

SDF Data Page Definitions Specification

15300018

Rev 2.05

15-April-2008



Revision History

Revision	Date	Author	Description
1.00		C. Chase	Initial Release.
1.01		C. Chase	Changed System 3000 SBP to short from unsigned short.
1.02		C. Chase	Added raw data fields
1.03		C. Chase	Revised raw data fields, added raw data config word.
1.04	3/27/2002	C. Chase	Version 3 Header Release
2.00	3/06/2007	K. Garner	Version 4 Header Release
2.01	5/18/2007	K. Garner	Add wing angle of attack and emergency switch state
2.02	6/6/2007	K. Garner	Add layback method, layback fish position, pressure sensor offset, and magnetic heading offset.
2.03	6/14/2007	K. Garner	Fix size in reserved3 field in Type3 header.
2.04	11/05/2007	K. Garner	Documentation change only. Add notes about the ping marker that is in the file when saved by SonarPro®. Add tpuSwVersion and capabilityMask to SDF header.
2.05	04/15/2008	D. Clapp / K. Garner	Added flag for txVersion, and numSamplesExtra for MF processing. Added system capabilities bit to indicate towfish is a 5000 V2.

Table of Contents

Revision History	2
Table of Contents	3
1. Introduction	4
2. Description	4
2.1. <i>Header</i>	4
2.2. <i>Data Format</i>	7
2.2.1. <i>System 3000</i>	7
2.2.2. <i>System 5000</i>	8
2.2.3. <i>System 7000</i>	10

1. Introduction

This document describes the data page formats for the System 3000 (NGS), System 5000 and System 7000.

2. Description

Each system has a unique data page structure as defined by the pageVersion field in the header. The header is ostensibly the same for each page structure but the data portion of a data page is unique. The pageVersion field can have the following values:

Table 1 – Valid pageVersion values

Towfish Type	Header version	pageVersion value
3000	3	3000
3000	4	3001
5000	3	5000
5000	4	5001
7000	3	7000
7000	4	7001

The data page structure is the data page produced by the System’s Transceiver and Processing Unit (TPU). See the section titled **SDF Files** for how these individual data pages are concatenated into “.sdf” files.

2.1. Header

The header is a structure that contains information about the current ping. Information such as position, time, speed and other pertinent data is in the header prior to the channel data. The header is defined as follows:

```
typedef struct
{
    U32 numberBytes;          /* total number of bytes in page */
    U32 pageVersion;
    U32 configuration;
    U32 pingNumber;
    U32 numSamples;         /* count of samples in processed side-scan data,
                           * if processed data exists in page, or count * *
                           * of samples in rawdata channels if no
                           * processed data exists. The difference is
                           * stored in numSamplesExtra (type3 hdr).
                           */
    U32 beamsToDisplay;
    U32 errorFlags;
    U32 range;
    U32 speedFish;
    U32 speedSound;
    U32 resMode;
```

```

U32 txWaveform;
U32 respDiv;
U32 respFreq;
U32 manualSpeedSwitch;
U32 despeckleSwitch;
U32 speedFilterSwitch;

U32 year;
U32 month;
U32 day;
U32 hour;
U32 minute;
U32 second;
U32 hSecond;
U32 fixTimeHour;
U32 fixTimeMinute;
float fixTimeSecond;

float heading;          /* heading from towfish compass */
float pitch;            /* pitch from compass */
float roll;             /* roll from compass */
float depth;            /* from towfish */
float altitude;         /* from towfish */
float temperature;     /* from towfish */
float speed;            /* from GPS, updated on GPS update,m/s */
float shipHeading;     /* from GPS - Course Over Ground*/
float magneticVariation; /* from GPS */

double shipLat;        /* from GPS, radians */
double shipLon;        /* from GPS, radians */
double fishLat;        /* rads */
double fishLon;        /* rads */

/* Added at version 3 */

U32 tvgPage;
U32 headerSize;       /* number of bytes in header */

U32 fixTimeYear;
U32 fixTimeMonth;
U32 fixTimeDay;

float auxPitch;        /* aux data from AUV or other sensors */
float auxRoll;
float auxDepth;
float auxAlt;

float cableOut;

float fseconds;        /* fractional seconds */

U32 altimeter;        /* altimeter off|on */

U32 sampleFreq;

U32 depressorType;
U32 cableType;

