ECA GROUP		
eXtented Triton Format (XTF)		
FILE FORMAT SPECIFICATION REV.42		
ORIGIN **BES**	**DOCUMENT N°** **2511-8-02-003**	**ISSUE** **00**

NB. OF PAGES **42**	**WRITTEN BY: KERMARREC Elodie** **DATE: 25/10/2019**	**COPY** **N° /**

DOCUMENT FOLLOW-UP

Issue	Date	Author	PURPOSE OF THE MODIFICATION
00	10/12/2019	EKE	Document redesign. Simplification of the XTF format description. Modification of the manufacturer custom information management.

	Name/Title	Date	Signature
Written by:	KERMARREC Elodie Studies Engineer	10/12/2019	
Checked by:	SAMAN Ludovic Software Engineer	29/01/2020	
Approved by:	TAUVRY Sebastien Marine Data Processing Manager	28/02/2020	

REV	DESCRIPTION OF CHANGE	DATE	BY	APP
X1	First Draft	15/01/2002		RLC
X2	Update References	30/01/2002	RLC	
X3	Add Reference to Read_xtf.c and demo_xtf.c	12/02/2002	RLC	
X4	Add reference to MillivoltScale.	29/03/2002	RLC	
X5	Update structure size for XTFPINGHEADER	09/04/2002	RLC	
X6	Added XTFHIGHSPEEDSENSOR structure and updated header types for XTFPINGHEADER.	31/05/2002	RLC	
X7	Added ISISFORWARDBEAMHEADER and XTFBEAMXYZA Structures.	03/06/2002		
X8	Update FileHeader's SonarType. Added XTF_BATHY_SNIPPET data format and SNP0, SNP1 structures associated with the XTF_BATHY_SNIPPET packet type. Updated description of XTFPINGCHANHEADER.	12/08/2002	RLC	
X9	Reviewed an edited for accuracy	20/08/2002	RS	RS
X10	Remove Read_XTF.c reference.	24/09/2002	RLC	RLC
X11	XTFPINGHEADER/XTFBATHHEADER, HIGHSPEEDSENSOR, XTFBEAMXYZA offset listings were incorrect,updated to display correct offsets.	17/10/2002	RLC	RLC
X12	Added XTF_SARA_CAATI_HEADER packet description Updated XTFATTITUDEDATA structure to include new fields, new packet types XTF_HEADER_KLEIN3000_DATA_PAGE, XTF_HEADER_POS_RAW_NAVIGATION	24/03/2003	RLC	RLC
X13	Added section 2.3.1, Odd-numbered sidescan sonar channels Corrected the EventNumber byte offset in the XTFPINGHEADER structure (deleted the CurrentLineID field)	27/04/2004	LCS	LCS
X14	Further update to EventNumber and explanation	20/09/2004	GVS	GVS
X15	Added CODA Echoscope	02/01/2005	GVS	GVS
X16	Added CODA Echoscope Config (and corrected name)	14/02/2005	GVS	GVS
X17	Added QPS data records for single beam echosounders and multi-transducer echosounders	02/01/2006	DDB	DDB
X18	Added Benthos C3D, Edgetech 4200, Benthos SIS1624, C-MAX, Edgetech MP-X . Modified XTFATTITUDE Reserved3[10] to Reserved3[1]. Modified XTFPINGCHANHEADER to include a WeightFactor field in bytes 58 and 59.	19/04/2006	GVS	GVS
X19	Added Reson 7125	06/06/2006	GVS	GVS
X20	Added Kongsberg SAS; corrected weighting data type	06/10/2006	GVS	GVS
X21	Corrected size of usAmpl in XTFBEAMXYZA structure	27/12/2006	GVS	GVS
X22	Add 32bit logging capability (CHANINFO Byte74) and Klein v4 Header Type 108 in XTFPINGHEADER	10/01/2006	GVS	GVS
X23	Added CODA Echoscope Image HeaderType = 72	22/06/2007	GVS	GVS
X24	Added XTFRAWCUSTOMHEADER	22/06/2007	GVS	GVS
X25	Added XTF_HEADER_Q_MULTIBEAM structure	30/06/2008	GVS	GVS
X26	Corrected description of WORD Microseconds	18/12/2008	GVS	GVS
X27	Added various new sonar types inc QINSy R2Sonics, C-Max, GeoAcoustics	29/09/2010	GVS	GVS
X28	Updates to Type 3 Attitude packet and addition of type 42 navigation and type 84 gyro packets	29/09/2010	GVS	GVS
X29	Updates to R2Sonic sonar types for QINSy and Triton	07/01/2010	GVS	GVS
X30	Added Klein 3500, 5900 and Edgetech 4600	10/03/2011	GVS	GVS
X31	Added Reson Type 76 – 7027 packet	10/04/2011	GVS	GVS

X32	Remove Reson Type 76 – 7027 packet	02/12/2011	GVS	GVS
X33	Add Appendix for Recon – 7nnn packets	24/12/2011	GVS	GVS
X34	Added cable out hundredths	21/02/2012	GVS	GVS
X35	Changes to CHANINFO remove latency adjust byte count	04/02/2012	GVS	GVS
X36	Added Reson 7018 Watercolumn and other data packets	10/07/2014	GVS	GVS
X37	Added R2Sonic Watercolumn and other data packets	14/10/2014	GVS	GVS
X38	Added DT-100 and Kraken Sonar Types	17/09/2015	GVS	GVS
X39	Added EM2040 and Klein5Kv2 Sonar Types	17/09/2015	GVS	GVS
X40	Added FSI and K4900 Sonar Types	22/09/2015	GVS	GVS
X41	Re-introduced SampleFormat to CHANINFO Added Reserved PacketID table to XTFRAWCUSTOMHEADER section. Kraken sonar type 62 deletion (redundant with Kraken sonar type 65)	16/09/2016	LSA	SET
X42	Update XTFFILEHEADER and XTFPINGHEADER structures Updated SonarType and HeaderType lists	11/01/2018 10/12/2019	LSA / EKE	SET

Table of contents

List of Illustrations

List of Tables

1. Introduction

1.1. Purpose

This document is intended to address file format and suggested ways to process XTF files.

1.2. What's new

ECA Group has decided to maintain the XTF format and to pursue the publication of the XTF format specification for all the XTF community.

This V42 revision is a document available on the ECA Group website: https://www.ecagroup.com/en/xtf-file-format and updates will be published regularly.

The major changes of this new revision are as follows:

- *Redesign of the document.* The document had been modernized and simplified to allow a better understanding of the XTF format. The document history will be still available in the V41 revision.
- *More flexibility for the integration of specific data from sonar manufacturer or software developer. "*SonarType" (XTFFILEHEADER) becomes "SensorsType". The current code remains unchanged but a range of 100 new code start from 1000 will be allocated for each major XTF community member who asks for it. For punctual need, a code from 70 to 999 would be allocated. HeaderType is kept in its definition with the free of use of the range 200-255 according to the previously defined "SensorsType". The HeaderType unused ranges before 200 are temporary frozen. Those ranges may be allocated if several manufacturers ask for it.

For specific need, addition or update of the XTF format specification document, please contact us by e-mail: xtf-support@ecagroup.com.

1.3. Definitions, abbreviations, and acronyms.

Table 1 : Acronyms used

APP	Approved by
CTD	Conductivity Temperature Depth
EOF	End Of File
MRU	Motion Reference Unit
REV	Revision
RTK	Real Time Kinematic
XTF	eXtended Triton Format

1.4. Reference

Speed of sound in seawater at high pressures. *J. Acoust. Soc. Am.*, **62** (5), 1129-1135.). Chen Millero formula. (C. T. Chen and F. J. Millero, 1977)

2. Overall Description

2.1. FORMAT PERSPECTIVE

The XTF file format (e**X**tended **T**riton **F**ormat) was created to answer the need for saving many different types of sonar, navigation, telemetry and bathymetry information. The format can easily be extended to include new types of data that may be encountered in the future.

2.2. TECHNICAL SPECIFICATIONS

2.2.1. Binary Data Representation

Except for some bathymetry data (which is logged “raw”), all data is written with Intel 80x86 byte ordering (LSB to MSB). If an XTF file is to be processed on a non-Intel computer such as one from Sun Microsystems, Inc., Silicon Graphics, Inc., or Apple Computer, Inc., the order of the bytes in all values must be exactly reversed. For example, a float value (4 bytes) would need to be reordered from (1,2,3,4) to (4,3,2,1) in the target machine’s memory before treating the number as a floating-point value. This effectively converts the value from little-endian (least-significant byte first) to big-endian (most-significant byte first).

2.2.2. Data Types

All sizes/formats given in this document are as follows. All data types are signed unless otherwise specified.

Table 2 : Data representation types for XTF header and data packets

Data Type	Microsoft® Data Type	Bytes	Range of Values
char	char	1	-128 to 127
short	short	2	-32,768 to 32767
int	int	*	Standard is 4 bytes but number of bytes is system dependent for a 32-bit OS. Range for a 32bit signed int (–2,147,483,648 to 2,147,483,647)
long	long	4	(–2,147,483,648 to 2,147,483,647)
float	float	4	3.4E +/- 38 (7 digits)
double	double	8	1.7E +/- 308 (15 digits)
	BYTE	1	Unsigned integer (0 to 255)
	WORD	2	Unsigned integer (0 to 65,535)
	DWORD	4	Unsigned integer (0 to 4,294,967,295)
Hex	Hexadecimal	0x0	“x” represents a value in Hexadecimal.

