DESCRIPTION OF THE *HAC* STANDARD FORMAT FOR RAW AND EDITED HYDROACOUSTIC DATA, VERSION 1.0

Y. Simard, I. McQuinn, M. Montminy, C. Lang,
D. Miller, C. Stevens, D. Wiggins and C. Marchalot

Regional Science Branch
Department of Fisheries and Oceans
Maurice Lamontagne Institute
P.O. Box 1000
Mont-Joli , Québec
Canada G5H 3Z4

1997

Canadian Technical Report of Fisheries and Aquatic Sciences 2174

Pêches et Océans Fisheries and Oceans

Canada

Canadian Technical Report of Fisheries and Aquatic Sciences

Technical reports contain scientific and technical information that contributes to existing knowledge but which is not normally appropriate for primary literature. Technical reports are directed primarily toward a worldwide audience and have an international distribution. No restriction is placed on subject matter and the series reflects the broad interests and policies of the Department of Fisheries and Oceans, namely, fisheries and aquatic sciences.

Technical reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report is abstracted in *Aquatic Sciences and Fisheries Abstracts* and indexed in the Department's annual index to scientific and technical publications.

Numbers 1-456 in this series were issued as Technical Reports of the Fisheries Research Board of Canada. Numbers 457-714 were issued as Department of the Environment, Fisheries and Marine Service, Research and Development Directorate Technical Reports. Numbers 715-924 were issued as Department of Fisheries and the Environment, Fisheries and Marine Service Technical Reports. The current series name was changed with report number 925.

Technical reports are produced regionally but are numbered nationally. Request for individual reports will be filled by the issuing establishment listed on the front cover and title page. Out-of-stock reports will be supplied for a fee by commercial agents.

Rapport technique canadien des sciences halieutiques et aquatiques

Les rapports techniques contiennent des renseignements scientifiques et techniques qui constituent une contribution aux connaissances actuelles, mais qui ne sont pas normalement apropriés pour la publication dans un journal scientifique. Les rapports techniques sont destinés essentiellement à un public international et ils sont distribués à cet échelon. Il n'y a aucune restriction quant au sujet; de fait, la série reflète la vaste gamme des intérêts et des politiques du ministère de Pêches et des Océans, c'est-à-dire les sciences halieutiques et aquatiques.

Les rapports techniques peuvent être cités comme des publications complètes. Le titre exact paraît au-dessus du résumé de chaque rapport. Les rapports techniques sont résumé dans la revue *Résumé des scieces aquatiques et halieutiques*, et ils sont classés dans l'index annuel des publications scientifiques et techniques du Ministère.

Les numéros 1 à 456 de cette série ont été publiés à titre de rapports techniques de l'Office des recherches sur les pêcheries du Canada. Les numéros 457 à 714 sont parus à titre de rapports techniques de la Direction générale de la recherche et du développement, Service des pêches et de la mer, ministère de l'Environnement. Les numéros 715 à 924 ont été publiés à titre de rapports techniques du Service des pêches et de la mer, ministère des Pêches et de l'Environnement. Le nom actuel de la série a été établi lors de la parution du numéro 925.

Les rapports techniques sont produits à l'échelon régional, mais numérotés à l'échelon naitonal. Les demandes de rapports seront satisfaites par l'établissement auteur dont le nom figure sur la couverture et la page du titre. Les rapports épuisés seront fournis contre rétribution par des agents commerciaux.

Canadian Technical Report of
Fisheries and Aquatic Sciences 2174

1997

Description of the *HAC* standard format for raw and edited hydroacoustic data, version 1.0

Y. Simard[1], I. McQuinn[1], M. Montminy[2], C. Lang[3],
D. Miller[3], C. Stevens[3], D. Wiggins[4] and C. Marchalot[5]

Regional Science Branch
Department of Fisheries and Oceans
Maurice Lamontagne Institute
P.O. Box 1000
Mont-Joli, Québec
Canada G5H 3Z4

[1]: Dept. Fisheries and Oceans, Maurice Lamontagne Institute, P.O. Box 1000, Mont-Joli, Québec, Canada G5H 3Z4
[2]: Blizzard Logiciels, 1135 Murray, #3, Québec, Québec, Canada G1S 3B7
[3]: Dept. Fisheries and Oceans, Northwest Atlantic Fisheries Center, P.O. Box 5667, St. John's, Newfoundland, Canada A1C 5X1
[4]: Biosonics, 4027 Leary Way NW, Seattle, WA 98107, USA
[5]: Ifremer, DITI/DSI, Centre de Brest, B.P. 70 - 29280 Plouzané, France

Cat. No. Fs 97-6/2174E ISSN 0706-6457

Correct citation for this publication:

Simard, Y., I. McQuinn, M. Montminy, C. Lang, D. Miller, C. Stevens, D. Wiggins and C. Marchalot. 1997. Description of the *HAC* standard format for raw and edited hydroacoustic data, version 1.0. Can. Tech. Rep. Fish. Aquat. Sci. 2174: vii + 65 pp.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Simard, Y., I. McQuinn, M. Montminy, C. Lang, D. Miller, C. Stevens, D. Wiggins and C. Marchalot. 1997. Description of the *HAC* standard format for raw and edited hydroacoustic data, version 1.0. Can. Tech. Rep. Fish. Aquat. Sci. 2174: vii + 65 pp.

The international scientific community has recently stressed the need for a standard format to facilitate the exchange of fisheries acoustic data and/or their processing tools. This report presents a versatile standard format for raw and edited hydroacoustic data. This format resulted from a workshop held by the Department of Fisheries and Oceans and from discussions with various users and echosounder manufacturers around the world. It is called the *HAC* format, an acronym for *H*ydro*AC*oustics. These three characters provide the extension for the file names. The *HAC* standard format is nonrestrictive and independent of the computer platform. It can accommodate most echosounders, of single or multiple channels, and allows easy additions for future developments. The information in the files is structured by tuples, which are labeled groups of bytes assigned to a particular type of information. The various tuple types required for standard fisheries acoustics and common echosounders are defined in detail. The *HAC* standard format will be maintained by a coordination group, and tools to facilitate its use will be available on the DFO Internet site (http://207.167.196.2//hydro/homepage.html).

RÉSUMÉ

Simard, Y., I. McQuinn, M. Montminy, C. Lang, D. Miller, C. Stevens, D. Wiggins and C. Marchalot. 1997. Description of the *HAC* standard format for raw and edited hydroacoustic data, version 1.0. Can. Tech. Rep. Fish. Aquat. Sci. 2174: vii + 65 pp.

La communauté scientifique internationale a récemment fait état de la nécessité d'un format standard pour faciliter les échanges de données et/ou de leurs outils d'analyse en acoustique halieutique. Ce rapport présente un format standard versatile pour des données hydroacoustiques brutes ou éditées. Ce format résulte d'un atelier tenu par le Ministère des Pêches et des Océans et de discussions avec divers utilisateurs et manufacturiers d'échosondeurs à travers le monde. Il fut nommé le format *HAC*, un sigle rappelant *H*ydro*AC*oustique. Ces trois caractères sont utilisés comme extension aux noms de fichiers. Le format standard *HAC* est non restreignant et indépendant de la famille d'ordinateurs employée. Il peut accomoder la plupart des échosondeurs, à voies simples ou multiples, et permet des additions faciles pour les développements futurs. Dans les fichiers, l'information est structurée par des tuples, qui sont des groupes d'octets étiquetés auxquels un type particulier d'information est assigné. Les divers types de tuples requis en acoustique halieutique standard ainsi que pour l'usage d'échosondeurs courants sont définis en détail. Le format standard *HAC* sera entretenu par un groupe de coordination et des outils pour faciliter son utilisation seront disponibles sur le site Internet du MPO (http://207.167.196.2//hydro/homepage.html).

INTRODUCTION

CONTEXT

The need for standard formats at various levels of data processing in fisheries acoustics was stressed by the international community at the ICES-Fisheries Acoustics Science and Technology meeting in Woods Hole, Mass., USA, on 17-19 April 1996 (ICES 1996 CM/B:3). In 1995, within the framework of its National Hydroacoustic Program (NHP), the Department of Fisheries and Oceans (DFO) identified the need for a standard format to store the fisheries acoustic data acquired by various sensors used in DFO laboratories. To elaborate a proposal for the format, a workshop, involving users and manufacturers, was held at the Maurice Lamontagne Institute (MLI) on 12-14 December 1995. This document presents the versatile standard format that resulted from this workshop and from further discussions and comments from users and manufacturers around the world. The file name extension ***.HAC***, for ***h***ydro***ac***oustics, identifies this format. This format will be updated regularly on the NHP Internet site (http://207.167.196.2//hydro/homepage.html), where tools to facilitate its use will be available.

ADVANTAGES OF A STANDARD DATA FORMAT

Users recognize that standards are necessary for sharing data and/or processing algorithms, but they look for formats that are primarily versatile and nonrestrictive. The present format, which deals with raw and/or edited hydroacoustic data, has these essential qualities. It also satisfies the following needs: (1) accommodation of most echosounders (single to multiple channels), (2) variable data resolutions in time/space and amplitude, (3) efficiency and versatility in data storage and access, (4) room for additions and future developments, (5) forward and backward compatibility in time, (6) self-contained files, (7) forward and backward scrolling possibility within the file, and (8) computer platform independence. This format is designed to preserve the raw data through as many data analysis steps as possible, including the data edition or classification steps.

TUPLE FILE STRUCTURE

The adopted format combined two proposals from Biosonics and MLI, identified as a *tuple file* format. In a tuple data file, the information is chopped and packaged into slices, called *tuples*. A tuple is defined as a structured group of bytes. The different types of tuples are labeled by tags. Some tuple types can be related to others in a hierarchical order, analogous to object-oriented programming, where common information is separated from particular information to prevent redundancy. Some tuples then inherit the properties of others by parent/child relationships.

