

FAU format

General comments to the format.

When data from the hydrographic survey is converted to FAU, it is vital for the further process that the structure in the observations is conserved, i.e. that there is a Survey data structure for each beam in each ping. If, during the conversion, some beam possibly cannot be assigned an observation, a substitute Survey data structure must be inserted at its place and its quality member must be set to 128.

The int and the time_t are 4bytes long each, the short is 2 bytes and char 1 byte, each byte consisting of 8 bits.

Only quality values between 1 and 15 are valid quality indicators, the other bits in this byte are reserved for additional flagging.

The FAU data format is:

```
typedef struct {
int          N;          /* Northing of depth, in units of centimeter . */
int          E;          /* Easting of depth, in units of centimeter. */
int          depth;     /* observed depth in units of centimeter. */
time_t      sec;        /* UNIX time for the ping */
short       angle;      /* observed beam angle with respect to the Nadir, in units of 1/100 degree; */
char        heave;     /* observed heave for the beam, in units of 2cm */
char        roll;       /* observed roll for the beam: in units of 1/10 degree; */
unsigned char quality;  /* quality of the depth observation */
unsigned char amplitude; /* amplitude of the observation, not used for the moment */
char        pitch;      /* observed pitch for the beam in units of 1/10 degree */
unsigned char centisec; /* fractional ping time in units of 1/100sec. to be appended to sec */
}Survey_data;
```

The only relevant quality allowed for flagging data is: "Deleted by the multibeam system".

```
#define ERRORBIT          128 /* 1000 xxxx set in quality field when obs. is deleted by the multibeam system */
```

Other quality values used by customer is:

```
#define MULTIPATH          144 /* 1001 xxxx set in quality field when a multiple path depth is identified */
#define UNVERMSP           160 /* 1010 xxxx bits for flagging observations by parameter settings for MapSpike */
#define ERRORSSP           192 /* 1100 xxxx bits for flagging observations by VISE */
```

MapSpikes and Vise are in-house software.