- AisBinaryMessageRfc/Summary Specification Format for AIS Binary Messages

It is likely that what is in svn is newer that what is here.

Specification Revision

This document, when completed, will be known as revision **1.0**. When completed this document may not be altered unless the version number is incremented.

svn rev: $Revision: 2115 $

svn date: $Date: 2006-05-15 08:37:40 -0400 (Mon, 15 May 2006) $

Status

This document is meant to present ideas about how AIS binary messages **might** be specified. Please give feed back as to parts of this specification that you feel should be changed or removed. Also what is missing? Thanks! -kurt

This memo provides information for the Maritime and Internet communities. It does not specify an Internet standard of any kind. Distribution of this memo is RESTRICTED to NOAA, USCG RDC, and UNH. This restriction is only for when the document is in progress.

Author

Kurt Schwehr / kurt@ccom.unh.edu

Abstract

The document describes a technique for using XML (Extensible Markup Language) to define the binary content (payload) for maritime based AIS (Automatic Identification System) binary messages (messages 6 and 8). These messages allow regional authorities to define messages tailored to their requirements. By providing a bit level description in XML, developers of these binary messages will be able to more clearly specify messages to software engineers implementing communication systems that must handle this traffic. Along with the XML specification, this draft document provides a sample implementation of code to generate a sample encoder/decoder of AIS binary message payloads. This system is also informally known as the "AIS Binary Decoder Ring."

The XML message definition file specifies the order, size, and type of the bit stream. However, this draft does not specify semantics or how binary messages should be presented on an ECDIS or other display device. An XML schema and an additional program will provide validation of the XML message definitions.

It is the author's hope that this specification will also work for larger sized messages sent over higher data rate links and it may be adaptable to other problem domains such as multibeam sonar format codecs.

Note

A table of acronyms is located at the end of this document.

Introduction

The current specification for the Maritime Automatic Identification System (AIS) is defined in the ITU-R.M.1371-1. The space of available messages with in t1371-1 specification is restricted to 64 messages and requires large amounts of time to get new messages approved. The specification does provide two messages (6 and 8) for regionally defined messages.

This specification defines an XML language for describing the contents of AIS binary messages. It can be thought of as a meta language from which parsers can be generated and tested.

To Do

- Flush out the description of the format with XML examples and go through the use.
- Think about how this could be slightly more general as a packed binary message description much like the XDR RFC. e.g. for multibeam data Perhaps DAC, FID, and eFID become address1, address2, ... addressN fields?
- Fix the schema so that struct-includes can be anywhere
- Make a example code generator for C++.
- Need a clear description of "bit stuffing" and any related issues.

Goals

- Human readable specification that is also machine readable
- Allow automated testing and validation of implementations based on the specification
- Provide specification of the order of fields, length of fields, and type of fields
- Specify the scaling and offsets to be applied to the field between the application and the AIS network layer
- Declare the units after removing scaling
- The specification must be independent of programming language (e.g. can be implemented in C, C++, Java, Python)
- Optionally allow a parser to directly use the specification rather than code generated based on the specification

Assumptions

- Bandwidth over the AIS VHF system is limited. Messages must be packed in a minimum of space.
- The data rates of AIS messages are low (is this a really true?)
- Representation and semantics will be specified by a higher level specification and is outside the scope of this document.
- Computer navigation systems are powerful enough to where we do not have to worry about the specification file size
- The specification will not need to be sent over AIS. Vessels and shore stations can periodically update their software via higher bandwidth means.

What this draft does NOT mean

The goal of this specification is not to lock ECN, ECDIS, or other vendors into using the XML binary message specification or some code derived directly from the XML. Implementations may use what ever means necessary to parse and encode these messages.

Future Work

It would be beneficial to create a system that converts the XML specification file into a more natural

document. A main candidate is to use XSLT to create HTML and PDF documents to assist with using AIS data streams.

- Write a DTD
- Write other schemas such as RelaxNG and Schematron
- Create code generators for Java, Ruby, C#, etc

Prior work

There have been many such systems designed similar what is proposed here. I have tried to draw on the best of the many systems. The most relevant specification is "RFC 1832 - XDR: External Data Representation". The AIS binary message XML specification is a simplification and XMLization of RFC-1832 with additions that fit the specific requirements of AIS.

http://www.faqs.org/rfcs/rfc1832.html

AIS Version 2

Working groups are currently busy with this specification, but I have not seen any results.

Coast Guard PAWSS

I do not know much about this.

Base Station Standard

FIX: It is in trac. Kurt needs to read this.

St. Lawrence Seaway

The St. Lawrence Seaway is actively using the AIS binary messages. These messages are defined in:

```
St. Lawrence Seaway AIS data Messaging
      Formats and Specifications

Harmonized with U.S. Coast Guard PAWSS
             AIS Messages

            Revision 4.0A
             May 8, 2002
```

These messages are defined in a way in which this specification can easily assimilate the existing messages.

