h

```
#ifndef IS_H
#define IS_H

/*
 * Definitions for ISDAT client - manager interaction
 */

#ifndef NeedFunctionPrototypes
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)
#define NeedFunctionPrototypes 1
#else
#define NeedFunctionPrototypes 0
#endif /* __STDC__ */
#endif /* NeedFunctionPrototypes */

#ifndef NeedWidePrototypes
#if defined(NARROWPROTO)
#define NeedWidePrototypes 0
#else
#define NeedWidePrototypes 1 /* default to make interopt. easier */
#endif
#endif

#include <Isutil.h>

#if NeedFunctionPrototypes
#include <X11/Intrinsic.h>
#endif

#ifdef __cplusplus /* do not leave open across includes */
extern "C" { /* for C++ V2.0 */
#endif

#ifdef __STDC__
typedef void* IsPointer;
#else
typedef char* IsPointer;
#endif
typedef unsigned long IsClientId;

typedef struct _IsPart *IsPart;
```

```
/* bit definitions for the IsTmInfo mode element */
#define IsTM_REALTIME 1
/* reserved: 2 */
#define IsTM_SEQUENTIAL 4
#define IsTM_CONTINUOUS 8

typedef struct _IsTmInfo { /* sent to clients by the time manager */
    int mode; /* see above */
    int project;
    int member;
    IsTime start; /* specified start time */
    IsTime interval; /* specified interval */
    IsTime contEnd; /* stop time of continuous mode, when continuous
mode is disabled it is set to start + interval */
    int infoChange; /* set if infoStart/infoInterval is changed */
    IsTime infoStart;
    IsTime infoInterval;
#ifdef IRIS4
    IsTime step; /* specified step */
#endif
} IsTmInfo;

typedef struct _IsTimeMessage { /* sent to time manager by some clients */
    IsTime start; /* specified start time */
    IsTime interval; /* specified interval */
} IsTimeMessage;

/*
 * Pipes
 */
#define IsPIPE_MAX_DIMS 5

/* type */
#define IsPIPE_FLOAT 1
#define IsPIPE_ASCII 2

typedef struct _IsPipeDesc {
    int type;
    int dimension;
    int samples[IsPIPE_MAX_DIMS];
} IsPipeDesc;

/*
 * Callbacks
 */
typedef void (*IsCallbackProc)(
#ifdef NeedFunctionPrototypes
    int, /* reason */
    IsPointer, /* closure - data the application registered */
    IsPointer /* call_data - callback specific data */
#endif
);

/*
 * Callback reasons

```



```
*/
#define IsCR_NEW_CLIENT 5
#define IsCR_TM_INFO 6
#define IsCR_CLIENTS_DONE 9
#define IsCR_SELECTIVE_REDRAW 10
#define IsCR_ERROR_MESSAGE 11
#define IsCR_FILTER_POPUP 12 /* filter only */
#define IsCR_PENDING_DATA 13 /* filter only */
#define IsCR_CHANGE_TIME 14

/*
 * Manager declarations.