A tuple file is a series of tuples. Tuple types describe how particular types of information are stored. The storage of information by tuple is versatile and facilitates the access to specific

information. A file could have as many tuple types as needed to code the information *in extenso* or only a few tuple types to represent a minimum of information. Further, a data access program may search only for the tuple types of interest for a particular application. The format structure thus allows upward and backward compatibility in time. The addition of new tuple types to accommodate new instruments or applications is readily possible (forward compatibility). New versions of the program can always read the old tuple types, i.e. old files (backward compatibility). This *HAC* standard data format is therefore of great interest for the storage of raw fisheries hydroacoustic data at the acquisition step as well as for adding new information at further processing steps, including edition commands or echo classifications resulting from various analyses. In this way, the raw data are kept unaltered and are always available to the scientist for new applications, while the data processing information that is required for proper interpretation remains attached to the file. The format is thus useful for storing hydroacoustic data in large data banks as well as for data exchange at various levels of processing among the scientific community.

THE *HAC* STANDARD FORMAT TUPLE TYPES

To qualify as an *HAC* standard hydroacoustic data format, a file: 1) must start with the code 172 (=hexadecimal 0xAC) stored in a ULONG* word, to identify the bytes encoding mode of the computer platform, and 2) must contain five basic tuple types (Fig. 1, shaded bold boxes). These essential tuples are: the *signature tuple*, the *echosounder tuple*, the *channel tuple*, the *ping tuple*, and the *position tuple*. The *start of run* and *end of run tuples*, the *end of file tuple* and the *general threshold tuple* (Fig. 1, clear bold boxes) are also strongly recommended. The ping tuple, the channel tuple and the echosounder tuple are hierarchically related by parent/child relationships. A variety of optional satellite tuples (Fig. 1, thin boxes) complete the possible file content.

HAC TUPLE TYPES: CODE ALLOCATION

The tuple type field is 16-bit long (USHORT), which means that 65536 different tuples types can be defined (0 to 65535). The tuple type codes reserved for the tuple classes and types of the *HAC* standard format are given in table 1. Note that other tuple types could easily be added to handle new information. This could be either survey data (codes from 0 upward) or file structure description data (codes from 65535 backward) to help file checking or browsing. The last code of the various ranges of codes defined in table 1 should be reserved for coding expansion using subcodes (e.g. see temporary tuple 1, table 25).

---

* See definitions section

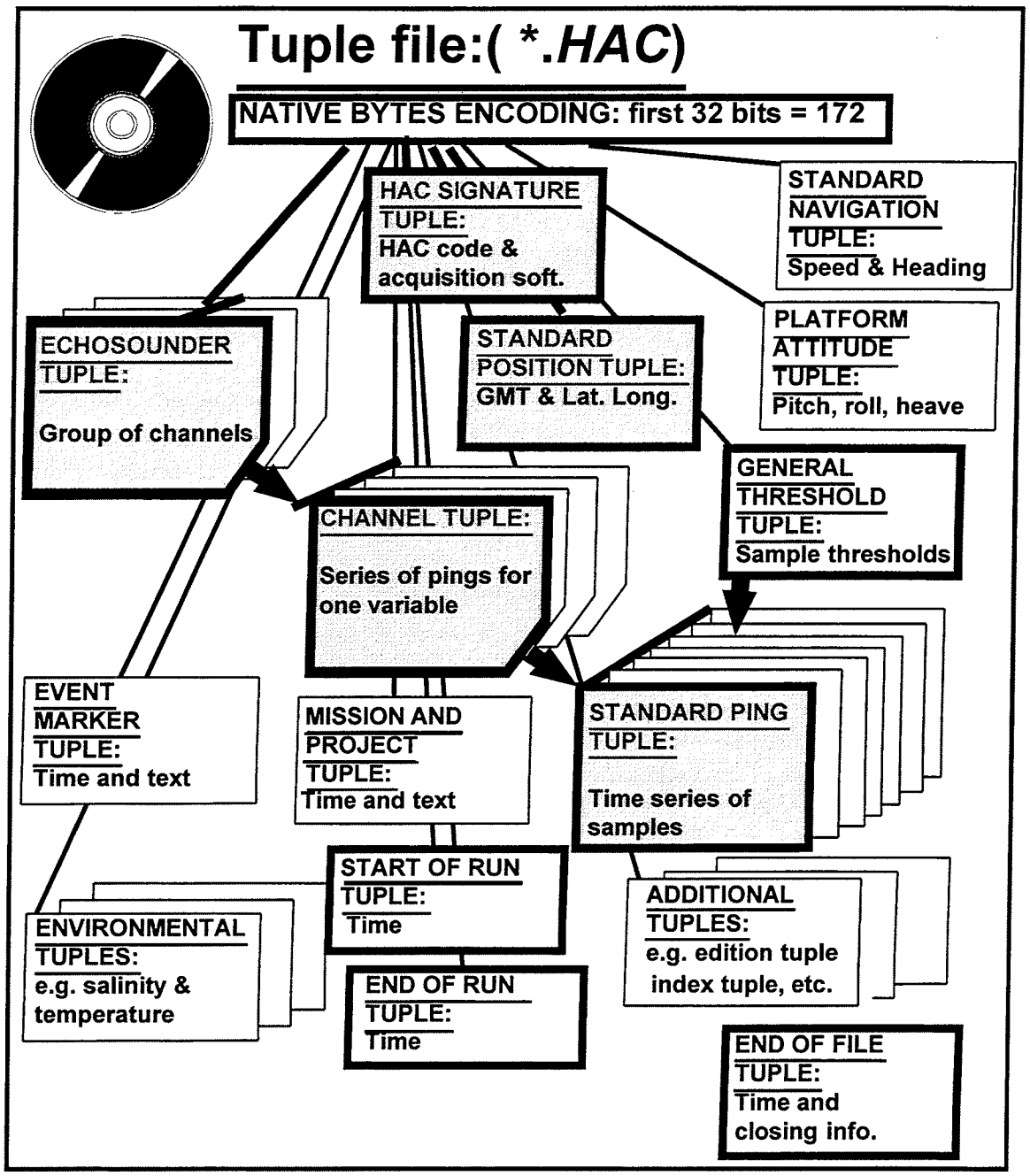

Figure 1. Organization of the information by tuples in the *HAC* standard hydroacoustic data format. The shaded tuples are required, the others are optional.

Table 1. Code numbers reserved as identifiers for the various tuple types of the *HAC* standard hydroacoustic data format. Shaded areas indicate what is defined in the present version 1.0 of the format.

Tuple class reserved		**Tuple type reserved**		**Remarks**
Name	**Range of code numbers**	**Name**	**Code number**	
	...			
Mission and project tuples	10 -19	Mission and project	10	Time, data and text
Position tuples	20 - 29	Standard position	20	Geographic and time references
Navigation tuples	30 - 39	Standard navigation	30	Speed and heading
Platform attitude tuples	40 - 49	Platform attitude	40	Pitch, roll and heave of platform
	...			
Echosounder tuples	100 - 999	Biosonics 102	100	Biosonics range: [100 - 199]
		Simrad EK-500	200	Simrad range: [200 - 299]
				Micrel range: [300 - 349]
Channel tuples	1000 - 9999	Biosonics 102	1000	Biosonics range: [1000 - 1999]
		Simrad EK-500.	2000	Simrad range: [2000 - 2999]
				Micrel range: [3000 - 3499]
Ping tuples	10000 -10099	Standard ping U-32	10000	Time series of samples. Uncompressed 32-bit sample format range: [10000 - 10009]
		Ping U-32-16-angles	10001	Time series of split-beam phase angles of samples.
		Ping C-32	10010	Time series of samples. Compressed 32-bit sample format range: [10010 - 10019]
		Ping CE-32	10020	Time series of samples. Compressed and encoded 32-bit sample format range: [10020 - 10029]
		Ping U-16	10030	Time series of samples. Uncompressed 16-bit sample format range: [10030 - 10039]

Tuple class reserved		Tuple type reserved		Remarks
Name	**Range of code numbers**	**Name**	**Code number**	
		Ping U-16-angles	10031	Time series of split-beam phase angles of samples.
		Ping C-16	10040	Time series of samples. Compressed 16-bit sample format range: [10040 - 10049]
		Ping CE-16	10050	Time series of samples. Compressed and encoded 16-bit sample format range: [10050 - 10059]
Threshold tuples	10100 - 10109	General threshold	10100	Constant and time-varied threshold (TVT)
Event marker tuples	10110 - 10119	Event marker	10110	Time and text
	...			
Environmental tuples	11000 - 11999	STD profile	11000	Salinity, temperature and depth
Edition tuples	12000 - 12999			Reserved for new tuples under development
Classification tuples	13000 - 13999			Reserved for new tuples under development
	...			
Temporary or private tuples	65396 - 65405	Temporary tuple 1	65396	Temporary or testing tuples, or tuples specific to a particular or private application
Index tuples	65406 - 65515	Index 1	65406	List of the tuple types present in the *HAC* file
Opening and closing file tuples	65516 - 65525	Start of run	65516	Traces the opening and appending to the file
		End of run	65517	Traces the closing to the file
End of file tuples	65526 - 65634	End of file	65534	Must be the last tuple of a tuple file
HAC signature tuple	65635	*HAC* signature	65535	*HAC* file signature

ACKNOWLEDGMENTS

The initial ideas for this format came from the hydroacoustic standard data format workshop held at MLI in December 1995. The participants were: Chris Lang, Dan Miller and George Rose from DFO/Northwest Atlantic Fisheries Center; Jérome Benoit, Martin Castonguay, Ian McQuinn, Yves Samson, Yvan Simard and Daniel Thibault from DFO/Maurice Lamontagne Institute; Howard Edel from DFO/Headquarters; David Marino from Biosonics; John Gillis from Simrad; Bob Asplin from Simrad/Mesotech; Peter Simpkin from INRS-Océanologie/Université du Québec à Rimouski. Howard Nes, from Simrad, also significantly contributed to the standard format. Comments from Robert Kieser, from DFO/Pacific Biological Station, were appreciated. We thank Laure Devine, from DFO/Maurice Lamontagne Institute, for text revision.