```

```

F32 shieveXoff;
F32 shieveYoff;
F32 shieveZoff;
F32 GPSheight;

U32 rawDataConfig; /* upper word = stbd config, lower = port */

/* Added at version 4 */
/* Size of this header addition should stay 256 bytes (64 U32s) */
U32 header3ExtensionSize; /* Size of only this header
* extension. Must be equal to 256
* bytes. */

U32 sbpTxWaveform; /* Tx waveform for Sub Bottom
* Profiler (SBP) */
U32 sbpPreAmpGain; /* 0 = low 1 = high */
U32 sbpDataRaw; /* 0 = Processed SBP data, 1 = Raw
* SBP data */
U32 sbpNumSamples; /* Number of SBP samples in data
* page. May be different from
* Side Scan */
U32 sbpSampleFreq; /* Sample frequency of SBP channel */
U32 sbpTxWaveformVersion /* Tx waveform version of the SBP
* Tx board. */

float wingAngle; /* Angle of actuated wing in
* degrees */
U32 emergencySwitchState; /* State of system emergency switch
* - 1 = on, 0 = off */
/* Layback position parameters set by SonarPro. Set to zero by
TPU. */
U32 laybackMethod; /* Method used to calculate
layback. 0 = Pythagorean
theorem. */
double laybackFishLat; /* fish latitude as determined by
layback calculation (in rads) */
double laybackFishLon; /* fish longitude as determined by
layback calculation (in rads) */

float fishHeadingOffset; /* Magnetic heading offset applied
to towfish heading, in degrees*/
float pressureSensorOffset; /* psi */

/* Added at TPU s/w version 6.13 */
U32 tpuSwVersion; /* Version of the TPU s/w,
0xVVNNMMDD where
VV = Major Version Number
NN = Minor Version Number
MM = Month
DD = Day

/* Added at vxWorks version 6.17.*/
U32 capabilityMask; /* Bit mask defining various system
*capabilities.
* Bit 0 = Configured for raw data (System
* 5000 only)
* Bit 1 = Configured for actuated wing
* Bit 2 = Configured with Sub Bottom

```

```

        * Profiler option
        * Bit 3 = Configured with header ver. 4
        * Bit 4 = Configured to allow a single
        * oversampled frequency (3000 only)
        * Bit 5 = Configured to allow dual
        * frequency operation (3000 only)
        * Added at tpuSwVersion 6.17.
        * Bit 6 = 5000 (V1 or V2) system with
        * ver. 2 Demux
        * Bit 7 = 5000 V2 towfish
        */
/* Added at vxWorks version V6.19 */
    U32 txVersion;    /* TPU Transmitter Version 0
        * = 5000, 1 = 5000 V2
        */
/* Added at vxWorks version V6.22 */
    U32 numSamplesExtra; /* The extra number of samples included
        * in each data channel to account for
        * chirp Tx waveforms. This value is zero
        * for other waveforms.
        */

/* Reserved data portion.  Array size set to keep this header addition
at 256 bytes. */
    U32 reserved3[44];

} SYS5000HEADER;

```

Note: The original version of the header (pageVersion < 3) had 44 words. The version 3 header contains 64 words. The version 4 header contains 128 words. The added words are noted in the comments.

2.2. Data Format

The header is followed by a unique data structure for each system. The data structure is also flexible in that the first two bytes of each channel indicate the number of samples for that channel (Note: The exception is the System 3000 with the Sub Bottom Profiler utilizes 4 bytes for the number of samples for the Sub Bottom channel). For example, if a channel were defined as:

```
    unsigned short channel1[];
```

The first two bytes indicate the number of unsigned short data samples that follow. The arrays are therefore variable size and form a type of link-list (via the size information in the first two bytes).

2.2.1. System 3000

The System 3000 Version 4 data channel structure is as follows:

```
    unsigned short port1f[];    /* Port low-freq */
    unsigned short stbdlf [];   /* Stbd low-freq */

```

```

unsigned short porthf[];      /* Port high-freq */
unsigned short stbdhf[];     /* Stbd high-freq */
long sbp[];                  /* Sub-bottom profiler */

```

Each of the side scan data channel samples is a 16-bit unsigned quantity. Each of the Sub-bottom channel samples is a 32-bit signed quantity.

The System 3000 Version 3 data channel structure is as follows:

```

unsigned short portlf[];     /* Port low-freq */
unsigned short stbdlf[];    /* Stbd low-freq */
unsigned short porthf[];    /* Port high-freq */
unsigned short stbdhf[];    /* Stbd high-freq */
short sbp[];                /* Sub-bottom profiler */

```

Each of the data channel samples is a 16-bit unsigned quantity.

Note: The “configuration” word in the data page header may be examined to determine the type of data in the data page. The “configuration” word is a bit mask that may contain a logical OR of the values in [Table 2](#).