 */

#define IsNOTIFY_ALL (IsClientId)0

/*
 * Function declarations in alphabetical order
 */

extern void IsAddCallback(
#if NeedFunctionPrototypes
int,
IsCallbackProc,
IsPointer
#endif
);

extern void IsCallCallbacks(
#if NeedFunctionPrototypes
int,
IsPointer
#endif
);

extern void IsCallPipe(
#if NeedFunctionPrototypes
Widget,
char *,
IsPipeDesc *,
float **
#endif
); /* client only */

extern void IsChangeTime(
#if NeedFunctionPrototypes
IsTimeMessage *
#endif
); /* client only */

extern void IsClientExec(
#if NeedFunctionPrototypes
char *
#endif
); /* manager only */
```

```
extern void IsClientNotify(  
#if NeedFunctionPrototypes  
IsClientId,  
IsTmInfo *  
#endif  
); /* manager only */  
  
extern char *IsClientPath(  
#if NeedFunctionPrototypes  
#endif  
); /* manager only */  
  
extern void IsCreateSystemMenu(  
#if NeedFunctionPrototypes  
Widget  
#endif  
);  
  
extern void IsExec(  
#if NeedFunctionPrototypes  
char *  
#endif  
); /* manager only */  
  
extern void IsFindManager(  
#if NeedFunctionPrototypes  
int,  
char **,  
Display *  
#endif  
);  
  
extern void IsSendToManager(  
#if NeedFunctionPrototypes  
char *  
#endif  
);  
  
extern void IsFilter(  
#if NeedFunctionPrototypes  
void  
#endif  
); /* filter only */  
  
extern IsTmInfo *IsGetTmInfo(  
#if NeedFunctionPrototypes  
void  
#endif  
); /* client only */  
  
extern void IsInitialize(  
#if NeedFunctionPrototypes  
int,  
char **,  
Display *  
#endif
```

```
);

extern void IsMainLoop(
#if NeedFunctionPrototypes
void
#endif
);

extern void IsManager(
#if NeedFunctionPrototypes
void
#endif
); /* manager only */

extern void IsPipeRead(
#if NeedFunctionPrototypes
IsPipeDesc *,
float **
#endif
); /* filter only */

extern void IsPipeWrite(
#if NeedFunctionPrototypes
IsPipeDesc *,
float *
#endif
); /* filter only */

extern void IsRedrawMe(
#if NeedFunctionPrototypes
void
#endif
); /* client only */

extern void IsRegisterPipe(
#if NeedFunctionPrototypes
Widget w,
char *
#endif
); /* client only */

extern void IsSetTmInfo(
#if NeedFunctionPrototypes
IsTmInfo *
#endif
); /* manager only */

extern void IsWarning(
#if NeedFunctionPrototypes
char *,
...
#endif
);

extern void IsError(
#if NeedFunctionPrototypes
```

```
char *,
...
#endif
);

#ifdef __cplusplus
} /* for C++ V2.0 */
#endif

#endif /* IS_H */
```

## A.6 File IsVersion.h

```
#ifndef IS_VERSION_H
#define IS_VERSION_H

/*
 * Definition of version, release and patch level for the
 * complete Isdat system.
 */
#define IsVERSION 1
#define IsRELEASE 0
#define IsPATCH_LEVEL 0

#endif /* IS_VERSION_H */
```

## A.7 File Isutil.h

```
#ifndef ISUTIL_H
#define ISUTIL_H

/*
 * Definitions for ISDAT utility functions
 */

#ifndef NeedFunctionPrototypes
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)
#define NeedFunctionPrototypes 1
#else
#define NeedFunctionPrototypes 0
#endif /* __STDC__ */
#endif /* NeedFunctionPrototypes */

#ifndef NeedWidePrototypes
#if defined(NARROWPROTO)
#define NeedWidePrototypes 0
#else
#define NeedWidePrototypes 1 /* default to make interrupt. easier */
#endif
#endif

#if NeedFunctionPrototypes
#include <time.h>
#endif
```

```
#include <IsVersion.h>
#include <Vmalloc.h>

#ifdef __cplusplus /* do not leave open across includes */
extern "C" { /* for C++ V2.0 */
#endif

#ifndef NULL
#define NULL 0
#endif /* NULL */

/*
 * Memory Management
 */
#define IsNew(type) ((type *) malloc((unsigned) sizeof(type)))