DEFINITIONS

A/D:	Analog to digital converter
ANSI C Standard time:	Time elapsed since midnight (00:00:00) 1 January 1970, in seconds
CH1:	CH1 is the DFO/NHP/DAT multi-sounder multi-channel Windows95 acquisition software that uses the *HAC* standard data format.
DOUBLE:	64-bit signed word
DSP:	Digital signal processor
FLOAT:	floating point number
Hardware channel:	Physical channel from which the data are coming
LONG:	32-bit signed word
RLE	Run-Length Encoding method of data compression
SHORT:	16-bit signed word
Software channel:	Virtual channel describing a particular type of data, see channel tuple
ULONG:	32-bit unsigned word
USHORT:	16-bit unsigned word

THE *HAC* STANDARD FORMAT TUPLE TYPES:

DETAILED DESCRIPTION

TUPLE SYNTAX RULES

All tuple types must have the following four required fields. The tuple types always begin by a field that gives the tuple data size. The second field is always the tuple type code, which is analogous to the tag of many file types containing mixed information. Tuple data fields follow. Then comes a tuple attribute field which gives some attributes to the tuple (e.g. original tuple, edited tuple). The closing tuple field, called the tuple backlink, gives the tuple size (which is always 10 bytes longer than the tuple data size, to account for to the length of the three fields: tuple data size, tuple type, and tuple backlink). It is included to ease backward scrolling within the file. The first tuple in an *HAC* file must be the *HAC* signature tuple. The last tuple must be the end of file tuple. The index tuples should be ordinarily placed immediately before the end of file tuple

N.B. The standard hydroacoustic data format always starts with the number 172 (= hexadecimal 0xAC, for ACoustics) stored in 32 bits. This is to allow data access programs to distinguish whether the bytes are stored using the LSB (least significant byte; e.g. Intel) or MSB (most significant byte; e.g. HP, Sun, Motorola) order, which varies among computer platforms. Data access programs should make sure that this number is properly decoded; byte swapping is needed to properly decode the file if the byte order of the native platform differs from the data access platform. FLOATS are not used to keep coding simple, to prevent misinterpretations from different computer platforms, and to preserve the dynamic ranges of the fields. Since some computer platforms (except Intel) require that LONGs (4 bytes) be stored at a byte address that is a multiple of 4, all LONGs in *HAC* format are stored at offset bytes that are multiples of 4.

ECHOSOUNDER TUPLE

An echosounder is defined as a group of channels. The echosounder tuple contains the information that is common to all channels of the group. The echosounder tuple is machine specific; the tuple type code is therefore specific to the machine. If several machines are operated simultaneously, then several echosounder tuples will exist. (The range of values reserved for encoding the echosounder tuple type are 100 - 999).

Table 2. Echosounder tuple for the Biosonics Model 102

Echosounder tuple for the Biosonics Model 102, with signal acquired via A/D or A/D-DSP boards, using, for example, the acquisition software CH1 of DFO. This is a two-frequency dual-beam analog echosounder that alternates frequency from ping to ping in the multifrequency mode. This tuple type template could be used for similar analog echosounders.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 62 bytes	byte	62
4	Tuple type	2	USHORT	Tuple type code: **100**. This is the tuple type code for the Biosonics Model 102 echosounder. (Tuples 100 - 199 are reserved for Biosonics echosounders).	unitless	100
6	Number of software channels	2	USHORT	Number of software channels associated with this echosounder.	unitless	[1 - 65535]
8	Echosounder document identifier	4	ULONG	Unique identification number for the echosounder document (i.e. the group of channels). The channels are tied to the echosounder by the echosounder document identifier, which is repeated in the channel tuples.	unitless	[0 - 4294967296]
12	Sound speed	2	USHORT	Speed of sound. **N.B.** Sound speed and sound speed profiles could also be computed from environmental tuples.	0.1 $m\ s^{-1}$	[0.0 - 6553. 5 $m\ s^{-1}$] In water: [1450.0 - 1550. 0 $m\ s^{-1}$]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
14	Ping interval	2	USHORT	Interval between 2 pings. In the multifrequency mode of the Biosonics 102, the acoustic frequency is alternating every ping.	0.01 s	[0.00 - 655.35 s] (= up to 10.92 min)
16	Transmitter attenuation setting	2	SHORT	The attenuation factor of the transmitter corresponding to the echosounder setting. This setting corresponds to a given source level recorded in the source level field of the channel tuple.	0.1 dB	[-3276.8 to +3276.7 dB] Biosonics 102 options: [-13, -10, -6, -3, 0 dB]
18	Multiplexing mode	2	USHORT	The operating transmitter mode of the echosounder. **0** = F1/X1 (frequency 1 on transducer 1) **1** = F2/X2 (frequency 2 on transducer 2) **2** = Ext. (transmitter and transducer controlled externally) **3** = F1/F2 multiplexed F1/X1 and F2/X2 **4** = F1/X2 (frequency 1 on transducer 2) **5** = F2/X1 (frequency 2 on transducer 1)	unitless	[0 - 65535] Biosonics 102 options: [0, 1, 2, 3, 4, 5]
20	Blanking at TVG max range	2	SHORT	The gain operating mode after the TVG max. range, from the blank at range switch **0** = normal mode, the gain is maintained constant at the value reached at the TVG max range. **1** = blank at range mode: the gain drops to zero at the TVG max range	unitless	[0 - 65535] Biosonics 102 options: [0; 1]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
22	TVG max. range	2	USHORT	The range up to which the TVG is applied. Biosonics 102 echosounders normally apply TVG over the following two decade ranges: 1.25 - 125 m, 2.5 - 250 m, 5 - 500 m or 10 - 999.9 m, depending on hardware setting. TVG is computed from the transducer face and applied from the min. range up to the max. range. This max. range can, however, be limited to a smaller range by a range switch, whose setting would give the value of the present field. After this TVG max. range the gain is either maintained constant at the value reached at this range or dropped to zero if the TVG blank at range switch is on.	0.1 m	[0.0 - 6553.5 m] Biosonics 102 options: [0.0 - 999.9 m]
24	Blanking up to range	2	USHORT	Blanking range up to which the receiver output is blanked to zero. It is often set to the TVG min. range of either, 1.25, 2.5, 5.0 and 10.0 m. A blanking range of 0 sets the TVG off.	0.1 m	[0.0 - 6553.5 m] Biosonics 102 options: [0.0 - 999.9 m]
26	Calibrator signal	2	SHORT	Calibrator signal sent to the receiver from the echosounder setting: **0** = off, no calibrator signal is sent (This is the usual setting when echosounding).	1 dB	[-32768 to +32767 dB] Biosonics 102 options: [-40, -20, 0 , +20 dB]
28	Calibrator mode	2	USHORT	Type of sound wave sent to the receiver: **0** = pulse, a pulse repeated at the interval given by the separation field below **1** = constant wave	unitless	[0 - 65535] Biosonics 102 options: [0 or 1]
30	Calibrator separator	2	USHORT	Separation distance for the calibrator sound pulse when this mode is selected (previous field).	0.1 m	[0.0 - 6553.5 m] Biosonics 102 options: [0.0 -99.9 m]
32	Remarks	30	CHAR	Character string comment, up to 30 characters. This field could be used to store the echosounder serial number.	ASCII char.	30 characters

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
62	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
64	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
68	Backlink	4	ULONG	Tuple size: 72 bytes	byte	72

Table 3. Echosounder tuple for the Simrad EK-500.

This is a three-frequency split-beam digital echosounder that transmits on all frequencies at the same time. The following tuple applies to the different versions of the Simrad EK-500, up to version 5.2.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 70 bytes	byte	70
4	Tuple type	2	USHORT	Tuple type code: **200**. This is the tuple type code for the Simrad EK-500. (Tuples 200 - 299 are reserved for Simrad echosounders).	unitless	200
6	Number of software channels	2	USHORT	Number of software channels associated with this echosounder.	unitless	[1 - 65535]
8	Echosounder document identifier	4	ULONG	Unique identification number for the echosounder document (i.e. the group of channels).	unitless	[0 - 4294967296]
12	Sound speed	2	USHORT	Speed of sound. **N.B.** Sound speed and sound speed profiles could also be computed from environmental tuples.	0.1 m s^{-1}	[0.0 - 6553. 5 m s^{-1}] In water: [1450.0 m s^{-1} - 1550. 0 m s^{-1}]
14	Ping mode	2	USHORT	Ping mode: **0** = off **1** = normal **2** = External (trigged from an external source)	unitless	[0 - 65535] EK-500 options [0, 1, 2]
16	Ping interval	2	USHORT	Interval between 2 pings.	0.01 s	[0.00 - 655.35 s] (= up to 10.92 min)
18	Transmit power	2	USHORT	Nominal output power: **0** = normal **1** = reduced (The transmit power is reduced from its nominal value by 20 dB on all transceiver channels).	unitless	[0 - 65535] EK-500 options: [0 or 1]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Noise margin	2	USHORT	The margin to add to the system noise to threshold out the samples.	1 dB	[0 to 65535 dB] EK-500 options: [0 to 40 dB]
22	Sample range	2	USHORT	Range for the sample angle and sample power telegrams.	1 m	[0 m - 65535 m] EK-500 options: [0 - 10000]
24	Super layer: Type	2	USHORT	Super layer type: **0** = off **1** = surface **2** = bottom **3** = pelagic (This layer type must be chosen to get values from the Ethernet port when the bottom is not found [e.g. for passive acoustics])	unitless	[0 - 65535] EK-500 options: [1, 2, 3, 4]
26	Super layer: Number	2	USHORT	Layer number for the super layer.	unitless	[0 - 65535] EK-500 options: [1- 10]
28	Super layer: Range	2	USHORT	Super layer thickness.	0.1 m	[0.0 - 6553.5 m] EK-500 options: [0.0 - 1000.0 m]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
30	Super layer: Start	4	LONG	Beginning depth of super layer relative to the transducer depth or the detected bottom. Negative values indicate starting depth below the reference depth.	0.1 m	[-214748364.8 to +214748364.7 m] EK-500 options: [-10.0 - 9999.9 m]
34	Super layer: Margin	2	USHORT	Margin relative to the transducer depth or the bottom, to stop the super layer.	0.1 m	[0.0 - 6553.5 m] EK-500 options: [0.0 - 10.0 m]
36	Super layer: Sv threshold	2	SHORT	Threshold to accept Sv samples.	1 dB	[-32768 to +32767 dB] EK-500 options: [-100 to 0 dB]
38	EK-500 version	4	ULONG	EK-500 version number.	0.00	[0.00 - 655.35] Present range: [0.00 - 5.20]
42	Remarks	30	CHAR	Character string comment, up to 30 characters. This field could be used to store the echosounder serial number.	ASCII char.	30 characters
72	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
76	Backlink	4	ULONG	Tuple size: 80 bytes	byte	80