Descriptions for the fields are labeled with keys to indicate value status of the field. The status keys are shown in the table below.

Table 3 : Field status types

Status	Description
M	Mandatory (must be filled in or set to a default value)
R[=value]	Recommended input (set to value if not used, or if no value given set to 0)
O[=value]	Optional (set to value if not used, if not value given set to 0)
U	Unused. Reserved for future use

2.2.3. Note to programmers

When using the structures described in this document, note that the packing should be 1. In the Microsoft Visual

C++ compiler, the statement

#pragma pack(1)

should be placed before the structure definitions and

#pragma pack()

after the definitions (or equivalent). By default, Microsoft compilers use a packing of eight, which will result in different structure alignment than described in this document.

All structures should be zero-filled before use. Unused values should remain zero.

3. XTF Format Description

3.1. Methodology

An XTF file can be thought of as a “pool” of data. If you use XTF to collect data during a survey, you can add data to the file at any time without needing to synchronize your data packets.

For example, bathymetry data may be logged five times per second while sonar data is being logged at 10 times per second. No storage space is wasted and no “holes” are created in the saved data stream. While processing an XTF file, the processing software can easily ignore unknown or unnecessary data packets. Any software that reads XTF files should also ignore unnecessary packets because it guarantees compatibility with files that may contain new kinds of data that may be included in the future.

Some users may think that the XTF file format frequently changes. That thought comes from a basic misunderstanding of the XTF methodology. As new kinds of sensors are introduced into the marketplace, new XTF packet types are created to store the unique data produced by those sensors. Those packets may not be recognized by legacy software programs, but those programs should be written to benignly skip over unrecognized XTF packets.

Since the pool of data in an XTF file is written asynchronously, it is impossible to calculate a byte offset for a specific record in the file. However, there is a straightforward method to quickly search a file for any specific data packet. This method is described later in this appendix.

3.2. General Description

Data stored in an XTF file uses a general message format. Each XTF file begins with a file header record and is followed by one or more data packets. The file header data is stored in the XTFFILEHEADER structure. Each XTFFILEHEADER contains room for six channels. Channel data is stored in the CHANINFO structure.

Note: A “channel” in XTF is generated from a “ping.” Basic sidescan sonars are two channels. Dual-frequency sidescan sonars are four channels. A single bathymetry system is a single channel. Speed sensors, altimeters, or any other sensor that outputs data as a single numeric value (typically over a serial port) is NOT considered a channel in XTF. This kind of numeric data is entered into the system and stored in dedicated fields within the XTF files.

The basic XTF file header record is 1024 bytes in size. It can be larger than 1024 bytes when the total number of channels to be stored in the file is greater than six. In this event, the total size of the file header record grows in increments of 1024 bytes until there is enough room to hold all of the CHANINFO structures.

To simplify the XTF reading, the XTF data packets can be padded so that the total packet size is a multiple of 64 bytes, but this is not a requirement.

Two important elements of the file header are:

- Number of sonar channels
- Number of bathymetry channels

These are used to determine how many CHANINFO structures will be in the header record. The CHANINFO structures for all of the sonar channels will always precede the structures for the bathymetry channels.
Except where otherwise documented, all values are stored using the metric system (typically meters) or degrees of angle.

3.2.1. Odd-numbered sidescan sonar channels

For odd-numbered channels, the sample order is reversed. This is done so that the channels will display in a conventional manner in the waterfall window. When channels are selected as sub-bottom, the sample order is not reversed.

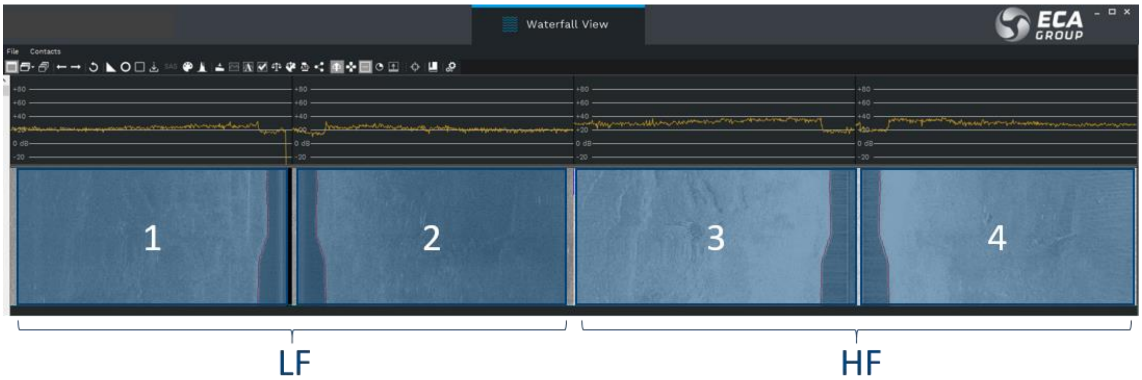
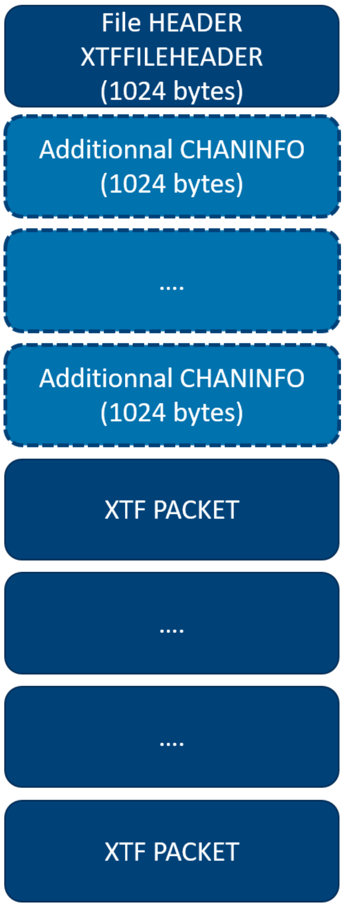


Figure 1 : Waterfall window – channels display

3.3. XTF FILE DATA LAYOUT

The file header is the first data in the file. Depending on total number of sonar and bathy channels, CHANINFO structures may follow the file header. After the File Header and possible CHANINFO structures, data packets follow until the end of the file.



XTFFILEHEADER is described in 3.3.1 and in 3.3.2

CHANINFO is described in 3.3.3

XTF PACKET is described in part 4

Figure 2 : XTF File Data Layout

3.3.1. XTF File Header Layout

The XTF File header structure is described in: Table 4. The size is 1024 bytes. If more than six channels of data are to be logged in the XTF file, then the header can grow in increments of 1024 bytes to allow for additional CHANINFO structures are required. The CHANINFO structure is described in Table 5.

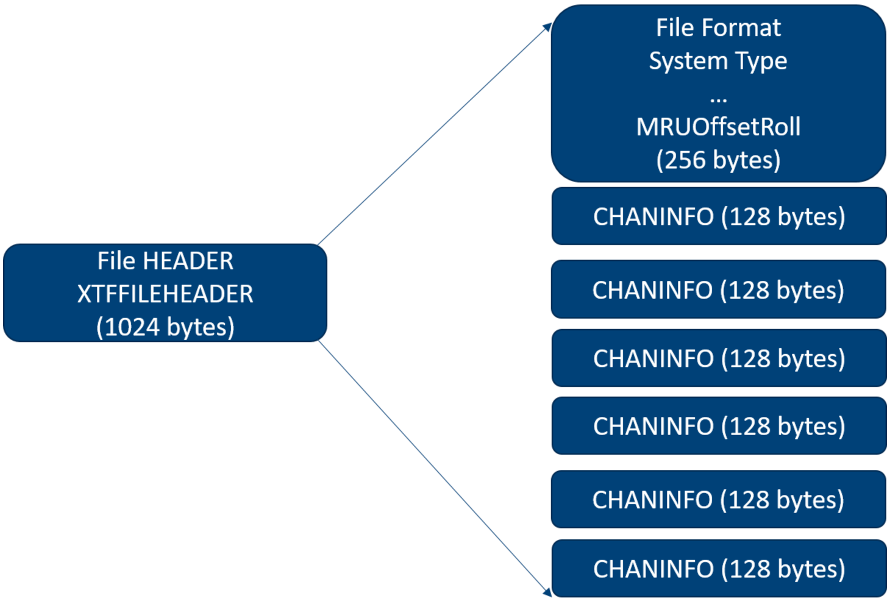


Figure 3 : XTF File Header data layout

3.3.2. XTFFILEHEADER Structure

Table 4 : XTFFILEHEADER structure

XTFFILEHEADER				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
FileFormat	BYTE	0	M	Set to 123 (0x7B)
SystemType	BYTE	1	M	Set to 1
RecordingProgramName	char[8]	2	M	Example: "ECA_REC"
RecordingProgramVersion	char[8]	10	M	Example: "556" for version 5.56
SonarName	char[16]	18	R	Name of server used to access sonar. Example: "C31_SERV.EXE"
SensorsType	WORD	34	M	0=NONE,default. 1 = JAMSTEC, Jamstec chirp 2-channel subbottom. 2 = ANALOG_C31, PC31 8-channel. 3 = SIS1000, Chirp SIS-1000 sonar. 4 = ANALOG_32CHAN, Spectrum with 32-channel DSPlink card. 5 = KLEIN2000, Klein system 2000 with digital interface. 6 = RWS, Standard PC31 analog with special nav code. 7 = DF1000, EG&G DF1000 digital interface. 8 = SEABAT, Reson SEABAT 900x analog/serial. 9 = KLEIN595, 4-chan Klein 595, same as