/*
 * Isdat Time
 */

#define IsTIME_MJD_DIFF 631152000 /* seconds between 1/1 1970 and 1/1 1950 */
#define IsSEC_PER_DAY 86400 /* seconds per day */
#define IsYMD_HMS_LEN 32
#define IsSEC_PER_STC 4.68751353e-3 /* seconds per one satellite time clock */

typedef struct _IsTime { /* define Isdat time (IsTime) */
    long s; /* seconds since January 1, 1970 */
    long ns; /* and nanoseconds */
} IsTime;

/*
 * Isdat Time declarations
 */
extern void IsAddTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern void IsAddTimeFloat(
#if NeedFunctionPrototypes
IsTime *,
double
#endif
);
extern int IsCmpTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern void IsDivTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
```

```
#endif
);
extern void IsDivTimeFloat(
#if NeedFunctionPrototypes
IsTime *,
double
#endif
);
extern IsTime IsFloat2Time(
#if NeedFunctionPrototypes
double
#endif
);
extern IsTime IsRetAddTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern IsTime IsRetDivTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern IsTime IsRetMulTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern IsTime IsRetSubTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern void IsTime2Hms(
#if NeedFunctionPrototypes
IsTime *,
char *
#endif
);
extern double IsTime2Float(
#if NeedFunctionPrototypes
IsTime *
#endif
);
extern double IsTime2Mjd(
#if NeedFunctionPrototypes
IsTime *
#endif
);
extern void IsTime2Seconds(
#if NeedFunctionPrototypes
IsTime*,
```

```
char *
#endif
);
extern unsigned int IsTime2VikStw(
#if NeedFunctionPrototypes
IsTime *
#endif
);
extern void IsTime2YmdHms(
#if NeedFunctionPrototypes
IsTime*,
char *
#endif
);
extern void IsMjd2Time(
#if NeedFunctionPrototypes
double,
IsTime *
#endif
);
extern void IsMulTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern void IsMulTimeFloat(
#if NeedFunctionPrototypes
IsTime *,
double
#endif
);
extern void IsSeconds2Time(
#if NeedFunctionPrototypes
char *,
IsTime *
#endif
);
extern void IsSubTime(
#if NeedFunctionPrototypes
IsTime *,
IsTime *
#endif
);
extern void IsSubTimeFloat(
#if NeedFunctionPrototypes
IsTime *,
double
#endif
);
extern long IsTimeGm(
#if NeedFunctionPrototypes
struct tm *
#endif
);
extern void IsVikStw2Time(
```

```
#if NeedFunctionPrototypes
int,
unsigned int,
IsTime *
#endif
);
extern void IsYmdHms2Time(
#if NeedFunctionPrototypes
char *,
IsTime *
#endif
);

/*
 * Isdat File function declaration
 */
extern void IsDumpCore(
#if NeedFunctionPrototypes
char *, /* coreName */
int /* pid */
#endif
);

extern char *IsFile(
#if NeedFunctionPrototypes
char *
#endif
);

extern char *IsGetPatchLevel(
#if NeedFunctionPrototypes
void
#endif
);

extern char *IsGetVersion(
#if NeedFunctionPrototypes
void
#endif
);

#ifdef i386
extern int isnan(
#if NeedFunctionPrototypes
double
#endif
);
#endif

#if defined(__hpux) || defined(i386) || defined(apollo) || defined(ultrix) || defined(sgi)
extern double quiet_nan(
#if NeedFunctionPrototypes
int
#endif
);
#endif
```



```
#ifdef __cplusplus
} /* for C++ V2.0 */
#endif

#endif /* ISUTIL_H */
```

## A.8 File Ph.h

```
#ifndef PH_H
#define PH_H

#include <X11/Intrinsic.h>
#include <phigs.h>

#define SI_PHIGS 1
#define SUN_PHIGS 2
#define HP_PHIGS 3

#ifdef PUESC_RASTER_RESIZE
#define PHIGS_VERSION SUN_PHIGS
#endif
#ifdef PWODSX
#define PHIGS_VERSION HP_PHIGS
#endif
#ifndef PHIGS_VERSION
#define PHIGS_VERSION SI_PHIGS /* default to X11R5 SI PHIGS */
#endif