CHANNEL TUPLE

This tuple type is machine- and channel-specific; the tuple type code is therefore related to the machine and to the hardware and software channels and. Virtual channels can exist (e.g. an echo-classification channel that combines the information from four other channels), and they would have their own tuple type code. This tuple type should include an image of the machine settings and calibration parameters for the given channel. Besides the four fields common to all tuples, the 3 following fields are required: the software channel identifier, the echosounder document identifier, and the sampling rate. (The range of values reserved for the channel tuple type fields is 1000 - 9999.)

Table 4. Channel tuple for the Biosonics Model 102.

Channel tuple for the Biosonics model 102, with signal acquired via an A/D-DSP boards with the acquisition software CH1 of DFO/NHP/DAT. This tuple type template could be used for similar analog echosounders.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 98 bytes.	byte	98
4	Tuple type	2	USHORT	Tuple type code: **1000.** This is the tuple type code for the Biosonics 102. (Tuple type codes 1000 - 1999 are reserved for Biosonics echosounders).	unitless	1000
6	Software channel identifier	2	USHORT	Unique identifier for this software data channel This identifier must be unique for the whole file in order to associate the pings to their proper parent channel. **N.B.** This is not the hardware channel number.	unitless	[0 - 65535]
8	Echosounder document identifier	4	ULONG	Identification number for the parent echosounder document (i.e. the group of channels) to which this data channel belongs. It is the echosounder document identifier field of the echosounder tuple.	unitless	[0 - 4294967295]
12	Sampling rate	4	ULONG	Digitization rate for this channel.	sample s^{-1}	[0 - 4294967295 sample s^{-1}]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
16	Type of data sample	2	USHORT	Type of data sample: **0** = Volts **1** = Sv (Scattering volume in dB) **2** = TS (Target strength of single targets in dB) **3** = Offaxis mechanical angles of single targets Others to be defined.	unitless	[0 - 65535] Presently for the Biosonics 102: [0, 1, 2, 3]
18	Time varied gain mode	2	USHORT	Time-varied gain (TVG) applied for this channel: **0** = 20 log R TVG **1** = 40 log R TVG **N.B.** This TVG is applied from a min. range up to the TVG max. range field of the echosounder tuple. When this TVG max. range is null or smaller than the blanking up to range, or the blanking up to range is set to zero, no TVG is applied.See pertinent fields in table 2.	unitless	[0 - 65535] Presently for the Biosonics 102: [0 or 1]
20	Transceiver channel number	2	USHORT	Hardware channel number from which the data are coming. It is convenient to use the same channel numbers as from the echosounder. The Biosonics 102 has the narrow-beam signal on channel 1, the wide-beam signal on channel 2, and the simultaneous 20 log R narrow-beam on channel 3. When the 40 log R TVG switch is on, channels 1 & 2 are the 40 log R signals. **N.B.** This field is not the software channel number.	unitless	[0 - 65535] Presently for the CH1 DSP-A/D: [1, 2, 3]
22	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
24	Acoustic frequency	4	ULONG	Acoustic frequency.	Hz	[0 - 4294967295 Hz] Fisheries acoustics range: [100 - 1000000 Hz]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
28	Installation depth of transducer.	4	ULONG	Installation depth of transducer relative to the sea surface.	0.01 m	[0.00 - 42949672.95 m] Working range: [0.00 - 999.99 m]
32	Alongship angle offset of the transducer face	2	SHORT	Mechanical offset angle of the transducer face relative to the horizontal in the alongship plane. Negative is below the horizontal and 0 degree is in the fore direction.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-360.0 to +360.0 degree]
34	Athwartship angle offset of the transducer face	2	SHORT	Mechanical offset angle of the transducer face relative to the horizontal in the athwartgship plane. Negative is below the horizontal and 0 degree is in the starboard direction.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-360.0 to +360.0 degree]
36	Alongship angle offset of the main axis of the acoustic beam	2	SHORT	Mechanical offset angle of the main axis of the acoustic beam of the transducer relative to the vertical in the alongship plane. Negative is in the aft direction. Zero (0) is perpendicular to the transducer face.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-20.0 to +20.0 degree]
38	Athwartship angle offset of the main axis of the acoustic beam	2	SHORT	Mechanical offset angle of the main axis of the acoustic beam of the transducer relative to the vertical in the athwartship plane. Negative is in the port direction below the horizontal. Zero (0) is perpendicular to the transducer face.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-20.0 to +20.0 degree]
40	Absorption of sound	2	USHORT	Absorption of sound (α) in the propagation medium.	0.01 dB km^{-1}	[0.00 - 655.35 dB km^{-1} Practical range: [0.00 - 300.00 dB km^{-1}]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
42	Pulse length	2	USHORT	Duration of the transmitted pulse.	0.1 ms	[0.0 ms - 6553.5 ms] Biosonics 102 range [0.1 ms - 9.9 ms]
44	Bandwidth	2	USHORT	Transceiver specific bandwidth.	0.01 kHz	[0.00 - 655.35 kHz] Biosonics 102 options: [1.25, 2.50, 5.00, 10.0 kHz]
46	Calibration source level	2	USHORT	Source level (SL). **N.B.** The attenuation factor of the transmitter, given in the echosounder tuple, is added to this field to get the effective source level.	0.01 dB μPa @ 1 m	[0.00 to 655.35 dB] Practical range: [150.00 - 250.00 dB]
48	3 dB beamwidth of the transducer beam	2	USHORT	Half power (3 dB) beamwidth of the transducer beam (narrow or wide)	0.1 degree	[0.0 - 6553.5 degree] Practical range: [1.0 to 50.0 degree]
50	Beam pattern	2	USHORT	Beam pattern factor (expected value of b^2) for this transducer beam (narrow or wide). (unitless). N. B. The directivity index (DI) in dB is 10 log the present field and 10 log ψ = -DI + 7.7 dB.	0.000001	[0.000000 to 0.065535] Practical range: [0.000100 to 0.009000]
52	Wide-beam drop-off	2	USHORT	The wide-beam drop-off (d) is the factor relating the narrow-beam directivity in dB (Bn) to the difference between the narrow-beam and wide-beam (Bw) directivities in dB: Bn = d (Bn-Bw). It describes the decrease in wide beam directivity over the angular range of the narrow beam.	0.0001	[0.0000 - 6.5535] Practical range: [1.0000 to 1.5000]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
54	Calibration receiving sensitivity	2	SHORT	Calibration receiving sensitivity of the transducer for this TVG-amplified data channel. **N.B.** The receiver gain **is included** in the value of this field, which is the VR of the sonar equation.	0.01 dB v /μPa @ 1 m	[-327.68 to +327.67 dB] Practical range: [-200.00 to -100.00 dB]
56	Receiver gain	2	SHORT	Receiver gain of the echosounder. **N.B.** The receiver gain **is included** in the receiving sensitivity value to get the VR of the sonar equation.	0.01 dB	[-327.68 to +327.67 dB] Biosonics 102 options: [-18, -12, -6, 0, +6, +12, +18, +24]
58	Bottom detection: minimum level	2	SHORT	Level for the bottom detection in the units selected in the above field "Type of sample data [0, 1 or 2]".	0.001 volts, Sv and TS in 0.01 dB	For volts: [-32.768 to +32.767 volts] Practical range: [2.500 to 15.000 volts]; For Sv and TS: [-327.68 to +327.67 dB] Practical range: [-150.00 to 0.00 dB]
60	Bottom window min.	4	ULONG	Minimum depth for bottom detection window.	0.01 m	[0.00 - 42949672.95 m] Working range: [0.00 - 999.99 m]
64	Bottom window max.	4	ULONG	Maximum depth for bottom detection window in m. For the CH1 software this is also the maximum depth up to which data will be acquired.	0.01 m	[0.00 - 42949672.95 m] Working range: [0.00 - 999.99 m]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
68	Remarks	30	CHAR	Character string comment, up to 30 characters. This field could be used to store the transducer serial number.	ASCII char.	30 characters
98	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
100	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
104	Backlink	4	ULONG	Tuple size: 108 bytes.	byte	108