				ANALOG_C31. 10 = EGG260, 2-channel EGG260, same as ANALOG_C31. 11 = SONATECH_DDS, Sonatech Diver Detection System on Spectrum DSP32C. 12 = ECHOSCAN, Odom EchoScanII multibeam (with simultaneous analog sidescan). 13 = ELAC, Elac multibeam system. 14 = KLEIN5000, Klein system 5000 with digital interface. 15 = Reson Seabat 8101. 16 = Imagenex model 858. 17 = USN SILOS with 3-channel analog. 18 = Sonatech Super-high res sidescan sonar. 19 = Delph AU32 Analog input (2 channel).. 20 = Generic sonar using the memory-mapped file interface. 21 = Simrad SM2000 Multibeam Echo Sounder. 22 = Standard multimedia audio. 23 = Edgetech (EG&G) ACI card for 260 sonar through PC31 card. 24 = Edgetech Black Box. 25 = Fugro deeptow. 26 = C&C's Edgetech Chirp conversion program. 27 = DTI SAS Synthetic Aperture processor (memmap file). 28 = Fugro's Osiris AUV Sidescan data. 29 = Fugro's Osiris AUV Multibeam data. 30 = Geoacoustics SLS. 31 = Simrad EM2000/EM3000. 32 = Klein system 3000. 33 = SHRSSS Chirp system 34 = Benthos C3D SARA/CAATI 35 = Edgetech MP-X 36 = CMAX 37 = Benthos sis1624 38 = Edgetech 4200 39 = Benthos SIS1500 40 = Benthos SIS1502 41 = Benthos SIS3000 42 = Benthos SIS7000 43 = DF1000 DCU 44 = NONE_SIDESCAN 45 = NONE_MULTIBEAM 46 = Reson 7125 47 = CODA Echoscope 48 = Kongsberg SAS 49 = QINSy 50 = GeoAcoustics DSSS 51 = CMAX_USB 52 = SwathPlus Bathy

				53 = R2Sonic QINSy 54 = Converted SwathPlus Bathy 55 = R2Sonic Triton 56 = Edgetech 4600 57 = Klein 3500 58 = Klein 5900 59 = EM2040 60 = Klein5Kv2 61 = DT100 64 = ECA navigation 65 = Kraken 66 = Klein 4900 67 = FSI HMS622 68 = FSI HMS6x4 69 = FSI HMS6x5 1000-1099 = ECA sensors 1100-1199 = IxBlue sensors 1200-1299 = QPS sensors 1300-1399 = Chesapeake sensors 1400-1499 = R2Sonics sensors If available for distribution, manufacturer’s data packet descriptions will be referenced in the last part of the document: Part 5.
NoteString	char[64]	36	R	
ThisFileName	char[64]	100	R	Name of this file. Example:"LINE12-B.XTF"
NavUnits	WORD	164	M	0=Meters (i.e., UTM) or 3=Lat/Long
NumberOfSonarChannels	WORD	166	M	if > 6, header grows to 2K in size
NumberOfBathymetryChannels	WORD	168	M	
NumberOfSnippetChannels	BYTE	170	M	
NumberOfForwardLookArrays	BYTE	171	M	
NumberOfEchoStrengthChannels	WORD	172	M	
NumberOfInterferometryChannels	BYTE	174	M	
Reserved1	BYTE	175		Reserved ECA
Reserved2	BYTE	176	U	Reserved ECA
Reserved3	BYTE	177	U	Reserved. Set to 0.
ReferencePointHeight	float	178	O	Height of reference point above water line (m)
Navigation System Parameters				
ProjectionType	BYTE[12]	182	U	Not currently used. Set to 0.
SpheriodType	BYTE[10]	194	U	Not currently used. Set to 0.
NavigationLatency	long	204	O	Latency of nav system in milliseconds. (Usually GPS).
OriginY	float	208	U	Not currently used. Set to 0.
OriginX	float	212	U	Not currently used. Set to 0.
NavOffsetY	float	216	O	Orientation of positive Y is forward.
NavOffsetX	float	220	O	Orientation of positive X is to starboard.
NavOffsetZ	float	224	O	Orientation of positive Z is down. Just like depth.
NavOffsetYaw	float	228	O	Orientation of positive yaw is turn to right.
MRUOffsetY	float	232	O	Orientation of positive Y is forward
MRUOffsetX	float	236	O	Orientation of positive X is to starboard.
MRUOffsetZ	float	240	O	Orientation of positive Z is down. Just like depth.
MRUOffsetYaw	float	244	O	Orientation of positive yaw is turn to right.
MRUOffsetPitch	float	248	O	Orientation of positive pitch is nose up.

MRUOffsetRoll	float	252	O	Orientation of positive roll is lean to starboard.
ChanInfo	CHANINFO[6]	256	M	Data for each channel. The CHANINFO structures for all sidescan channels will always precede the structures for the bathymetry channels. If more than 6 structures are required, the header can grow in increments of 1024 bytes to allow for more CHANINFO structures.

The overall size is 1024 bytes.

3.3.3. CHANINFO structure

Table 5 : CHANINFO Structure

CHANINFO				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
TypeOfChannel	BYTE	0	M	SUBBOTTOM=0, PORT=1, STBD=2, BATHYMETRY=3
SubChannelNumber	BYTE	1	O	Index for which CHANINFO structure this is.
CorrectionFlags	WORD	2	O	1=sonar imagery stored as slant-range, 2=sonar imagery stored as ground range (corrected)
UniPolar	WORD	4	O	0=data is polar, 1=data is unipolar
BytesPerSample	WORD	6	M	1 (8-bit data) or 2 (16-bit data) or 4 (32-bit)
Reserved	DWORD	8	U	Previously this was SamplesPerChannel. The number of samples per channel can vary from ping to ping if the range scale changes. Because of this, the NumSamples value in the XTFPINGCHANHEADER structure (defined in Section 3.18) holds the number of samples to read for a given channel. For standard analog systems, this Reserved value is still filled in with 1024, 2048 or whatever the initial value is for SamplesPerChannel.
ChannelName	char[16]	12	O	Text describing channel. i.e., "Port 500"
VoltScale	float	28	O	This states how many volts are represented by a maximum sample value in the range [-5.0 to +4.9998] volts. Default is 5.0.
Frequency	float	32	O	Center transmit frequency
HorizBeamAngle	float	36	O	Typically 1 degree or so
TiltAngle	float	40	O	Typically 30 degrees
BeamWidth	float	44	O	3dB beam width, Typically 50 degrees
OffsetX	float	48	O	Orientation of positive X is to starboard. Note: This offset is entered in the Multibeam setup dialog box
OffsetY	float	52	O	Orientation of positive Y is forward. Note: This offset is entered in the Multibeam setup dialog box
OffsetZ	float	56	O	Orientation of positive Z is down. Just like depth. Note: This offset is entered in the Multibeam setup dialog box
OffsetYaw	float	60	O	Orientation of positive yaw is turn to right. If the multibeam sensor is reverse mounted (facing backwards), then OffsetYaw will be around 180 degrees. Note: This offset is entered in the Multibeam setup dialog box
OffsetPitch	float	64	O	Orientation of positive pitch is nose up. Note: This offset is entered in the Multibeam setup dialog box

OffsetRoll	float	68	O	Orientation of positive roll is lean to starboard. Note: This offset is entered in the Multibeam setup dialog box
BeamsPerArray	WORD	72	O	For forward look only (i.e., Sonatech DDS)
SampleFormat	BYTE	74	R	0 = Legacy 1 = 4-byte IBM float 2 = 4-byte integer 3 = 2-byte integer 4 = unused 5 = 4-byte IEEE float 6 = unused 7 = unused 8 = 1-byte integer
ReservedArea2	Char[53]	75	U	Unused Set value to 0

3.3.4. XTFPINGHEADER data layout

This data layout is usually used for sidescan data or FLS.

The value of NumChansToFollow in XTFPINGHEADER (structure defined in Table 6) determines the number of XTFPINGCHANHEADERs (structure defined in Table 7) that follows the XTFPINGHEADER.