#ifndef NeedFunctionPrototypes
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)
#define NeedFunctionPrototypes 1
#else
#define NeedFunctionPrototypes 0
#endif /* __STDC__ */
#endif /* NeedFunctionPrototypes */

#ifndef NeedWidePrototypes
#if defined(NARROWPROTO)
#define NeedWidePrototypes 0
#else
#define NeedWidePrototypes 1 /* default to make interopt. easier */
#endif
#endif

#ifdef __cplusplus /* do not leave open across includes */
extern "C" { /* for C++ V2.0 */
#endif

extern int PhOpenPhigs(
#if NeedFunctionPrototypes
Display *
#endif
);
```

```
extern int PhOpenWs(  
#if NeedFunctionPrototypes  
Pint,  
Widget  
#endif  
);
```

```
extern void PhigsProcessEvent(  
#if NeedFunctionPrototypes  
Widget,  
XtPointer,  
XEvent *,  
Boolean *  
#endif  
);
```

```
#ifdef __cplusplus  
}  
#endif
```

```
#endif /* PH_H */
```

## A.9 File Ts.h

```
#ifndef TS_H  
#define TS_H
```

```
/*  
 * Definitions for Ts  
 */
```

```
#ifndef NeedFunctionPrototypes  
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)  
#define NeedFunctionPrototypes 1  
#else  
#define NeedFunctionPrototypes 0  
#endif /* __STDC__ */  
#endif /* NeedFunctionPrototypes */
```

```
#ifndef TsSQR  
#define TsSQR(x) ((x)*(x))  
#endif
```

```
#define TsPI 3.1415926535  
#define TsTWOPI (2 * TsPI)
```

```
#define TsMAXSAMPLES 4096
```

```
#define TsPSD 0  
#define TsCORREL 1  
#define TsLOWPASS 2  
#define TsHIGHPASS 3  
#define TsBANDPASS 4  
#define TsNOTCH 5
```

```
#define TsXCOH 6
#define TsXPHASE 7
#define TsXCXY 8
#define TsXQXY 9
#define TsAUTOCORREL 10

#define TsPOWER2ERROR 1
#define TsNFFTERROR 2
#define TsSHIFTEERROR 3
#define TsNSAMPERROR 4
#define TsMEMERROR_NFREQ 5
#define TsMEMERROR_ORDER 6
#define TsMEMERROR_FSAMP 7
#define TsMEMERROR_NPOINTS 8
#define TsMEMERROR_N2POINTS 9
#define TsARMAERROR_M 10
#define TsARMAERROR_M2BIG 11
#define TsARMAERROR_IPIQ 12

#define TsCHOLESKYERROR -21

#define TsNWIND 4          /* change this when you add windows ! */
#define TsRECT 0
#define TsHAMMING 1
#define TsBLACK_HARR3 2
#define TsGAUSSIAN 3
#define TsBETA 3.5
/* etc... */

#define TsREMOVE_MEAN 1
#define TsREMOVE_TREND 2

#define TsARMA_LSMYWE 0
#define TsARMA_MAYNEFIR00ZA 1

#ifdef __cplusplus      /* do not leave open across includes */
extern "C" {           /* for C++ V2.0 */
#endif

static float ENBW[TsNWIND] = { 1.0, 1.36, 1.73, 1.90 };
/* equivalent noise bandwidths: insert for each WindType */

typedef struct _FFTStruct {
    int npoints; /* number of points in data arrays */
    int nfft; /* number of points in fft */
    int nsamp; /* number of data points in fft */
    int nshift; /* number of points to shift between ffts */
    int nseg; /* number of data segments */
    float fsamp; /* sampling frequency */
    int windowtype; /* fft data window type */
    int units; /* units: DbPHYS, DbRAW or DbTM */
    int remove; /* remove trend or mean */
} FFTStruct;

typedef struct _MEMStruct {
```

```
int order;          /* MEM estimate order (number of coefficients) */
int npoints;       /* number of points in data arrays */
int nfreq;        /* number of frequencies/psd estimates to return */
float fsamp;      /* sampling frequency */
int units;        /* units: DbPHYS, DbRAW or DbTM */
int remove;       /* remove trend or mean */
} MEMStruct;

typedef struct _ARMAstruct {
int arOrder;       /* AR estimate order (number of coefficients) */
int npsd;          /* number of points that arma returns */