Table 5. Channel tuple for the Simrad model EK-500.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 98 bytes.	byte	98
4	Tuple type	2	USHORT	Tuple type code: **2000**. This is the tuple type code for the Simrad EK-500 raw data. (Tuples 2000 - 2999 are reserved for Simrad echosounders).	unitless	2000
6	Software channel identifier	2	USHORT	Unique identifier for this software data channel This identifier must be unique for the whole file in order to associate the pings to their proper parent channel **N.B.** This is not the hardware channel number.	unitless	[0 - 65535]
8	Echosounder document identifier	4	ULONG	Identification number for the parent echosounder document (i.e. the group of channels) to which this data channel belongs. It is the echosounder document identifier field of the echosounder tuple.	unitless	[0 - 4294967295]
12	Sampling rate	4	ULONG	Digitization rate for this channel. Fixed according to acoustic frequency.	sample s^{-1}	[0 - 4294967295 sample s^{-1}]
16	Type of data sample	2	USHORT	Type of data sample: **0** = electrical phase angle from the split-beam analysis **1** = power (raw Sv before the TVG) **2** = Sv (Scattering volume) **3** = TS (Target strength)	unitless	[0 - 65535] Presently: [0, 1, 2, 3]
18	Transceiver channel number	2	USHORT	EK-500 transceiver (1, 2 or 3).	unitless	[0 - 65535] Presently: [1, 2, 3]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Acoustic frequency	4	ULONG	Acoustic frequency.	Hz	[0 - 4294967295 Hz] Fisheries acoustics range: [100 -1000000 Hz]
24	Installation depth of transducer.	4	ULONG	Installation depth of transducer relative to the sea surface.	0.01 m	[0.00 - 42949672.95 m] EK-500 range: [0.00 - 999.99 m]
28	Alongship angle offset of the transducer face	2	SHORT	Mechanical offset angle of the transducer face relative to the horizontal in the alongship plane. Negative is below the horizontal and 0 degree is in the fore direction.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-360.0 to +360.0 degree]
30	Athwartship angle offset of the transducer face	2	SHORT	Mechanical offset angle of the transducer face relative to the horizontal in the athwartgship plane. Negative is below the horizontal and 0 degree is in the starboard direction.	0.1 degree	[-3276.8 to +3276.7 degree] Working range: [-360.0 to +360.0 degree]
32	Alongship angle offset of the main axis of the acoustic beam	2	SHORT	Mechanical offset angle of the main axis of the acoustic beam of the transducer relative to the vertical in the alongship plane. Negative is in the aft direction. Zero (0) is perpendicular to the transducer face.	0.1 degree	[-3276.8 to +3276.7 degree] EK-500 range: [-20.0 to +20.0 degree]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
34	Athwartship angle offset of the main axis of the acoustic beam	2	SHORT	Mechanical offset angle of the main axis of the acoustic beam of the transducer relative to the vertical in the athwartship direction. Negative is in the port direction. Zero (0) is perpendicular to the transducer face.	0.1 degree	[-3276.8 to +3276.7 degree] EK-500 range: [-20.0 to +20.0 degree]
36	Absorption of sound	2	USHORT	Absorption of sound (α) in the propagation medium.	0.01 dB km^{-1}	[0.00 - 655.35 dB km^{-1}] Practical range: [0.00 -300.00 dB km^{-1}]
38	Pulse length mode	2	USHORT	This field indicates the selected transceiver specific duration of the transmitted pulse: **0** = short **1** = medium **2** = long	unitless	[0 - 65535] EK-500 options: [0;1; 2]
40	Bandwidth mode	2	USHORT	This field indicates the selected transceiver specific bandwidth: **0** = narrow **1** = wide N.B. Auto: this mode is not coded because the choice (narrow or wide) made by the EK-500 is indicated in the EK-500 telegram that the acquisition program reads.	unitless	[0 - 65535] EK-500 options: [0;1]
42	Max. power	2	USHORT	Transmit power referred to the transducer terminals.	watt	[0 - 65535] EK-500 range: [1 watt - 10000 watt]
44	Alongship angle sensitivity	2	USHORT	The electrical phase angle in degrees for one mechanical phase angle in degrees in the fore-and-aft direction, specific to the split-beam transducer. A value of 1.0 indicates that the electrical angles are in units of mechanical angles.	0.1	[0.0 - 6553.5] EK-500 range: [0.0 - 100.0]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
46	Athwartship angle sensitivity	2	USHORT	The electrical phase angle in degrees for one mechanical phase angle in degrees in the starbord-and-port direction, specific to the split-beam transducer. A value of 1.0 indicates that the electrical angles are in units of mechanical angles.	0.1	[0.0 - 6553.5] EK-500 range: [0.0 - 100.0]
48	Alongship 3 dB beamwidth of the transducer	2	USHORT	Half power (3dB) beamwidth of the transducer in the alongship plane.	0.1 degree	[0.0 - 6553.5 degree] EK-500 range: [1.0 to 50.0 degree]
50	Athwartship 3 dB beamwidth of the transducer	2	USHORT	Half power (3dB) beamwidth off the transducer in the athwartship plane.	0.1 degree	[0.0 - 6553.5 degree] EK-500 range: [1.0 to 50.0 degree]
52	Two way beam angle	2	SHORT	Equivalent two way beam opening solid angle: [=10 log ((β1 * β2) / 5800)], where β1 is the longitudinal beamwidth in degrees and β2 is the transversal beamwidth in degrees). NB: Directivity index in dB: DI = 10 log (2.5/ (sin(β1 / 2) * sin(β2 / 2))). (see EK-500 user manual).	0.01 dB	[-327.68 to +327.67 dB] EK-500 range: [-99.90 to 0.00 dB]
54	Calibration transducer gain	2	USHORT	Peak transducer gain used during computation of the data sample corresponding to the above-selected "type of data sample" (either Sv or TS) (see EK-500 user manual).	0.01 dB	[0.00 dB - 655.35 dB] EK-500 range: [0.00 to 99.90 dB]
56	Bottom detection: minimum level	2	SHORT	Volume backscattering level for the bottom detector`s back search function.	0.01 dB	[-327.68 to +327.67 dB] EK-500 range: [-80.00 to 0.00 dB]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
58	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
60	Bottom window min. depth	4	ULONG	Minimum depth for bottom detection window.	0.01 m	[0.00 - 42949672.95 m] EK-500 range: [0.00 - 10000.00 m]
64	Bottom window max. depth	4	ULONG	Maximum depth for bottom detection window.	0.01 m	[0.00 - 42949672.95 m] EK-500 range: [0.00 - 15000.00 m]
68	Remarks	30	CHAR	Character string comment, up to 30 characters. This field could be used to store the transducer serial number.	ASCII char.	30 characters
98	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
100	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
104	Backlink	4	ULONG	Tuple size: 108 bytes.	byte	108

STANDARD PING TUPLE U-32

A ping is a time-series of samples. Samples are raw data, or they may come from processed information. Samples may be encoded with many formats. In the standard ping tuple U-32, the sample value and the sample sequence are encoded using LONG and ULONG words respectively. See tables 19-24 for other ping tuple types using different sample formats and for split-beam phase angle data formats. The different ping tuple types should be compared for their resolution and their efficiency to store and recuperate the information. (Tuple type codes 10000 - 10099 are reserved for ping tuples).

Table 6. Standard ping tuple U32.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10000**	unitless	10000
6	Time fraction †	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time†	4	ULONG	Time of the transmitted pulse. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder. Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]

† To not use FLOATS , the time, which could be stored with precision by a DOUBLE, is divided here in two fields, s-integer (ANSI C standard time) and s-fraction (to the 0.1 ms). There are other ways of encoding time that require less space but we have chosen this way for clarity and safety.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered, under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not dete cted. Negative values are reserved for future use.	0.001 m	[-2147483.648 m to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	Sample sequence number	4	ULONG	Sample sequence number since the beginning of the ping (samples < threshold contribute to the sequence count).	unitless	[0 - 4294967295]
28	Sample value (> threshold)	4	LONG	Sample value on 32 bits. (For phase angles from the split-beam analysis, see Ping tuple U-32-16-angles, table 19)	depending on the "type of data sample" of the channel tuple: 0.000001 volts, or 0.000001 dB, for Sv or TS	For volts: [-2147.483648 to +2147.483647 volts] Practical range: [0.000000 to 25.000000 volts]; for Sv and TS: [-2147.483648 to 2147.483647 dB] Practical range: [-150.000000 to 0.000000 dB];
...	continued					
...	Sample sequence number	4	ULONG	idem		
...	Sample value (> threshold)	4	LONG	idem		
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes)	byte	[40 - 4 giga]

STANDARD POSITION TUPLE

This tuple type is used to store the position. The data can be collected with a refreshing rate different from the ping rate. The format below saves the time difference between the sampling time of the ping tuple and GMT. (Tuple type codes 20 - 29 are reserved for position tuples).