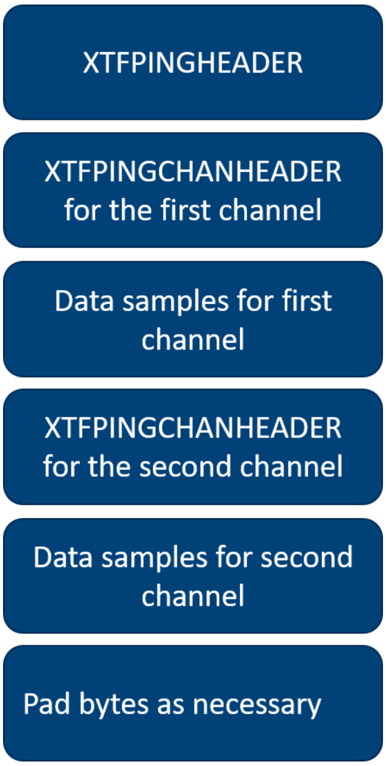


Figure 4 : XTF Sonar Ping Header Data Layout (example for two-channel Sidescan)

3.3.5. XTFBATHHEADER data layout

This data layout is usually used for bathymetry data.

XTFBATHHEADER structure is defined in Table 6. The structure is followed by a payload of bathymetry data, logged “raw” – that is, the data is unchanged and is logged exactly as received from the multibeam system. The packet is then padded with zero-filled bytes to bring the total XTF packet size to an even multiple of 64 bytes.

For details on processing the actual bathymetry data, consult the bathymetry system manufacturer.

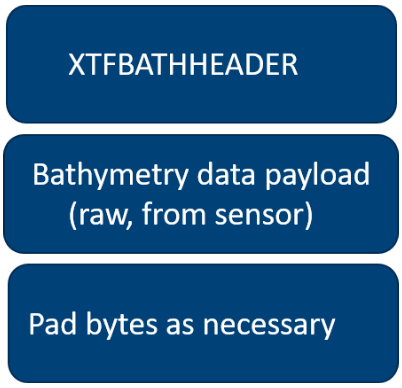


Figure 5 : XTF Bathymetry Ping Header Data Layout

3.4. XTFPINGHEADER AND XTFBATHHEADER (FOR SONAR AND BATHYMETRY DATA PACKETS)

Table 6 : XTFPINGHEADER / XTFBATHHEADER

XTFPINGHEADER/ XTFBATHHEADER				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	**See Header Type Table in Table 8**
SubChannelNumber	BYTE	3	M	If **HeaderType** is bathymetry, this indicates which head; if **HeaderType** is forward-looking sonar, and then this indicates which array. Also, Klein 5000 beam numbers are logged here.
NumChansToFollow	WORD	4	M	If **HeaderType** is sonar, number of channels to follow.
Reserved1	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	WORD	10	M	Total byte count for this ping including this ping header.
Year	WORD	14	M	Ping year
Month	BYTE	16	M	Ping month
Day	BYTE	17	M	Ping day
Hour	BYTE	18	M	Ping hour
Minute	BYTE	19	M	Ping minute
Second	BYTE	20	M	Ping seconds
HSeconds	BYTE	21	M	Ping hundredths of seconds (0-99)
JulianDay	WORD	22	O	Julian day of a ping's occurrence.
EventNumber	DWORD	24	O	Last logged event number; nav interface template token=**O**
PingNumber	DWORD	28	M	Counts consecutively (usually from 0) and increments for each update.
SoundVelocity	float	32	M	m/s
OceanTide	float	36	O	Altitude above Geoide (from RTK), if present; ELSE Ocean tide in meters; nav interface template token = **{t}**
Reserved2	DWORD	40	U	Unused. Set to 0.
ConductivityFreq	float	44	O	Conductivity frequency in Hz. nav interface template token = **Q Raw** CTD information. The Freq values are those sent up by the Seabird CTD. The Falmouth Scientific CTD sends up computed data.
TemperatureFreq	float	48	O	Temperature frequency in Hz. nav interface template token = **b Raw** CTD information. The Freq values are those sent up by the Seabird CTD. The Falmouth Scientific CTD sends up computed data.

PressureFreq	float	52	O	Pressure frequency in Hz. nav interface template token = **0**. Raw CTD information. The Freq values are those sent up by the Seabird CTD. The Falmouth Scientific CTD sends up computed data.
PressureTemp	float	56	O	Pressure temperature (Degrees C); nav interface template token = **; Raw** CTD information. The Freq values are those sent up by the Seabird CTD. The Falmouth Scientific CTD sends up computed data.
Conductivity	float	60	O	Conductivity in Siemens/m; nav interface token = **{c}**; can be computed from **Q** Computed CTD information. When using a Seabird CTD, these values are computed from the raw Freq values (above).
WaterTemperature	float	64	O	Water temperature in Celsius. nav interface token = **{w}**; can be computed from **b**. Computed CTD information. When using a Seabird CTD, these values are computed from the raw Freq values (above).
Pressure	float	68	O	Water pressure in psia; nav interface token = **{p}**; can be computed from **0**. Computed CTD information. When using a Seabird CTD, these values are computed from the raw Freq values (above).
ComputedSoundVelocity	float	72	O	Meters/second computed from Conductivity, WaterTemperature, and Pressure using the Chen Millero formula (1977), formula (JASA, 62, 1129-1135)
MagX	float	76	O	X-axis magnetometer data in mgauss. Nav interface template token = **e**. Sensors Information.
MagY	float	80	O	Y-axis magnetometer data in mgauss. Nav interface template token = **w**. Sensors Information.
MagZ	float	84	O	Z-axis magnetometer data in mgauss. Nav interface template token = **z**. Sensors Information.
AuxVal1	float	88	O	Sensors Information. Nav interface template token = **1**. Auxiliary values can be used to store and display any value at the user's discretion.
AuxVal2	float	92	O	Sensors Information. Nav interface template token = **2**. Auxiliary values can be used to store and display any value at the user's discretion.
AuxVal3	float	96	O	Sensors Information. Nav interface template token = **3**. Auxiliary values can be used to store and display any value at the user's discretion.
Reserved3	float	100	U	Reserved ECA
Reserved4	float	104	U	Reserved ECA
Reserved5	float	108	U	Reserved ECA
SpeedLog	float	112	O	Sensors Information. Speed log sensor on towfish in knots; Note: This is not fish speed. Nav interface template token = **s**.
Turbidity	float	116	O	Sensors Information. Turbidity sensor (0 to +5 volts) multiplied by 10000. nav interface template token = **|** (the "pipe" symbol).
ShipSpeed	float	120	O	Ship Navigation information. Ship speed in knots. nav interface template token = **v**.
ShipGyro	float	124	O	Ship Navigation information. Ship gyro in degrees. nav interface template token = **G**.
ShipYcoordinate	double	128	O	Ship Navigation information. Ship latitude or northing in degrees. nav interface template token = **y**.
ShipXcoordinate	double	136	O	Ship Navigation information. Ship longitude or easting in degrees. nav interface template token = **x**.
ShipAltitude	WORD	144	O	Ship altitude in decimeters

ShipDepth	WORD	146	O	Ship depth in decimeters.
FixTimeHour	BYTE	148	R	Sensor Navigation information. Hour of most recent nav update. nav interface template token = **H**.
FixTimeMinute	BYTE	149	R	Sensor Navigation information. Minute of most recent nav update. nav interface template token = **I**.
FixTimeSecond	BYTE	150	R	Sensor Navigation information. Second of most recent nav update. nav interface template token = **S**.
FixTimeHsecond	BYTE	151	R	Sensor Navigation information. Hundredth of a Second of most recent nav update.
SensorSpeed	float	152	R	Sensor Navigation information. Speed of towfish in knots. Used for speed correction and position calculation; nav interface template token = **V**.
KP	float	156	O	Sensor Navigation information. Kilometers Pipe; nav interface template token = **{K}**.
SensorYcoordinate	double	160	R	Sensor Navigation information. Sensor latitude or northing; nav interface template token = **E**. Note: when NavUnits in the file header is 0, values are in meters (northings and eastings). When NavUnits is 3, values are in Lat/Long. Also see the Layback value, below.
SensorXcoordinate	double	168	R	Sensor Navigation information. Sensor longitude or easting; nav interface template token = **N**. Note: when NavUnits in the file header is 0, values are in meters (northings and eastings). When NavUnits is 3, values are in Lat/Long. Also see the Layback value, below.
SonarStatus	WORD	176	O	Tow Cable information. System status value, sonar dependant (displayed in Status window).
RangeToFish	WORD	178	O	Slant range to sensor in decimeters; nav interface template token = **?** (question mark). Stored only – not used in any computation.
BearingToFish	WORD	180	O	Bearing to towfish from ship, stored in degrees multiplied by 100; nav interface template token = **>** (greater-than sign).
CableOut	WORD	182	O	Tow Cable information. Amount of cable payed out in meters; nav interface template token = **o**.
Layback	float	184	O	Tow Cable information. Distance over ground from ship to fish; nav interface template token = **l**.
CableTension	float	188	O	Tow Cable information Cable tension from serial port. Stored only; nav interface template token = **P**
SensorDepth	float	192	R	Sensor Attitude information. Distance (m) from sea surface to sensor. The deeper the sensor goes, the bigger (positive) this value becomes. nav interface template token = **0 (zero)**
SensorPrimaryAltitude	float	196	R	Sensor Attitude information. Distance from towfish to the sea floor; nav interface template token = **7**.
SensorAuxAltitude	float	200	O	Sensor Attitude information. Auxiliary altitude; nav interface template token = **a.**
SensorPitch	float	204	R	Sensor Attitude information. Pitch in degrees (positive=nose up); nav interface template token = **8**.
SensorRoll	float	208	R	Sensor Attitude information. Roll in degrees (positive=roll to starboard); nav interface template token = **9**.
SensorHeading	float	212	R	Sensor Attitude information. Sensor heading in degrees; nav interface template token = **h**.
Heave	float	216	O	Attitude information. Sensors heave at start of ping. Positive value means sensor moved up. Note: These Pitch, Roll, Heading, Heave and Yaw values are those received closest in time to this sonar or bathymetry update. If a TSS or MRU is being used with a multibeam/bathymetry sensor, the user should use the