int maOrder;       /* MA estimate order (number of coefficients) */
int npoints;       /* number of points in data arrays */
int m;            /* number of points in data arrays */
int estimation;   /* TsARMA_LSMYWE or TsARMA_MAYNEFIROOZA*/
int remove;       /* remove trend or mean */
} ARMAstruct;

extern void Ts1FFT(
#if NeedFunctionPrototypes
float [], int, int
#endif
);

extern void Ts2FFT(
#if NeedFunctionPrototypes
float [], float [], float [], float[], int
#endif
);

extern void TsRealFFT(
#if NeedFunctionPrototypes
float [], int, int
#endif
);

extern void TsXCorrel(
#if NeedFunctionPrototypes
float *, float *, FFTStruct *, int, float *
#endif
);

extern void TsAutoCorrel(
#if NeedFunctionPrototypes
float *, FFTStruct *, int, float *
#endif
);

extern void TsSlowCorrel(
#if NeedFunctionPrototypes
float *,float *,int ,int ,int ,float *
#endif
);

extern void TsNrCorrel(
#if NeedFunctionPrototypes
```

```
float [], float [], int ,float []
#endif
);

extern int TsErrorToString(
#if NeedFunctionPrototypes
int
#endif
);

extern int TsCheckFFT(
#if NeedFunctionPrototypes
FFTStruct *
#endif
);

extern int TsCheckMEM(
#if NeedFunctionPrototypes
MEMStruct *
#endif
);

extern int TsCheckARMA(
#if NeedFunctionPrototypes
ARMAStruct *
#endif
);

extern int TsFilterData(
#if NeedFunctionPrototypes
float *, FFTStruct *, int, float, float, float
#endif
);

extern int TsRemoveFreqs(
#if NeedFunctionPrototypes
float *, FFTStruct *, float, float, float
#endif
);

extern int TsARMAPSD(
#if NeedFunctionPrototypes
float *data, ARMAStruct *info, float *psd
#endif
);

extern int TsEvlARMA(
#if NeedFunctionPrototypes
int , int ,float *,float *,float ,int ,float *
#endif
);

extern int TsARMAAutocorr(
#if NeedFunctionPrototypes
float *,int ,int ,float *,float *
#endif
);
```

);

```
extern int TsARMACholesky(  
#if NeedFunctionPrototypes  
float *,float *,int  
#endif  
);
```

```
extern int TsLsmywe(  
#if NeedFunctionPrototypes  
float *, int, int, int, int, float *, float *, float *  
#endif  
);
```

```
extern int TsMayneFirooza(  
#if NeedFunctionPrototypes  
float *, int, int, int, int, int,float *, float *, float *  
#endif  
);
```

```
extern void TsMEMPSD(  
#if NeedFunctionPrototypes  
float [], MEMStruct *, float [], float []  
#endif  
);
```

```
extern void TsMEMCof(  
#if NeedFunctionPrototypes  
float [], int, int, float *, float []  
#endif  
);
```

```
extern float TsEvlMEM(  
#if NeedFunctionPrototypes  
float, float [], int, float  
#endif  
);
```

```
extern void TsSpect(  
#if NeedFunctionPrototypes  
float [], float [], FFTStruct *  
#endif  
);
```

```
extern void TsXSpect(  
#if NeedFunctionPrototypes  
float [], float [], float [], float [], FFTStruct *  
#endif  
);
```

```
extern void TsInitWindow(  
#if NeedFunctionPrototypes  
float [], int, int, float *  
#endif  
);
```

```
extern void TsWindowData(  
#if NeedFunctionPrototypes  
float [], float [], int  
#endif  
);  
  
#ifdef __cplusplus  
} /* for C++ V2.0 */  
#endif  
  
#endif /* TS_H */
```

## A.10 File Ui.h

```
#ifndef UI_H  
#define UI_H  
  
#include <X11/Intrinsic.h>  
  
#ifndef NeedFunctionPrototypes  
#if defined(FUNCPROTO) || defined(__STDC__) || defined(__cplusplus) || defined(c_plusplus)  
#define NeedFunctionPrototypes 1  
#else  
#define NeedFunctionPrototypes 0  
#endif /* __STDC__ */  
#endif /* NeedFunctionPrototypes */  
  
#ifndef NeedWidePrototypes  
#if defined(NARROWPROTO)  