Table 7. Standard position tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 26 bytes	byte	26
4	Tuple type	2	USHORT	Tuple type code: **20**	unitless	20
6	Time fraction	2	SHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time at which the position was taken by the positioning system).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time at which the position was taken by the positioning system. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time. The difference with the GPS time gives the lag of the local time relative to the universal time.	s	[0 - 4294967295 s] (= up to year 2106)
12	GPS time (GMT)	4	ULONG	Universal time (GMT), given by the GPS, at which the position was taken by the positioning system. Absolute time, in seconds, elapsed since midnight (00:00:00) 1 January 1970.	s	[0 - 4294967295 s] (= up to year 2106)
16	Positioning system	2	USHORT	Positioning system used: **0** = Loran C **1** = GPS **2** = DGPS 0ther systems to be coded.	unitless	[0 - 65535] Presently: [0; 1; 2]
18	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Latitude	4	LONG	Latitude, in degrees. Negative values are in the southern hemisphere.	0.000001 degree (= precision of ~ 2 cm)	[-214.7483647 to 214.7483647 degree] Practical range: [-90.000000 to +90.000000 degree]
24	Longitude	4	LONG	Longitude, in degrees. Negative values are western coordinates.	0.000001 degree (= precision of ~ 2cm)	[-214.7483647 to 214.7483647 degree] Practical range: [-180.000000 to +180.000000 degree]
28	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
32	Backlink	4	ULONG	Tuple size: 36 bytes	byte	36

GENERAL THRESHOLD TUPLE

This tuple type is used to store the threshold used during the acquisition to exclude samples. It could be either constant threshold or time-varied thresholds (TVT). This threshold applies to the pings that are collected after the threshold setting time, until a new threshold tuple updates the threshold information. (Tuple type codes 10100 - 10109 are reserved for threshold tuples).

Table 8. General threshold tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 34 bytes	byte	34
4	Tuple type	2	USHORT	Tuple type code: **10100**	unitless	10100
6	Time fraction	2	USHORT	Time of threshold setting. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of threshold setting. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for the software data channel to which this threshold tuple is associated. The equation used for the TVT evaluation should be dependent on channel data units (volts, Sv or TS, etc.). Some channels may not require any thresholding.	unitless	[0 - 65535]
14	TVG max. range	2	USHORT	Maximum range from the transducer on which the TVG is applied by the echosounder. This is the TVG max. range field of the echosounder tuple (see table 2. Echosounder tuple for the Biosonics 102).	0.1 m	[0.0 - 6553.5 m] Biosonics 102 options: [0.0 - 999.9 m]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
16	TVG min. range	2	USHORT	Minimum range from which the TVG is applied by the echosounder. It is either the hardware determined TVG min range (of either 1.25, 2.5, 5.0 or 10.0 m) or the value of the blank at range field of the echosounder tuple (see pertinent fields in table 2. Echosounder tuple for the Biosonics 102).	0.1 m	[0.0 - 6553.5 m] Biosonics 102 options: [0.0 - 999.9 m]
18	TVT evaluation: Mode	2	USHORT	Time-varied threshold (TVT) evaluation mode: **0** = no TVT, a constant threshold is applied. When the mode is 0 and the "amplification field" is 0, the "offset field" specifies a constant threshold (which could be zero). When the offset is zero, no threshold is applied. **1** = manual (TVT is evaluated on the user's request) **2** = automatic (TVT was evaluated at regular time intervals).	unitless	[0 - 65535] Presently: [0, 1, 2]
20	TVT evaluation: Interval	2	USHORT	Time interval between two TVT evaluations under the automatic TVT evaluation mode of CH1 software.	s	[0 - 65535 s] (= up to 18.2 h)
22	TVT evaluation: No. of pings	2	USHORT	Number of pings over which the TVT is evaluated.	unitless	[0 - 65535]
24	TVT evaluation: Starting TVT ping number	4	ULONG	Starting TVT ping number of the ping series used for the TVT evaluation of this threshold tuple. The ending TVT ping number of the ping series is obtained by adding the above field "TVT evaluation: No. of pings" to the present field.	unitless	[0 - 4294967295]
28	TVT offset parameter or constant threshold	4	ULONG	Either the value of the constant threshold applied or the coefficient C of the TVT formula, given here for the voltage (i.e. not the energy, V^2): (1) for the 20 Log R TVG: A_{20} R $e^{\beta R}$ + C_{20}; and (2) for the 40 Log R TVG: A_{40} R^2 $e^{\beta R}$ + C_{40}; where R is the range, A and C are the estimated coefficients, and β is the sound absorption coefficient in nepers per m (not α, which is in dB). The curve is fitted for the range interval between TVG min. and TVG max. ranges. For Sv or TS, the formula are different.	0.000001	[0.000000 - 4294.967295]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
32	TVT amplification parameter	4	ULONG	Coefficient A of the TVT formula:, given here for the voltage (i.e. not the energy, V^2): (1) for the 20 Log R TVG: A_{20} R $e^{\beta R}$ + C_{20}; and (2) for the 40 Log R TVG: A_{40} R^2 $e^{\beta R}$ + C_{40}; where R is the range, A and C are the estimated coefficients, and β is the sound absorption coefficient in nepers per m (not α, which is in dB). The value zero indicates that a constant threshold is applied. The curve is fitted for the range interval between TVG min. and TVG max. ranges. For Sv or TS, the formula are different.	0.000001	[0.000000 - 4294.967295]
36	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
40	Backlink	4	ULONG	Tuple size: 44 bytes	byte	44

STANDARD NAVIGATION TUPLE

This tuple type is used to store the platform navigation information. (Tuple type codes 30 - 39 are reserved for navigation tuples).

Table 9. Standard navigation tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 18 bytes	byte	18
4	Tuple type	2	USHORT	Tuple type code: **30**	unitless	30
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time at which the navigation reading was taken).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time at which the navigation reading was taken. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Navigation system	2	USHORT	Navigation system used: **0** = mechanical log and gyro **1** = acoustic Doppler log and gyro Other systems to be coded.	unitless	[0 - 65535] Presently [0; 1;]
14	Heading	2	SHORT	Compass heading of the ship. Negative values are counterclockwise relative to North (0°).	0.1 degree	[-3276.8 to +3276.7 degree] Practical range: [-360.0 to +360.0 degree]
16	Navigation speed	2	USHORT	Speed of the ship relative to its immediate surrounding water.	0.001 m s^{-1}	[0.000 - 65.535 m s^{-1}] (= up to 33.7 knots)

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
18	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
20	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
24	Backlink	4	ULONG	Tuple size: 28 bytes	byte	28

PLATFORM ATTITUDE TUPLE

This tuple type is used to store the attitude of the platform. This could be the ship's attitude or the towed body's attitude. (Tuple type codes 40 - 49 are reserved for platform attitude tuples).

Table 10. Platform attitude tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 26 bytes	byte	26
4	Tuple type	2	USHORT	Tuple type code: **40**	unitless	40
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time at which the platform attitude reading was taken).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time at which the platform attitude reading was taken. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique software channel identifier to which this platform attitude information applies. A value of 65535 indicates all channels (e.g. when the platform is the ship or when all transducers are on the same towed body).	unitless	[0 - 65535]
14	Platform referred to	2	USHORT	The platform to which the attitude information is referring: **0** = ship **1** = towed body 1 **2** = towed body 2 Others platforms to be coded.	unitless	[0 - 65535] Presently: [0; 1; 2]
16	Alongship offset	2	SHORT	Distance between the transducer and the reference point of the attitude sensor in the fore and aft direction. Negative values are on the aft side of the reference point of the attitude sensor.	0.01 m	[-327.68 to 327.67 m]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
18	Athwartship offset	2	SHORT	Distance between the transducer and the reference point of the attitude sensor in the starboard and port direction. Negative values are on the port side of the reference point of the attitude sensor.	0.01 m	[-327.68 to 327.67 m]
20	Pitch	2	SHORT	Inclination of the platform relative to the horizontal plane in the fore-and-aft direction. Negative angles are below the horizontal and positive above.	0.1 degree	[-3276.8 to +3276.7 degree] Practical range: [-90.0 to +90.0 degree]
22	Roll	2	SHORT	Inclination of the platform relative to the horizontal plane in the starboard-and-port direction. Negative angles are below the horizontal and positive above.	0.1 degree	[-3276.8 to +3276.7 degree] Practical range: [-90.0 to +90.0 degree]
24	Heave	2	SHORT	Heave of the platform.	0.01 m	[-327.68 to 327.67 m]
26	Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
28	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
32	Backlink	4	ULONG	Tuple size: 36 bytes	byte	36 bytes

EVENT MARKER TUPLE

This tuple type is used to mark events occurring during the acquisition or the edition at the user's request. (Tuple type codes 10110 - 10119 are reserved for event marker tuples).

Table 11. Event marker tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[14 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10110**	unitless	10110
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time at which the event marker was called).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time at which the event marker was called. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
Character string comment. The tuple size (multiple of 4 bytes) will vary with the length of this field.						
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[24 - 4 giga]

ENVIRONMENTAL TUPLES

Environmental tuples are used to store environmental variables describing the conditions in the study area. One tuple could be define for each sensor type, e.g. salinity, temperature, depth, light, meteorological information, refreshed at their own rate. Multi-variableble sensors (e.g. CTD: conductivity, temperature, depth) should have only one tuple. (Tuple type codes 11000 - 11999 are reserved for environmental tuples).

Table 12. STD profile tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[10 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **11000** **Code reserved for the STD profile.**	unitless	11000
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time at which the STD profile started).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time at which the STD profile started. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
	Heading information, temperature, salinity, depth	Formats to be determined, e.g. the DFO/MLI TS8 format for CTD data.				
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[20 - 4 giga]

MISSION AND PROJECT TUPLE

This tuple type is used to store general information on the mission, ship, and research project. (Tuple type codes 10 - 19 are reserved for mission and project tuples).

Table 13. Mission and project tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[10 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10**	unitless	10
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C time to get a time precision of 0.0001 s (Local time of tuple creation).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Local time of tuple creation. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
Information (character string); structure, if any, to be determined. The tuple size (multiple of 4 bytes) will vary with the length of this field. The following fields could be used: Institute name and address, research program, objective, target species, name of the person in charge, his address, telephone, fax and E-mail, platform name (ship, mooring or fixed locations), mission chief, starting date, ending date, sampling area corners (maximum latitude, minimum longitude; minimum latitude, maximum longitude), echosounder model, operating method (single beam echointegration, multifrequency, split-beam, dual-beam single target TS measurements, chirp, wideband), frequency(ies), calibration method and date (standard spheres, hydrophones), transducer mounting (hull, towed body, fixed, etc.), sampling design (systematic, stratified random, adaptive, etc.), data acquisition system, resolution in time and space, data storage medium, noise treatment at acquisition (none, constant threshold, time-varied threshold), TVG applied and effective range (none, analog or digital 20 log R &40 log R), ground truthing method & gear used, no. of sample and depth, environmental variable measured (continuously or at discrete locations), etc.						