				higher-resolution attitude data found in the XTFATTITUDEDATA structures.
Yaw	float	220	O	Attitude information. Sensor yaw. Positive means turn to right. Note: These Pitch, Roll, Heading, Heave and Yaw values are those received closest in time to this sonar or bathymetry update. If a TSS or MRU is being used with a multibeam/bathymetry sensor, the user should use the higher-resolution attitude data found in the XTFATTITUDEDATA structures. Since the heading information is updated in high resolution, it is not necessary to log or use Yaw in any processing.
AttitudeTimeTag	DWORD	224	R	Attitude information. In milliseconds - used to coordinate with millisecond time value in Attitude packets. (M)andatory when logging XTFATTITUDE packets.
DOT	float	228	O	Misc. Distance Off Track
NavFixMilliseconds	DWORD	232	R	Misc. millisecond clock value when nav received.
ComputerClockHour	BYTE	236	O	Computer clock time when this ping was received. May be different from ping time at start of this record if the sonar time-stamped the data and the two systems aren't synched. This time should be ignored in most cases.
ComputerClockMinute	BYTE	237	O	
ComputerClockSecond	BYTE	238	O	
ComputerClockHsec	BYTE	239	O	
FishPositionDeltaX	short	240	O	Additional Tow Cable and Fish information from Trackpoint. Stored as meters multiplied by 3.0, supporting +/- 10000.0m (usually from trackpoint); nav interface template token = **{DX}**.
FishPositionDeltaY	short	242	O	Additional Tow Cable and Fish information from Trackpoint. X, Y offsets can be used instead of logged layback nav interface template token = **{DY}**.
FishPositionErrorCode	BYTE	244	O	Additional Tow Cable and Fish information from Trackpoint. Error code for FishPosition delta x,y. (typically reported by Trackpoint).
OptionalOffset	DWORD	245	O	Triton 7125 only
CableOutHundredths	BYTE	249	O	Hundredths of a meter of cable out, to be added to the CableOut field.
ReservedSpace2	BYTE[6]	250	U	Unused. Set to 0.

The overall size is 256 bytes

3.4.1. XTFPINGCHANHEADER structure

XTFPINGCHANHEADER is used to hold data that can be unique to each channel from ping to ping. One of these headers follows each XTFPINGHEADER, no XTFPINGCHANHEADERS follow a XTFBATHHEADER.

Table 7 : XTFPINGCHANHEADER structure

XTFPINGCHANHEADER				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
ChannelNumber	WORD	0	M	Typically 0=port (low frequency) 1=stbd (low frequency) 2=port (high frequency) 3=stbd (high frequency)
DownsampleMethod	WORD	2	O	2 = MAX; 4 = RMS
SlantRange	float	4	M	Slant range of the data in meters
GroundRange	float	8	O	Ground range of the data; in meters ($SlantRange^2$ - $Altitude^2$)
TimeDelay	float	12	O	Amount of time, in seconds, to the start of recorded data. (almost always 0.0).
TimeDuration	float	16	R	Amount of time, in seconds, recorded (typically SlantRange/750)
SecondsPerPing	float	20	R	Amount of time, in seconds, from ping to ping. (SlantRange/750)
ProcessingFlags	WORD	24	O	4 = TVG; 8 = BAC&GAC; 16 = filter, etc. (almost always zero)
Frequency	WORD	26	R	Ccenter transmit frequency for this channel.
InitialGainCode	WORD	28	O	Settings as transmitted by sonar
GainCode	WORD	30	O	Settings as transmitted by sonar
BandWidth	WORD	32	O	Settings as transmitted by sonar
ContactNumber	DWORD	34	U	Contact information . Upated when contacts are saved in Target utility.
ContactClassification	WORD	38	U	Contact information . Updated when contacts are saved in Target utility.
ContactSubNumber	BYTE	40	U	Contact information . Udated when contacts are saved in Target utility
ContactType	BYTE	41	U	Contact information . Updated when contacts are saved in Target utility
NumSamples	DWORD	42	M	Number of samples that will follow this structure. The number of bytes will be this value multiplied by the number of bytes per sample. BytesPerSample found in CHANINFO structure (given in the file header).
MillivoltScale	WORD	46	O	Maximum voltage, in mv, represented by a full-scale value in the data.If zero, then the value stored in the VoltScale should be used instead. VoltScale can be found in the XTF file header, ChanInfo structure. Note that VoltScale is

				specified in volts, while MillivoltScale is stored in millivolts. This provides for a range of –65,536 volts to 65,535 volts.
ContactTimeOffTrack	float	48	U	Time off track to this contact (stored in milliseconds)
ContactCloseNumber	BYTE	52	U	
Reserved2	BYTE	53	U	Unused. Set to 0.
FixedVSOP	float	54	O	This is the fixed, along-track size of each ping, stored in centimeters.
Weight	short	58	O	Weighting factor passed by some sonars, this value is mandatory for Edgetech digital sonars types 24, 35, 38, 48 and Kongsberg SA type 48
ReservedSpace	BYTE[4]	60	U	Unused. Set to 0.

The overall size is 64 bytes. The number of samples following the XTFPINGCHANHEADER is defined in NumSamples.

4. XTF Data Packets

4.1. GENERAL DATA PACKET STRUCTURE

The HeaderType unused ranges before 200 are temporary frozen. In the future, those ranges may be allocated only if several manufacturers ask for the allocation of a specific packet number.

Each packet begins with a key pattern of bytes, called the "magic number", which can be used to align the data stream to the start of a packet. This “magic number” is followed by the header type described in the following table. Then, the other information of the packet depends of the packet type, and will be describe in the following parts for a selection of packets.

Table 8 : HEADER TYPE, list of the packet name and number

HEADERTYPE			
Packet Number	**Packet Name**	**Comments**	**Paragraph**
0	XTF_HEADER_SONAR	Sidescan data	4.2.1
1	XTF_HEADER_NOTES	Notes - text annotation	4.2.2
2	XTF_HEADER_BATHY	Bathymetry data	4.2.3
3	XTF_HEADER_ATTITUDE	Attitude packet	4.2.4
4	XTF_HEADER_FORWARD	Forward look data (Sonatech)	
5	XTF_HEADER_ELAC	Elac raw data packet	
6	XTF_HEADER_RAW_SERIAL	Raw ASCII serial port data.	4.2.5
7	XTF_HEADER_EMBED_HEAD	Embedded header record - num samples probably changed	
8	XTF_HEADER_HIDDEN_SONAR	Redundant (overlapping) ping from Klein 5000.	
9	XTF_HEADER_SEAVIEW_PROCESSED_BATHY	Bathymetry (angles) for Seaview.	
10	XTF_HEADER_SEAVIEW_DEPTHS	Bathymetry from Seaview data (depths).	
11	XTF_HEADER_RSVD_HIGHSPEED_SENSOR	Used by Klein. 0=roll, 1=yaw.	
12	XTF_HEADER_ECHOSTRENGTH	Elac EchoStrength (10 values).	
13	XTF_HEADER_GEOREC	Used to store mosaic parameters.	
14	XTF_HEADER_KLEIN_RAW_BATHY	Bathymetry data from the Klein 5000.	
15	XTF_HEADER_HIGHSPEED_SENSOR2	High speed sensor from Klein 5000.	4.2.6
16	XTF_HEADER_ELAC_XSE	Elac dual-head.	
17	XTF_HEADER_BATHY_XYZA	Full Processed X,Y,Z,A bathy data packet	4.2.7
18	XTF_HEADER_K5000_BATHY_IQ	Raw IQ data from Klein 5000 server	
19	XTF_HEADER_BATHY_SNIPPET	Snippet data for the Reson 8111	
20	XTF_HEADER_GPS	GPS Position.	
21	XTF_HEADER_STAT	GPS statistics.	
22	XTF_HEADER_SINGLEBEAM	Single Beam Data	
23	XTF_HEADER_GYRO	Heading/Speed Sensor.	
24	XTF_HEADER_TRACKPOINT	TrackPoint Structure	
25	XTF_HEADER_MULTIBEAM	New Multibeam packet	
26	XTF_HEADER_Q_SINGLEBEAM	Qinsy	4.2.8
27	XTF_HEADER_Q_MULTITX	Qinsy	4.2.9
28	XTF_HEADER_Q_MULTIBEAM	Qinsy	4.2.10
42	XTF_HEADER_SOURCETIME_NAVIGATION	Source / Receive navigation	4.2.11
43	XTF_HEADER_EXTENDED_NAVIGATION	Extended Navigation	
50	XTF_HEADER_TIME	Time Packet	
60	XTF_HEADER_BENTHOS_CAATI_SARA	Custom Benthos data.	4.2.12