#define NeedWidePrototypes 0  
#else  
#define NeedWidePrototypes 1 /* default to make interopt. easier */  
#endif  
#endif  
  
#ifdef __cplusplus /* do not leave open across includes */  
extern "C" { /* for C++ V2.0 */  
#endif  
  
#ifndef NULL  
#define NULL 0  
#endif /* NULL */  
  
#define UiSUCCESS          0 /* everything's okay */  
#define UiBAD_VALUE       1 /* int parameter out of range */  
#define UiBAD_PART        2 /* parameter not a valid object */  
#define UiBAD_LAYOUT      3 /* bad layout description (parse error) */  
#define UiBAD_MATCH       4 /* parameter mismatch */  
  
#define UiMalloc XtMalloc  
#define UiRealloc XtRealloc  
#define UiFree XtFree  
  
typedef char* UiPointer;  
#ifndef XtSpecificationRelease /* Xt release 3 or older */
```

```
typedef char* XtPointer;
#endif

typedef struct _UiPart *UiPart;

/*
 * definitions for UiListScroll()
 */
#define UiLIST_TOP 0
#define UiLIST_BOTTOM 1
/*
 * definitions for UiBlink()
 */
#define UiBLINK_DISABLE 0

/*
 * Memory Management
 */

#define UiNew(type) ((type *) XtMalloc((unsigned) sizeof(type)))

/*
 * Part identifier map table stuff
 */
#ifdef __STDC__
#define UiID_MAP_ENTRY(name) {#name, &name},
#else
/* koko, this may not work for every compiler but there
   is no other way to do it. */
#define UiID_MAP_ENTRY(name) {"name", &name},
#endif
#define UiID_MAP_END {0, 0}

typedef struct _UiIdentifierMap {
    char *name;
    UiPart *id;
} UiIdentifierMap;

/*
 * Callbacks
 */
typedef void (*UiCallbackProc)(
    UiPart, /* id - object identifier */
    int, /* reason */
    UiPointer, /* closure - data the application registered */
    UiPointer /* call_data - callback specific data */
);

/*
 * Callback reasons
 */
#define UiCR_BUTTON 1
#define UiCR_EDITOR_CHANGED 2
#define UiCR_POT_CHANGED 3
```



```
#define UiCR_POT_DRAG 4
#define UiCR_KEY_PRESS 7
#define UiCR_LIST_SELECTION 8
#define UiCR_FILE_SELECTION 9
#define UiCR_DIALOG_RESPONSE 10

#ifdef __STDC__
#define UiCB_MAP_ENTRY(name) {#name, name},
#else
/* koko, this may not work for every compiler but there
   is no other way to do it. */
#define UiCB_MAP_ENTRY(name) {"name", name},
#endif
#define UiCB_MAP_END {0, 0}

/*
 * Callback structures
 */
typedef struct _UiCallbackMap {
    char *name;
    void (*func)();
} UiCallbackMap;

typedef struct _UiListCBS {
    char *item;
    int position;
    int selectedCount;
    char **selectedItems;
} UiListCBS;

/*
 * Dialog responses
 */
#define UiDIALOG_NO 1
#define UiDIALOG_YES 2

/*
 * Function declarations in alphabetical order
 */

extern void UiAddCallback(
#ifdef NeedFunctionPrototypes
UiPart,
int,
UiCallbackProc,
UiPointer
#endif
);
extern void UiBlink(
#ifdef NeedFunctionPrototypes
UiPart,
int
#endif
);
extern void UiButtonState(
```

```
#if NeedFunctionPrototypes
UiPart,
int
#endif
);
extern void UiCallCallbacks(
#if NeedFunctionPrototypes
UiPart,
int,
UiPointer
#endif
);
extern UiPart UiCreateLayout(
#if NeedFunctionPrototypes
UiPart,
char *,
UiIdentifierMap *,
UiCallbackMap *
#endif
);
extern UiPart UiCreateMenu(
#if NeedFunctionPrototypes
UiPart,
char *,
UiIdentifierMap *,
UiCallbackMap *
#endif
);
extern void UiDestroyPart(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern Widget UiDrawingWidget(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern void UiEditorSetValue(
#if NeedFunctionPrototypes
UiPart,
char *
#endif
);
extern void UiError(
#if NeedFunctionPrototypes
char *,
...