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[20 - 4 giga]

INDEX 1 TUPLE

This tuple type gives the list of the different tuple types present in the *HAC* file to help decode the information contained in the file. It should ordinarily be placed at the end of the file, before the end of file tuple. Other more complex index tuples could be created to help speed up the access to particular information in the file. (Tuple type codes 65406 - 65515 are reserved for index tuples).

Table 14. Index 1 tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[22 - 4 giga]
4	Tuple type	2	USHORT	Standard opening tuple type code: **65406**	unitless	65406
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C standard time to get a time precision of 0.0001 s (Tuple creation local time).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Tuple creation local time. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Tuple type list	x2 ...	USHORT	List of tuple types present in the *HAC* file. The list must include the 5 basic tuple types of an *HAC* file: the *HAC* signature tuple (65535), the position tuple ([20 - 29]), the echosounder tuple ([100 - 999]), the channel tuple ([1000 - 9999]), the ping tuple ([10000 - 10099]).	unitless	[0 - 65535]
...	Optional field: Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality) . Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[32 - 4 giga]

START OF RUN TUPLE

This tuple type traces the file opening and appending steps (runs). New start of run tuples are added each time data are appended to the file. These tuples are useful to select some segments of the files such as acquisition runs. (Tuple type codes 65516 - 65525 are reserved for opening and closing of file tuples).

Table 15. Start of run tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 10 bytes	byte	10
4	Tuple type	2	USHORT	Standard opening tuple type code: **65516**	unitless	65516
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C standard time to get a time precision of 0.0001 s (Starting time of the run).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Starting time of the run. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
16	Backlink	4	ULONG	Tuple size: 20 bytes	byte	20

END OF RUN TUPLE

This tuple type traces the file closing steps (runs). New end of run tuples are added each time data acquisition is stopped (end of run). These tuples are useful for selecting some segments of the files such as acquisition runs. (Tuple type codes 65516 - 65525 are reserved for the opening and closing of file tuples).

Table 16. End of run tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 10 bytes	byte	10
4	Tuple type	2	USHORT	Standard opening tuple type code: **65517**	unitless	65517
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C standard time to get a time precision of 0.0001 s (Ending time of the run).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Ending time of the run. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
16	Backlink	4	ULONG	Tuple size: 20 bytes	byte	20

END OF FILE TUPLE

This tuple marks the end of the file. The end of file tuple must be the last tuple in any tuple file. It provides some error checking against a file which has been prematurely truncated. (Tuple type codes 65526 - 65534 are reserved for end of files tuples).

Table 17. End of file tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[14 - 4 giga]
4	Tuple type	2	USHORT	Standard end of file tuple type code: **65534**	unitless	65534
6	Time fraction	2	USHORT	Fraction of a second to add to the following CPU ANSI C standard time to get a time precision of 0.0001 s (Local time at which the file was closed).	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C standard time	4	ULONG	Local time at which the file was closed. ANSI C time given by the CPU clock, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Closing mode	2	USHORT	How the file was closed: **0**: closed manually by the operator command **1**: closed automatically by the program **2**: closed by the program while an error was detected.	unitless	[0 - 65535] Presently: [0; 1]
Space for possible data, structure, if any, to be determined.						
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[24 - 4 giga]

HAC SIGNATURE TUPLE

This tuple identifies the *HAC* standard hydroacoustic data format. The signature tuple confirms the type of tuple file and it should be the first tuple of the file. After reading this tuple, a program can determine if it can interpret the file. (The tuple type code 65535 is reserved for the signature tuple.)

Table 18. *HAC* signature tuple.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: 14	byte	14
4	Tuple type	2	USHORT	Tuple type code: **65535**	unitless	65535
6	*HAC* identifier	2	USHORT	Unique code identifying the *HAC* tuple files. The chosen code was determined from the hexadecimal 0xACAC (decimal 44201), the two first letters of ***AC***oustics).	unitless	44201
8	*HAC* version	2	USHORT	*HAC* format version	0.00	[0.00 - 655.35]
10	Acquisition software version	2	USHORT	Version number of the acquisition software, e.g. 1.00.	0.00	[0.00 - 655.35] Practical range [00.00 - 99.99]
12	Acquisition software identifier	4	ULONG	Unique code identifying the software used to acquire the tuple file. The code chosen for CH1 software of DFO is 3741428908 (hexadecimal 0xDF01ACAC).	unitless	[0 - 4294967295]
16	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to + 2147483647]
20	Backlink	4	ULONG	Tuple size: 24 bytes	byte	24

OTHER PING TUPLE TYPES USING DIFFERENT SAMPLE ENCODING FORMATS

In a ping tuple, samples are encoded with a given format. Different ping tuple types are needed to encode samples with different formats. Some formats are presented below. The different ping tuple types should be compared for their resolution and their efficiency to store and recuperate the information. **N.B.** Other ping tuple types, using other sample encoding formats (e.g. 8 bit values for digital video), could be described and labeled by a code. (Tuple type codes 10000 - 10099 are reserved for ping tuples).

PING TUPLE U-32-16-ANGLES

The ping tuple U-32-16-angles could be used when the type of sample data are phase angles from the split-beam analysis (see table 5. Channel tuple for the Simrad model EK-500) to encode the alongship and athwartship electrical angles of the sample. Here in the ping tuple U-32-16-angles, the sample sequence is encoded using an uncompressed 32-bit format and the encoding of the two angles, with an uncompressed 16-bit format, requires 32 more bits. We have chosen to encode the two angles with two SHORTs instead of two LONGs to save space. (Tuple type codes 10000 - 10009 are reserved for ping tuples using uncompressed 32-bit format for sample encoding).

Table 19. Ping tuple U-32-16-angles.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10001**	unitless	10001
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered, under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	Sample sequence number	4	ULONG	Sample sequence number since the beginning of the ping (samples < threshold contribute to the sequence count)	unitless	[0 - 4294967295]
28	Sample value, alongship electrical phase angle	2	SHORT	Alongship electrical phase angle of the sample from the split-beam analysis. Zero (0) is the main axis of the transducer beam and positive is in the fore direction. See "Alongship angle sensitivity" in table 5 for electrical to mechanical angle conversion (see EK-500 user manual).	0.1 degree	[-3276.8 to 3276.7 degree] Practical range: [-180.0 to 180.0 degree]
30	Sample value, athwartship electrical phase angle	2	SHORT	Athwartship electrical phase angle of the sample from the split-beam analysis. Zero (0) is the main axis of the transducer beam and positive is in the starboard direction. See "Athwartship angle sensitivity" in table 5 for electrical to mechanical angle conversion (see EK-500 user manual).	0.1 degree	[-3276.8 to 3276.7 degree] Practical range: [-180.0 to 180.0 degree]
...	continued					
...	Sample sequence number	4	ULONG	idem		

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
...	Sample value, alongship electrical phase angle	2	SHORT	idem		
...	Sample value, athwartship electrical phase angle	2	SHORT	idem		
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[40 - 4 giga]

PING TUPLE C-32

In the ping tuple C-32, the sample value is encoded using a compressed 32-bit format for the zero series. (Tuple type codes 10010 - 10019 are reserved for ping tuples using compressed 32-bit format for sample encoding).

Table 20. Ping tuple C-32.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10010**	unitless	10010
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative values are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	No. of samples (> threshold) in this ping	4	ULONG	No. of samples (> threshold) in this ping (This information can also be computed from the tuple size).	unitless	[0 - 4294967295]
28	Sample value	4	LONG	Sample value on 32 bit or zero series (< threshold) compressed into RLE samples (the upper 16 bits are set to 1 and the lower 16 bits indicate the no. of zeros + 1; 65536 below threshold values can then be compressed into one RLE sample; no value smaller than -2147418112 or larger than 2147418111 can be encoded).	depending on the "type of data sample" of the channel tuple: 0.000001 volts or 0.000001 dB, for Sv or TS :	For volts: [-2147.418112 to 2147.418111 volts] Practical range: [0.000000 to 25.000000 volts]; For Sv and TS: [-2147. 418112 to 2147.418111 dB] Practical range: [-150.000000 to 0.000000 dB]
...	continued					
...	Sample value	4	LONG	idem		
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[40 - 4 giga]

PING TUPLE U-16

In the ping tuple U-16, the sample value is encoded using an uncompressed 16-bit format. (Tuple type codes 10030 - 10039 are reserved for ping tuples using uncompressed 16-bit format for sample encoding).

Table 21. Ping tuple U-16.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[26 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10030**	unitless	10030
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder. Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative values are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	Sample sequence number	2	USHORT	Sample sequence number since the beginning of the ping (samples < threshold contribute to the sequence count)	unitless	[0 - 65535]
26	Sample value (> threshold)	2	SHORT	Sample value on 16 bit. depending on the "type of data sample" of the channel tuple: 0.001 volts, or 0.01 dB, Sv and TS	For volts: [-32.767 to 32.767volts] Practical range: [0.000 to 25.000 volts]; for Sv and TS: [-327.68 to 327.67 dB] Practical range: [-150.00 to 0.00 dB]	
...	continued					
...	Sample sequence number	2	USHORT	idem		
...	Sample value (> threshold)	2	SHORT	idem		
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[36 - 4 giga]

PING TUPLE U-16-ANGLES

The ping tuple U-16-angles could be used when the type of sample data are phase angles from the split-beam analysis (see table 5. Channel tuple for the Simrad model EK-500) to encode the alongship and athwartship phase angles of the sample. Here in the ping tuple U-16-angles, the sample sequence and the two angles are encoded using an uncompressed 16-bit format. (Tuple type codes 10030 - 10039 are reserved for ping tuples using uncompressed 16-bit format for sample encoding).