61	XTF_HEADER_7125	7125 Bathy Data	
62	XTF_HEADER_7125_SNIPPET	7125 Bathy Data Snippets	4.2.13
63	XTF_HEADER_SWATHPLUS_BATHY		
65	XTF_HEADER_QINSY_R2SONIC_BATHY	QINSy R2Sonic bathymetry data	
66	XTF_HEADER_QINSY_R2SONIC_FTS	QINSy R2Sonics Foot Print Time Series (snippets)	
67	XTF_HEADER_R2SONICS_TRUEPIX		
68	XTF_HEADER_R2SONIC_BATHY	Triton R2Sonic bathymetry data	
69	XTF_HEADER_R2SONIC_FTS	Triton R2Sonic Footprint Time Series	
70	XTF_HEADER_CODA_ECHOSCOPE_DATA	Custom CODA Echoscope Data	
71	XTF_HEADER_CODA_ECHOSCOPE_CONFIG	Custom CODA Echoscope Data	
72	XTF_HEADER_CODA_ECHOSCOPE_IMAGE	Custom CODA Echoscope Data	
73	XTF_HEADER_EDGETECH_4600	ET4600 bathymetry data	
74	XTF_HEADER_EMRAW_BEAMANGLE	Simrad raw beam angle, datagram 4e	
75	XTF_HEADER_KLEIN5KV2_BATH	Klein bathy from 5000 v2	
76	XTF_HEADER_SOLSTICE	Solstice bathy	
77	XTF_HEADER_EMRAW_BEAMANGLE_66	Simrad raw beam angle, datagram 66	
78	XTF_HEADER_RESON_7018_ WATERCOLUMN	Reson 7125 watercolumn data	
79	XTF_HEADER_R2SONIC_WATERCOLUMN		
84	XTF_HEADER_SOURCETIME_GYRO	Heading/Speed Sensor	4.2.14
100	XTF_HEADER_POSITION	Raw position packet - Reserved for use by Reson, Inc. RESON ONLY.	
102	XTF_HEADER_BATHY_PROC	Bathy Proc	
103	XTF_HEADER_ATTITUDE_PROC	Attitude Proc	
104	XTF_HEADER_SINGLEBEAM_PROC	Single beam Proc	
105	XTF_HEADER_AUX_PROC	Aux Channel + AuxAltitude + Magnetometer	
106	XTF_HEADER_KLEIN3000_DATA_PAGE	Klein3000 SYS5000HEADER	
107	XTF_HEADER_POS_RAW_NAVIGATION	POS RAW Navigation	4.2.15
108	XTF_HEADER_KLEINV4_DATA_PAGE		
199	XTF_RAW_CUSTOMER_HEADER (OBSOLETE)	Anything goes data from specific vendor	4.2.16
200 - 255	Free of use for each manufacturer defined by the ManufacturerType in the XTFFILEHEADER	From this packet type are reserved for specific applications	

4.2. A SELECTION OF SPECIFIC DATA PACKETS STRUCTURES

4.2.1. Sidescan and subbottom data packet

Use existing XTF header type 0 = XTF_HEADER_SONAR
For more information, please refer to part 3.

4.2.2. Annotation data packet

Use existing XTF header type 1 = XTFNOTESHEADER

Figure 6 : XTFNOTEHEADER packet

Table 9 : XTFNOTEHEADER structure

XTFNOTESHEADER				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	1 = XTF_HEADER_NOTES (defined in Xtf.h)
SubChannelNumber	BYTE	3	O	0=XTF notes from Param window, 1=vessel name, 2=survey area, 3=operator name.
NumChansToFollow	WORD	4	U	Unused. Set to 0.
Reserved	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Must be 256 (size of this packet is always 256 bytes).
Year	WORD	14	M	Annotation Year
Month	BYTE	16	M	Annotation month
Day	BYTE	17	M	Annotation day
Hour	BYTE	18	M	Annotation hour
Minute	BYTE	19	M	Annotation minute
Second	BYTE	20	M	Annotation second
ReservedBytes	BYTE[35]	21	U	Unused. Set to 0.
NotesText	char [200]	56	M	Annotation text

The overall size is 256 bytes.

4.2.3. Bathymetry data packet

Use existing XTF header type 2 = XTF_HEADER_BATHY
For more information, please refer to part 3.

4.2.4. Attitude data packet

Use existing XTF header type 3 = XTFATTITUDEDATA

Figure 7 : XTFATTITUDEDATA packet

Table 10 : XTFATTITUDEDATA structure

XTFATTITUDEDATA				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	1	M	3 = XTF_HEADER_ATTITUDE (defined in Xtf.h)
SubChannelNumber	BYTE	2	O	When HeaderType is Bathy, indicates which head.
NumChansToFollow	WORD	4	O	If Sonar Ping, Number of channels to follow
Reserved1	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
Reserved2	DWORD[2]	14	U	Unused. Set to 0.
EpochMicroseconds	DWORD	22	O	0 -999999
SourceEpoch	DWORD	26	O	Source Epoch Seconds since 1/1/1970, will be followed attitude data even to 64 bytes
Pitch	float	30	O	Positive value is nose up
Roll	float	34	O	Positive value is roll to starboard
Heave	float	38	O	Positive value is sensor up.
Yaw	float	42	O	Positive value is turn right
TimeTag	DWORD	46	O	System time reference in milliseconds
Heading	float	50		In degrees, as reported by MRU. TSS doesn't report heading, so when using a TSS this value will be the most recent ship gyro value as received from GPS or from any serial port using 'G' in the template.
Year	WORD	54	O	Fix year.
Month	BYTE	56	O	Fix month.
Day	BYTE	57	O	Fix day.
Hour	BYTE	58	O	Fix hour.
Minutes	BYTE	59	O	Fix minute.
Seconds	BYTE	60	O	Fix seconds.
Milliseconds	WORD	61	O	(0 – 999). Fix milliseconds.
Reserved3	BYTE[1]	63	U	Unused. Set to 0.

The overall size is 64 bytes.

4.2.5. Raw Serial data packets

Use existing XTF header type 6 = XTFRAWSERIALHEADER

Figure 8 : XTFRAWSERIALHEADER packet

Table 11 : XTFRAWSERIALHEADER structure

XTFRAWSERIALHEADER				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	6 = XTF_HEADER_RAW_SERIAL (defined in Xtf.h)
SerialPort	BYTE	3	O	Serial port used to receive this data. COM1=1, COM2=2, etc. Set to 0 when data is received by other means (i.e., memory-mapped file).
NumChansToFollow	WORD	4	U	Unused. Set to 0.
Reserved	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
Year	WORD	14	M	Year
Month	BYTE	16	M	Month
Day	BYTE	17	M	Day
Hour	BYTE	18	M	Hour
Minute	BYTE	19	M	Minute
Second	BYTE	20	M	Seconds
HSeconds	BYTE	21	O	Hundredths of seconds (0-99)
JulianDay	WORD	22	O	Days since Jan 1
TimeTag	DWORD	24	O	Millisecond timer value
StringSize	WORD	28	M	Number of valid chars in RawAsciiData string
RawAsciiData	char[StringSize]	30	M	Characters of Raw ASCII data

4.2.6. High speed sensor packet

This packet is used to store high speed sensor data from Klein 5000.

Use existing XTF header type 15 = XTF_HEADER_HIGHSPEED_SENSOR

Figure 9 : XTF_HEADER_HIGHSPEED_SENSOR packet

Table 12 : XTFHIGHTSPEEDSENSOR structure

XTFHIGHSPEEDSENSOR				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	15 = XTFHIGHSPEEDSENSOR (defined in Xtf.h)
SubChannelNumber	BYTE	3	M	0=altitude, 1=roll, 2=yaw
NumChansToFollow	WORD	4	U	Unused. Set to 0
Reserved1	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
Year	WORD	14	M	
Month	BYTE	16	M	
Day	BYTE	17	M	
Hour	BYTE	18	M	
Minute	BYTE	19	M	
Second	BYTE	20	M	
HSeconds	BYTE	21	M	
NumSensorBytes	DWORD	22	M	Number of bytes of sensor data following this structure.
RelativeBathyPingNum	DWORD	26	M	Bathymetry ping number belonging to this sensor data.
Reserved3	BYTE[34]	30	U	Unused. Set to 0.

The overall size is 64 bytes.

4.2.7. Processed bathymetry XYZA

Use existing XTF header type 17 = XTF_HEADER_BATHY_XYZA

Figure 10 : XTF_HEADER_BATHY_XYZA packet

Table 13 : XTFBEAMXYZA structure

XTFBEAMXYZA				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
dPosOffsetTrX	double	0	M	Offset Northing from fish
dPosOffsetTrY	double	8	M	Offset Easting from fish
fDepth	float	16	M	Absolute Depth
dTime	double	20	M	Two way travel time
usAmpl	short	28	M	Amplitude
ucQuality	BYTE	30	M	Quality.