#endif
);
extern void UiFileSetDir(
#if NeedFunctionPrototypes
UiPart,
char *
#endif
);
```

```
extern void UiFreePartName(  
#if NeedFunctionPrototypes  
char **  
#endif  
);  
extern char *UiGetLabel(  
#if NeedFunctionPrototypes  
UiPart  
#endif  
);  
extern char *UiGetName(  
#if NeedFunctionPrototypes  
UiPart  
#endif  
);  
extern void UiGrabInput(  
#if NeedFunctionPrototypes  
UiPart  
#endif  
);  
extern UiPart UiInformationDialog(  
#if NeedFunctionPrototypes  
UiPart,  
char *  
#endif  
);  
extern Display *UiInitialize(  
#if NeedFunctionPrototypes  
int,  
char **  
#endif  
);  
extern void UiListAddItem(  
#if NeedFunctionPrototypes  
UiPart,  
char *  
#endif  
);  
extern void UiListDeleteItem(  
#if NeedFunctionPrototypes  
UiPart,  
char *  
#endif  
);  
extern void UiListScroll(  
#if NeedFunctionPrototypes  
UiPart,  
int  
#endif  
);  
extern void UiMainLoop(  
#if NeedFunctionPrototypes  
void  
#endif  
);  
extern void UiMapLayouts(  

```

```
#if NeedFunctionPrototypes
void
#endif
);
extern void UiMapPart(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern UiPart UiNameToPart(
#if NeedFunctionPrototypes
char *,
...
#endif
);

extern void UiOptionsMenuSelect(
#if NeedFunctionPrototypes
UiPart,
int
#endif
);
extern UiPart UiParent(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern char **UiPartToName(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern Widget UiPartToWidget(
#if NeedFunctionPrototypes
UiPart
#endif
);
extern void UiPotChange(
#if NeedFunctionPrototypes
UiPart,
double
#endif
);
extern UiPart UiQuestionDialog(
#if NeedFunctionPrototypes
UiPart,
char *
#endif
);
extern UiPart UiRelativeToPart(
#if NeedFunctionPrototypes
UiPart,
int,
...
#endif
);
```

```
extern void UiSetLabel(  
#if NeedFunctionPrototypes  
UiPart,  
char *  
#endif  
);  
extern void UiSetSensitivity(  
#if NeedFunctionPrototypes  
UiPart,  
int  
#endif  
);  
extern void UiTextChange(  
#if NeedFunctionPrototypes  
UiPart,  
int,  
int,  
int,  
char *  
#endif  
);  
extern void UiTextClear(  
#if NeedFunctionPrototypes  
UiPart,  
int  
#endif  
);  
extern void UiUnmapPart(  
#if NeedFunctionPrototypes  
UiPart  
#endif  
);  
extern void UiWarning(  
#if NeedFunctionPrototypes  
char *,  
...  
#endif  
);  
extern Widget UiWidget(  
#if NeedFunctionPrototypes  
UiPart  
#endif  
);  
  
#ifdef __cplusplus  
} /* for C++ V2.0 */  
#endif  
  
#endif /* UI_H */
```

## A.11 File Vmalloc.h

```
#ifndef __Vmalloc_h  
#define __Vmalloc_h
```

---

```
/* $Id: Vmalloc.h,v 1.1 1992/08/22 19:12:35 jhd Exp jhd $ */
```

```
#ifndef NULL
#define NULL 0
#endif
```

```
#ifndef MAX_P_SIZE
#define MAX_P_SIZE sizeof(void*)
#endif
```

```
#ifdef __cplusplus
extern "C" {
#endif
```

```
#if __STDC__ || defined(__cplusplus)
void *Vmalloc(unsigned size, unsigned dim, unsigned dims[], unsigned *bytes);
void *VaVmalloc(unsigned size, unsigned dim, ...);
void Vnormalize(void *data, unsigned dim, unsigned dims[]);
void Vrelocate(void *data, unsigned dim, unsigned dims[]);
unsigned VsplICE(void *data, unsigned size, unsigned dim, unsigned dims[],
unsigned offset);
#else
void *Vmalloc();
void *VaVmalloc();
void Vnormalize();
void Vrelocate();
unsigned VsplICE();
#endif
```

```
#ifdef __cplusplus
}
#endif
```

```
#endif /* __Vmalloc_h */
```