Table 22. Ping tuple U-16-angles.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10031**	unitless	10031
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative values are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	Sample sequence number	2	USHORT	Sample sequence number since the beginning of the ping (samples < threshold contribute to the sequence count)	unitless	[0 - 65535]
26	Sample value, alongship electrical phase angle	2	SHORT	Alongship electrical phase angle of the sample from the split-beam analysis. Zero (0) is the main axis of the transducer beam and positive is in the fore direction See "Alongship angle sensitivity" in table 5 for electrical to mechanical angle conversion (see EK-500 user manual).	0.1 degree	[-3276.8 to 3276.7 degree] Practical range: [-180.0 to 180.0 degree]
28	Sample value, athwartship electrical phase angle	2	SHORT	Athwartship electrical phase angle of the sample from the split-beam analysis. . Zero (0) is the main axis of the transducer beam and positive is in the starboard direction .See "Athwartship angle sensitivity" in table 5 for electrical to mechanical angle conversion (see EK-500 user manual).	0.1 degree	[-3276.8 to 3276.7 degree] Practical range: [-180.0 to 180.0 degree]
...	continued					
...	Sample sequence number	2	USHORT	idem		
...	Sample value, alongship electrical phase angle	2	SHORT	idem		
...	Sample value athwartship electrical phase angle	2	SHORT	idem		
...	Optional field: Space	2	USHORT	When needed: Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[40 - 4 giga]

PING TUPLE C-16

In the ping tuple C-16, the sample value is encoded using a compressed 16-bit format. (Tuple type codes 10040 - 10049 are reserved for ping tuples using compressed 16-bit format for sample encoding).

Table 23. Ping tuple C-16.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10040**	unitless	10040
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative values are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	No. of samples (> threshold) in this ping	4	ULONG	No. of samples (> threshold) in this ping (This information can also be computed from the tuple size).	unitless	[0 - 4294967295]
28	Sample value	2	SHORT	Sample value on 16 bit or zero series (< threshold) compressed into RLE samples (the upper 8 bits are set to 1 and the lower 8 bits indicate the no. of zeros + 1; 256 below threshold values can then be compressed into one RLE sample; no value smaller than -32511 or larger than 32511 can be encoded).	depending on the "type of data sample" of the channel tuple: 0.001 volts, or 0.01 dB for Sv and TS	For volts: [-32.511 to 32.511 volts] Practical range: [0.000 to 25.000 volts]; for Sv and TS: [-325.11 to 325.11 dB] Practical range: [-150.00 to 0.00 dB]
...	continued					
...	Sample value	2	SHORT	idem		
...	Optional field: Space	2	USHORT	When needed: Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes).	byte	[40 - 4 giga]

PING TUPLE CE-16

In the ping tuple CE-16, the sample value is encoded using a compressed and encoded 16-bit format. (Tuple type codes 10050 - 10059 are reserved for ping tuples using compressed 16-bit format for sample encoding).

Table 24. Ping tuple CE-16[‡].

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[30 - 4 giga]
4	Tuple type	2	USHORT	Tuple type code: **10050**	unitless	10050
6	Time fraction	2	USHORT	Time of the transmitted pulse. Fraction of a second to add to the CPU ANSI C time (next field) to get a time precision of 0.0001 s.	0.0001 s	[0 - 6.5535 s] Practical range: [0 - 0.9999 s]
8	Time CPU ANSI C Standard time	4	ULONG	Time of the transmitted pulse. CPU ANSI C time, in seconds. Usually the CPU clock is set to local time.	s	[0 - 4294967295 s] (= up to year 2106)
12	Software channel identifier	2	USHORT	Unique identifier for this software data channel to which the ping data is associated.	unitless	[0 - 65535]
14	Transmitter mode	2	USHORT	Operating mode of the transmitter: **0** = active the transducer is transmitting monotone pulse **1** = passive the transducer is not transmitting but only receiving **2** = test: a calibration signal is injected in the sounder Other modes could be added, e.g. chirp pulse transmission.	unitless	[0 - 65535] Presently: [0, 1, 2]

‡ This format is inspired from the Biosonics DT4000 data format, but it is different because a SHORT is used instead of a USHORT and all < threshold samples are included in the RLE. For details see below.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
16	Ping number	4	ULONG	Ping sequence number since the beginning of the file. This should be a permanent label of the pings that should not be altered in further processing steps, namely the edition steps.	unitless	[0 - 4294967295]
20	Detected bottom range	4	LONG	Positive values indicate the range from the transducer face where the bottom detection criteria were encountered under the above active transmitter mode. The sound speed field of the echosounder tuple is used for conversion of time to space. A value of 2147483.647 m means that the bottom was not detected. Negative values are reserved for future use.	0.001 m	[-2147483.648 to 2147483.647 m] Practical range: [0.000 - 15000.000 m]
24	No. of samples (> threshold) in this ping	4	ULONG	No. of samples (> threshold) in this ping (this information can also be computed from the tuple size).	unitless	[0 - 4294967295]
28	Sample value	2	SHORT windowing 3(13)12	Sample value encoded with a mantissa on the lower 12 bits (with a virtual 13 bit assumed to be 1) and an exponent on the upper bits no. 13 , 14 and 15. The objective is to store the most significant 13 bits. This gives a resolution of 1/8192 (0.012%) and a range of 20 bits ($2^{13} * 2^{7}$) (=120 dB = 20 log 2^{20}). Zeros series (< threshold) are compressed into RLE samples (the upper 8 bits are set to 1 and the lower 8 bits indicate the no. of zeros + 1; 256 below threshold values can then be compressed into one RLE sample; then no value smaller than -507840 or larger than 507840) can be encoded). For details see below.	depending on the "type of data sample" of the channel tuple, resolution of 1/8192, smallest absolute coded units: 0.0001 volts, or 0.001 dB, for Sv and TS	For volts: [-50.7840 to +50.7840 volts] Practical range: [0.0000 to 25.0000 volts]; for Sv and TS: [-507.840 to 507.840 dB] Practical range: [-150.000 to 0.000 dB]
...	continued					
...	Sample value	2	SHORT	Idem.		
...	Optional field: Space	2	USHORT	Space to allow the next field to be aligned on an address that is a multiple of 4.	unitless	0

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes)	byte	[40 - 4 giga]

Analysis of sample encoding format for the above ping tuple type CE-16, (code 10050)

SHORT =

BIT NO.															
sign	*15*	*14*	*13*	*12*	*11*	*10*	*9*	*8*	*7*	*6*	*5*	*4*	*3*	*2*	*1*

if (upper 8 bits *[9 to 16]*== 0xFF)

rle_count = lower 8 bits *[1-8]*+ 1

else

exponent = upper 3 bits *[13-15]*

mantissa = lower 12 bits *[1-12]*

if (exponent > 0)

counts = (0x1000 + mantissa) << (exponent-1) (1)

else

counts = mantissa (2)

(1): an implicit 13th bit is added to the mantissa if (exponent > 0);

(2): a single zero (0) value would be coded with (exponent ==0) and (mantissa==0); also all values between 1 and 4095 are represented with exponent==0

As expressed above, to have an rle_count we need all of the upper 8 bits to be equal to 1

1111 1111 xxxx xxxx *x = 0 or 1*

This means that no value above

counts = (0x1000+ 1110 1111 1111) << (7-1)

counts = (1 1110 1111 1111) * 2^6

counts = 507840 (decimal)

could be represented.

TEMPORARY OR PRIVATE TUPLE TYPES

The temporary or private tuples could be used to encode some information that is needed for a short period of time or to run some specific software applications. The definition of the fields of these tuples is left to the users. The *HAC* tuple syntax rules and required fields must be respected. Tuple type subcodes are used to label specific tuple types, as done below for the temporary tuple 1. (Tuple type codes 65496 - 65505 are reserved for temporary or private tuples).

Table 25. Temporary tuple 1.

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
0	Tuple size	4	ULONG	Tuple data size: variable	byte	[38 - 4 giga]
4	Tuple type	2	USHORT	Temporary tuple 1 type code: **65396**	unitless	65396
6	Tuple type level 1 subcode	2	USHORT	Tuple type level 1 subcode. Users can choose a subcode to label their specific information. Since there will not be a public description of these subcodes, contrary to the present *HAC* tuple type description, there is a risk that two different users will use the same subcode to label different types of information. Other levels of subcode (e.g. next field) could then be used to prevent such collisions. The user`s identification field could also be used for this purpose.	unitless	[0 - 65535]
8	Tuple type level 2 subcode	2	USHORT	Tuple type level 2 subcode. To be defined by the users.	unitless	[0 - 65535]
10	User`s identification	30	CHAR	Character string for identifying the user or the application, up to 30 characters.	ASCII char.	30 characters
Space for data, structure to be determined by the user.						

Offset (byte)	Field	Length (bytes)	Format	Content	Encoded units	Limit range
...	Tuple attribute	4	LONG	Attribute of the tuple: **0** = original tuple, e.g. nothing special to mention **1** = edited tuple Other attributes could be labeled by a code (e.g. tuple data quality). Negative codes should be used for special cases.	unitless	[-2147483648 to +2147483647]
...	Backlink	4	ULONG	Tuple size: variable (multiple of 4 bytes)	byte	[48 - 4 giga]

INDEX