The overall size is 31 bytes.

4.2.8. QPS Singlebeam data layout

For each single beam transducer update one XTFQPSSINGLEBEAM record is written to the XTF file. A single beam record is identified by it Header type (26 = XTF_HEADER_Q_SINGLEBEAM)

Figure 11 : XTFQPSSINGLEBEAM packet

The Record description is shown in the following table.

Table 14 : XTFQPSSINGLEBEAM structure

XTFQPSSINGLEBEAM				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	26 = XTF_HEADER_Q_SINGLEBEAM
SubChannelNumber	BYTE	3	M	ID in CHANNELINFO structures
NumChansToFollow	WORD	4	U	Unused. Set to 0.
Reserved1	WORD[2]	6	U	Unused. Set to 0. 2 * size of (Word)
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
TimeTag	DWORD	14	M	Time stamp given in milliseconds
Id	int	18	O	ID
SoundVelocity	float	22	M	Sound Velocity in m/sec
Intensity	float	26	O	Signal Strength
Quality	int	30	O	Quality
TwoWayTravelTime	float	34	M	Two way travel time in seconds
	WORD	38	M	Year
	BYTE	40	M	Month
	BYTE	41	M	Day
	BYTE	42	M	Hour
	BYTE	43	M	Minute
	BYTE	44	M	Second
	WORD	45	M	MilliSeconds
	BYTE	Reserved[7]	M	For future expansion

Note: To identify the transducer location of the update you should look up the sub channel number in the CHANINFO structures in the XTF file header. The XTF files generated by QPS will have for each channel info structure a unique sub channel number.

4.2.9. QPS MultiTx data layout

Use existing XTF header type 27 = XTF_HEADER_Q_MULTITX

For each single beam transducer update one XTFBATHHEADER record is written to the XTF file.
This Header is followed by N times XTFQPSMULTITXENTRY, where N is the number of transducers.
N is also written in the NumChansToFollow member of the Header.

Figure 12 : XTF_HEADER_Q_MULTITX data layout

The Record description is shown in the following table.

Table 15 : XTFQPSMULTITXENTRY structure

XTFQPSMULTITXENTRY				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
Id	int	0	O	Beam ID
Intensity	float	4	O	Signal Strength
Quality	int	8	O	Quality
TwoWayTravelTime	float	12	M	Two way travel time in seconds
DeltaTime	float	16	M	Difference between header in seconds
OffsetX	float	20	M	Location of ship's reference frame
OffsetY	float	24	M	Location of ship's reference frame
OffsetZ	float	28	M	Location of ship's reference frame
Reserved	float[4]	32	M	Reserved

Delta Time member is important to calculate the exact timestamp of the transducers ping time. In order to get the right absolute timestamp for the transducer then you must take the timetag from the XTFBATHHEADER and ADD the delta time to it. Usually the delta time figures are negative.

4.2.10. QPS Multibeam data layout

Use existing XTF header type 28 = XTF_HEADER_Q_MBEENTRY

For one multibeam system update one XTFBATHHEADER record is written to the XTF file. This header is followed by N times XTFQPSMBEENTRY, where N is the number of beams that updated.
N is also written in the NumChansToFollow member of the header.

Figure 13 : XTF_HEADER_Q_MBEENTRY data layout

The record description is shown in the following table.

Table 16 : XTFQPSMULITXENTRY structure

XTFQPSMULTITXENTRY				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
Id	int	0	O	Beam ID
Intensity	double	4	O	Signal Strength
Quality	int	12	O	Quality
TwoWayTravelTime	double	16	M	Two way travel time in seconds
DeltaTime	double	24	M	Beam time offset
Beam Angle	double	32	M	Beam angle
Tilt Angle	double	40	M	Tilt angle
Reserved	float[4]	48	M	Reserved

- Reported time in XTFBATHHEADER will always be the transmission (ping) time
- Delta Time can be used for profilers to calculate the ping time per beam.
- Beam Angle convention, Negative to port side, nadir beam 0degs, positive to starboard side.
- Tilt angle convention positive forward, negative backward (used for pitch steering)

4.2.11. Source / Receive navigation

Source time-stamped navigation data, holds updates of any navigation data.

Use existing XTF header type 42 = XTF_HEADER_NAVIGATION

Figure 14 : XTFHEADERNAVIGATION structure

Table 17 : XTFHEADERNAVIGATION structure

XTFHEADERNAVIGATION				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	42 = XTF_HEADER_NAVIGATION (defined in Xtf.h)
Reserved	BYTE[7]	3	M	Must be here!
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
Year	WORD	14	O	Source time Year
Month	BYTE	16	O	Source time Month
Day	BYTE	17	O	Source time Day
Hour	BYTE	18	O	Source time Hour
Minute	BYTE	19	O	Source time Minute
Second	BYTE	20	O	Source time Seconds
Microseconds	DWORD	21	O	0 - 999999
SourceEpoch	DWORD	25		Source Epoch Seconds since 1/1/1970
TimeTag	DWORD	29		System Reference time in milliseconds
Raw Y Coordinate	Double	33	O	Raw position from POSMV or other time stamped navigation source
Raw X Coordinate	Double	41	O	Raw position from POSMV or other time stamped navigation source
Raw Altitude	Double	49	O	Altitude, can hold real-time kinematics altitude
TimeFlag	BYTE	57	O	Time stamp validity: 0 = only receive time valid 1 = only source time valid 3 = both valid
Reserved1	BYTE[6]	58	U	Padding to make the structure 64 bytes

4.2.12. Benthos CAATI Packet data layout

Use existing XTF header type 60 = XTF_HEADER_BENTHOS_CAATI_SARA.

Store SARA/CAATI 3D data in an XTFPINGHEADER followed by one XTFPINGCHANHEADER followed by the Benthos SARA/CAATI "PINGINFO" data. For more information on the Benthos PINGINFO structure, please contact Benthos.

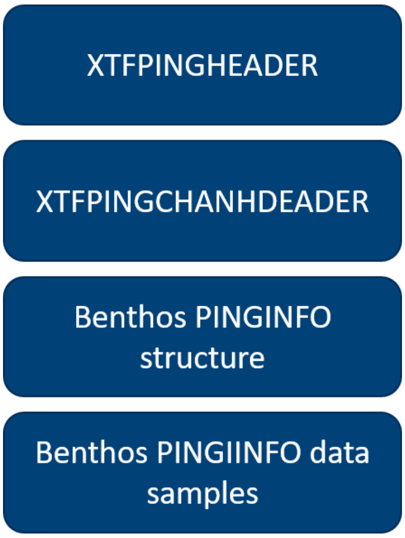


Figure 15 : BENTHOS CAATI SARA ping data layout

4.2.13. Reson 7125 Bathy Data Snippets

Use existing XTF header type 62 = XTF_HEADER_7125_SNIPPET

The XTF BATHY SNIPPET data starts with an XTFBATHHEADER then it is followed by SNP0, refer to Table 18. The number of SNP1 (refer to Table 19) structures to follow the SNP0 is determined by the beamcount value stored in the SNP0 structure.   The entire XTF packet is padded with zero-filled bytes to make the size an even multiple of 64.

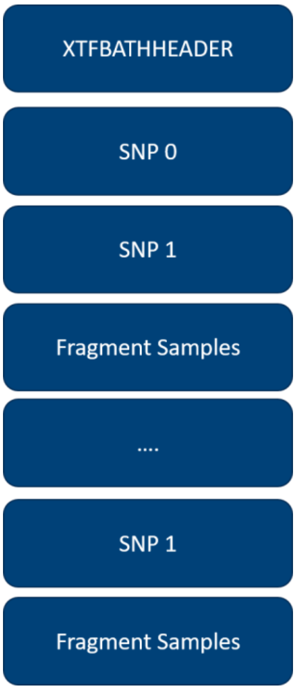


Figure 16 : XTF Bathy Snippet data layout

Table 18 : SNP0 structure (generated by Reson Seabat)

SNP0				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
ID	unsigned long	0	M	Identifier code. SNP0= 0x534E5030
HeaderSize	unsigned short	4	M	Header size, bytes.
DataSize	unsigned short	6	M	Data size following header, bytes.
PingNumber	unsigned long	8	M	Sequential ping number.
Seconds.	unsigned long	12	M	Time since 00:00:00, 1-Jan-1970
Millisec	unsigned long	16	M	
Latency	unsigned short	20	M	Time from ping to output (milliseconds)
SonarID	unsigned short[2]	22	M	Least significant four bytes of Ethernet address.
SonarModel	unsigned short	26	M	Coded model number of sonar.
Frequency	unsigned short	28	M	Sonar frequency (kHz).
SSpeed	unsigned short	30	M	Programmed sound velocity (m/sec).
SampleRate	unsigned short	32	M	A/D sample rate (samples/sec).
PingRate	unsigned short	34	M	Pings per second, 0.001 Hz steps.
Range	unsigned short	36	M	Range setting (meters).
Power	unsigned short	38	M	Power