Table 2 – Data page header “configuration” word bit mask values

Bit Value	Meaning
0x03	Data page contains Low Frequency Side Scan Data when set
0x0C	Data page contains High Frequency Side Scan Data when set
0x10	Data page contains SBP data when set

2.2.2. System 5000

The System 5000 data channel structure is as follows:

```

unsigned short chan1Data[]; /* processed side-scan data */
unsigned short chan2Data[];
unsigned short chan3Data[];
unsigned short chan4Data[];
unsigned short chan5Data[];
unsigned short chan6Data[];
unsigned short chan7Data[];
unsigned short chan8Data[];
unsigned short chan9Data[];
unsigned short chan10Data[];
short bathyPort1i[];       /* Bathy 1 synthesized from */
short bathyPort1q[];       /* element data */
short bathyPort2i[];       /* Bathy 2,3 raw from arrays */
short bathyPort2q[];
short bathyPort3i[];
short bathyPort3q[];
short bathyStbd1i[];
short bathyStbd1q[];
short bathyStbd2i[];
short bathyStbd2q[];

```

```

short bathyStbd3i[];
short bathyStbd3q[];
short echo1[];
short echo2[];
short subBottom1[];
short subBottom2[];
short rollSensor[];
short yawRate[];
short rawdataPort1i[]; /* 1-12 = port raw side-scan */
short rawdataPort1q[];
short rawdataPort2i[];
short rawdataPort2q[];
short rawdataPort3i[];
short rawdataPort3q[];
short rawdataPort4i[];
short rawdataPort4q[];
short rawdataPort5i[];
short rawdataPort5q[];
short rawdataPort6i[];
short rawdataPort6q[];
short rawdataPort7i[];
short rawdataPort7q[];
short rawdataPort8i[];
short rawdataPort8q[];
short rawdataPort9i[];
short rawdataPort9q[];
short rawdataPort10i[];
short rawdataPort10q[];
short rawdataPort11i[];
short rawdataPort11q[];
short rawdataPort12i[];
short rawdataPort12q[];
short rawdataPort13i[]; /* bathy string #2 */
short rawdataPort13q[];
short rawdataPort14i[];
short rawdataPort14q[]; /* bathy string #3 */
short rawdataStbd1i[]; /* 1-12 = starboard raw side-scan */
short rawdataStbd1q[];
short rawdataStbd2i[];
short rawdataStbd2q[];
short rawdataStbd3i[];
short rawdataStbd3q[];
short rawdataStbd4i[];
short rawdataStbd4q[];
short rawdataStbd5i[];
short rawdataStbd5q[];
short rawdataStbd6i[];
short rawdataStbd6q[];
short rawdataStbd7i[];
short rawdataStbd7q[];
short rawdataStbd8i[];
short rawdataStbd8q[];
short rawdataStbd9i[];
short rawdataStbd9q[];
short rawdataStbd10i[];
short rawdataStbd10q[];
short rawdataStbd11i[];

```

```
short rawdataStbd11q[];
short rawdataStbd12i[];
short rawdataStbd12q[];
short rawdataStbd13i[];
short rawdataStbd13q[];
short rawdataStbd14i[];
short rawdataStbd14q[];
```

2.2.3. System 7000

The System 7000 data channel structure is tentatively defined as follows:

```
unsigned short chan1Data[];
unsigned short chan2Data[];
unsigned short chan3Data[];
unsigned short chan4Data[];
unsigned short chan5Data[];
unsigned short chan6Data[];
unsigned short chan7Data[];
unsigned short chan8Data[];
unsigned short chan9Data[];
unsigned short chan10Data[];
unsigned short chan11Data[];
unsigned short chan12Data[];
unsigned short chan13Data[];
unsigned short chan14Data[];
unsigned short chan15Data[];
unsigned short chan16Data[];
unsigned short chan17Data[];
unsigned short chan18Data[];
unsigned short chan19Data[];
unsigned short chan20Data[];
unsigned short sector1Data[];
unsigned short sector2Data[];
unsigned short sector3Data[];
unsigned short sector4Data[];
unsigned short sector5Data[];
unsigned short sector6Data[];
unsigned short sector7Data[];
unsigned short sector8Data[];
short bathy1[];
short bathy2[];
short bathy3[];
short bathy4[];
short bathy5[];
short bathy6[];
short bathy7[];
short bathy8[];
short bathy9[];
short bathy10[];
short bathy11[];
short bathy12[];
short bathy13[];
short bathy14[];
short bathy15[];
short bathy16[];
```

```
short bathy17[];
short bathy18[];
short bathy19[];
short bathy20[];
short bathy21[];
short bathy22[];
short bathy23[];
short bathy24[];
unsigned short echo1[];
unsigned short echo2[];
unsigned short echo3[];
unsigned short echo4[];
unsigned short echo5[];
unsigned short echo6[];
unsigned short echo7[];
unsigned short echo8[];
short sensor1[];
short sensor2[];
short sensor3[];
short sensor4[];
```

3. SDF Files

The SonarPro® generated Sonar Data Files (SDF) have a “.sdf” file extensions. These files are in the form; [data page][data page] ... etc – where each data page is the ping marker followed by the SDF data page as described in the section titled **Description**. The ping marker is a 32-bit [value that](#) never changes and is equal to 0xFFFFFFFF ($2^{32}-1$).