Gain	unsigned short	40	M	(b15=auto, b14=TVG, b6..0=gain).
PulseWidth	unsigned short	42	M	Transmit pulse width (microseconds).
Spread	unsigned short	44	M	TVG spreading, n*log(R), 0.25dB steps.
Absorb	unsigned short	46	M	TVG absorption, dB/km, 1dB steps.
Proj	unsigned short	48	M	b7 = steering, b4..0 = projector type.
ProjWidth	unsigned short	50	M	Transmit beam width along track, 0.1 deg steps.
SpacingNum	unsigned short	52	M	Receiver beam spacing, numerator, degrees.
SpacingDen	unsigned short	54	M	Receiver beam spacing, denominator.
ProjAngle	short	56	M	Projector steering, degrees*PKT_STEER_RES
MinRange	unsigned short	58	M	Range filter settings
MaxRange	unsigned short	60	M	
MinDepth	unsigned short	62	M	Depth filter settings.
MaxDepth	unsigned short	64	M	Depth filter settings.
Filters	unsigned short	66	M	Enabled filters: b1=depth, b0=range.
bFlags	BYTE[2]	68	M	Bits 0 – 11 spare, Bits 12 – 14 snipMode, Bit 15 RollStab. Bit 0: roll stabilization enabled.
HeadTemp	Short	70	M	Head temperature, 0.1C steps.
BeamCnt	unsigned short	72	M	number of beams

The overall size is 74 bytes.

Table 19 : SNP1 structure

SNP1				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
ID	unsigned long	0	M	Identifier code. SNP1= 0x534E5031
HeaderSize	unsigned short	4	M	Header size, bytes.
DataSize	unsigned short	6	M	Data size following header, bytes.
PingNumber	unsigned long	8	M	Sequential ping number.
Beam	unsigned short	12	M	Beam number, 0..N-1.
SnipSamples	unsigned short	14	M	Snippet size, samples.
GainStart	unsigned short	16	M	Gain at start of snippet, 0.01 dB steps, 0=ignore.
GainEnd	unsigned short	18	M	Gain at end of snippet, 0.01 dB steps, 0=ignore.
FragOffset	unsigned short	20	M	Fragment offset, samples from ping.
FragSamples	unsigned short	22	M	Fragment size, samples.

The overall size is 24 bytes.

4.2.14. Gyro data structure

Source time-stamped gyro data holds updates of any gyro data.

Use existing XTF header type 84 = XTF_HEADER_SOURCETIME_GYRO

Figure 17 : XTFHEADERGYRO structure

Table 20 : XTFHEADERGYRO structure

XTFHEADERGYRO				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	84 = XTF_HEADER_SOURCETIME_GYRO (defined in Xtf.h)
Reserved	BYTE[7]	3	M	Must be here!
NumBytesThisRecord	DWORD	10	M	Total byte count for this ping including this ping header.
Year	WORD	14	O	Source time Year
Month	BYTE	16	O	Source time Month
Day	BYTE	17	O	Source time Day
Hour	BYTE	18	O	Source time Hour
Minute	BYTE	19	O	Source time Minute
Second	BYTE	20	O	Source time Seconds
Microseconds	DWORD	21	O	0 - 999999
SourceEpoch	DWORD	25	O	Source Epoch Since 1/1/1970
TimeTag	DWORD	29	O	System Time reference in milliseconds
Gyro	float	33	O	Raw heading (0 – 360)
TimeFlag	BYTE	37	O	Time stamp validity: 0 = only receive time valid 1 = only source time valid 3 = both valid
Reserved1	BYTE[26]	38	U	Padding to make the structure 64 bytes

4.2.15. POS RAW Navigation

Use existing XTF header type 107 = XTF_HEADER_POS_RAW_NAVIGATION

Figure 18 : XTFPOSRAWNAVIGATION packet

Table 21 : XTFPOSRAWNAVIGATION structure

XTFPOSRAWNAVIGATION				
Field	**Data type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	107 = XTF_HEADER_POS_RAW_NAVIGATION
SubChannelNumber	BYTE	3	U	Unused. Set to 0.
NumChansToFollow	WORD	4	U	Unused. Set to 0.
Reserved1	WORD[2]	6	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Must be 64. (Size of this packet is always 64 bytes).
Year	WORD	14	M	Fix year.
Month	BYTE	16	M	Fix month.
Day	BYTE	17	M	Fix day.
Hour	BYTE	18	M	Fix hour.
Minutes	BYTE	19	M	Fix minute.
Seconds	BYTE	20	M	Fix seconds.
MicroSeconds	WORD	21	M	(0 – 9999). Fix tenths of milliseconds.
RawYcoordinate	double	23	M	Raw position from POSRAW or other time stamped nav source.
RawXcoordinate	double	31	M	Raw position from POSRAW or other time stamped nav source.
RawAltitude	double float	39	O	Altitude, can hold RTK altitude.
Pitch	float	47	O	Positive value is nose up
Roll	float	51	O	Positive value is roll to starboard
Heave		55	O	Positive value is sensor up.
Heading	float	59	O	In degrees, as reported by MRU. TSS doesn't report heading, so when using a TSS this value will be the most recent ship gyro value as received from GPS or from any serial port using 'G' in the template.
Reserved2	BYTE	63	U	Unused.

4.2.16. Data from specific manufacturers (obsolete)

This custom header is now obsolete.

In order to use custom information in your XTF, please refer to the section “What’s new” paragraph 1.2

Use existing XTF header type 199 = XTF_RAW_CUSTOMER_HEADER

The purpose of this structure is that it should be used as a 64 byte header in front of some user defined data block. The NumBytesThisRecord field defines the length of this block of data +64 bytes for this header. It is not mandatory that the user defined data block is padded such that its total size is a multiple of 64 bytes, however for compatibility with other structures in XTF it is recommended.

Figure 19 : XTFRAWCUSTOMHEADER structure

Table 22 : XTFRAWCUSTOMHEADER structure

XTFRAWCUSTOMHEADER				
Field	**Data Type**	**Byte Offset**	**Status**	**Comment**
MagicNumber	WORD	0	M	Must be set to 0xFACE (hexadecimal value).
HeaderType	BYTE	2	M	199 = custom vendor data follows (defined in Xtf.h)
ManufacturerID	BYTE	3	O	(eg 1 = Benthos, 2 = Reson, 3 =Edgetech up to maximum of 256 vendors (see table)
SonarID	WORD	4	O	TBD (eg 4200, 1624, 4700, 7125 etc)
PacketID	WORD	6	O	TBD (eg 7000, 7503 etc)
Reserved1	WORD[1]	8	U	Unused. Set to 0.
NumBytesThisRecord	DWORD	10	M	Total byte count for this data packet including this header. (NumCustomerBytes +64) Note that the user data indicated by NumCustomerBytes may also be padded to a 64 byte boundary. (Optional but recommended)
Year	WORD	14	O	
Month	BYTE	16	O	
Day	BYTE	17	O	
Hour	BYTE	18	O	
Minute	BYTE	19	O	
Second	BYTE	20	O	
Hseconds	BYTE	21	O	Hundredths of seconds (0-99)
Julian Day	WORD	22	O	
Reserved2	WORD[2]	24	U	
PingNumber	DWORD	28	O	
TimeTag	DWORD	32	O	
NumCustomerBytes	DWORD	36	O	

Reserved3	BYTE[24]	40	U	Padding to make the structure 64 bytes

The manufacturers ID numbers are described in the following Table:

Table 23 : XTF_RAW_CUSTOMER_HEADER - Manufacturers ID Numbers

1	Benthos	14	27	40
2	Reson	15	28	41
3	Edgetech	16	29	42
4	Klein	17	30	43
5	CODA	18	31	44
6	Kongsberg	19	32	45
7	CMAX	20	33	46
8	Marine Sonics	21	34	47
9	Applied Signal	22	35	48
10	Imagenex	23	36	49
11	GeoAcoustics	24	37	50
12		25	38	51
13		26	39	52

The last 32 values in PacketID are reserved, regardless of ManufacturerID. These are reserved to allow generic data to be wrapped in the Custom packet. The following table defines the reserved PacketIDs:

Table 24 : XTF_RAW_CUSTOMER_HEADER - Reserved PacketIDs

Reserved PacketID	**Name**	**Comment**
65504	Generic XML	XML must be compliant to W3C standards.
65505	SETTINGS XML	See Appendix 4
65506-65535	Unused	--

5. Appendix: Manufacturer’s data packet descriptions

The manufacturer’s data packet descriptions (if they are available for distribution) will be referenced in this last part of the document. All this documents are available for download on the ECA Group website: https://www.ecagroup.com/en/xtf-file-format.

Table 25: Manufacturer's data packet description

Manufacturer	**Document or Link**	**Version**	**Date**
R2Sonic	*Appendix IX_R2Sonic 202X series Data Formats.pdf* Extract from *Sonic 2020 Operation Manual V3.3r003.pdf*	6.3	*February 2019*
QPS	https://confluence.qps.nl/qinsy/latest/en/overview-xtf-records-183828700.html	9.1	*November 2019*
Teledyne Marine	https://github.com/Teledyne-Marine/7k	3.12	*December 2020*