Proposed System

I propose an XML based message definition format. This system is able to meet all of the goals described above and relies on industry standard technologies. There are many libraries that support reading and validation of XML documents. Additionally, the specification will come with an XML Schema (xsd) that defines what is allowed in the AIS Binary Message XML specification file. Designers can validate new message definitions using the schema.

Additionally an XML delivery system will be included that allows for the packaging of test data sets. Hopefully this XML system will just be the AIS XML specification that is currently being worked on by the US AIS XML committee (FIX: ref who this is and the official name). If that specification turns out to be unsuitable, a simple replacement can be signed. For each message, a number of example messages will be defined such that the major corner cases can be tested by all AIS software vendors. Each example XML message will contain:

- An ASCII encoded binary field containing the bit stream as represented by 0's and 1's.
- The NMEA strings as they would be returned by an AIS modem.
- The fields broken out with scaling removed

Message Specification Format

The XML specification for one message is encapsulated in a "message" tag. The message contains the necessary information to serialize and deserialize AIS message information to/from the AIS binary message payload and the location machine representation used within an application.

Each specified message must evening fit into 1 to 5 slots (See the ITU-R M 1371-1 specification for the definition of an AIS VHF slot).

FIX: Need to work up a clear test for what fits within the message boundaries.

Padding

If a variable length message does not completely fill up a slot, the message should be padded with (FIX: with what? Does all 0's work? All ones will mess up AIS message synchronization).

Byte order

Byte order should be always be the same as that used in AIS. FIX: make sure I call this out correctly. All numeric values are **BIG ENDIAN**.

FIX: give some examples of what this means with both signed and unsigned integers

Basic layout

The super structure of ais-binary-msgs.xml will look like this:

```
<?xml version="1.0" encoding="utf-8"?>
<ais-binary-message version="1.0">

</ais-binary-message>
```

Name spaces

Each message name must be unique. The fields within a structure or message must be unique within that structure or message. All names must be less than 64 characters.

One message

Each message is wrapped in a "message" tag. The message tag contains attributes that specify the name, ais message number, the DAC (Designated Area Code), the FID (Functional ID), and the eFID (extended Functional ID). The attributes for DAC, FID, and eFID will be repeated inside of the message as fields to facilitate both parsing of messages and XPath XML searches for the message definition.

The aismsgnum can be "6", "8", or "6 8" depending on where the binary message may occur in AIS messages. There may be more message numbers in the future as the AIS specification is revised.

How to know which message (DAC, FID, eFID)

In order to know what message is being examined, the first bits of the message must be decoded. These fields will be included in the XML specification for all messages and marked as "**uint**".

The ITU 1371 specification of messages 6 and 8 states that the message starts off with 16 bits of the

application id:

DAC	FID
10 bits	6 bits

The DAC is the designated area code, which is 366 for the US. The function id (FID) says which application this data is for. This provides only 64 messages for the US which is not likely to be enough.

Extended FID

I propose that for a message with a US DAC (366) and an FID of 63 be of a class known as extended Functional Identifiers (eFID). This subset of messages will provide 12 bits for an extended message identifiers. This will give the regional authorities 4096 message types that can be used for a large number of purposes. Having only 64 possible message types is a serious limitation for the community that needs a larger range of messages.

NOTE: It is an open question as to how many messages are required for the life of the AIS system. By providing a very large number of messages, the system can accommodate many different types of applications without concern for running out of message identifiers, but what is the right number?

Basic Types

The basic types provide the foundation for an AIS binary message definition. Here is the list of basic types:

name	**description**	**size**
bool	boolean	1 bit
uint	unsigned integer	variable
sint	signed integer	variable
udecimal	unsigned decimal	variable
sdecimal	signed decimal	variable
float	ieee float point	32 bits
double	ieee float point	64 bits
aisstr6	As defined in the AIS specification	6 bits
ascii7	ASCII character codes	7 bits

The above values may come in arrays, so strings are created with Arrays as defined below.

Decimals are fixed point values stored as an integer plus a scalar divisor. FIX: explain better

Fields

A Fields is basic unit. Each field is required to have name, number of bits, and type attributes. The name must be of the form "[a-zA-Z0-9_]*" meaning only letters, numbers, and the underscore ('_'). The type must be from the Basic Types table above. The minimum start of a tag looks like this:

```
<field name="somefieldname4" numberofbits="3" type="unit">
```

Each field is also required to have a human readable description tag.

```
<field name="onland" numberofbits="1" type="bool">
  <description>Set this flag true if the station or vessel is on or over land</descripti
```

Required Value

If a field must have a specific AIS binary encoded value, specify it here.

Scaling

Units

> All fields must have a units field. This is a string that follows the (FIX: WHICH) standard for textual representation of units. Or do we have to make our own table units? Please no!

FIX: What units should many values be in? Is a "count" a unit? Or should there be a "None" unit?

FIX: Can we use SI? http://en.wikipedia.org/wiki/SI Probably need to override quite a bit.

http://hpcrd.lbl.gov/staff/olken/metrication.htm Talk about the Mars Climate Orbiter and unit specification.

Unavailable value

The unavailable tag specifies the value that is to be used when the data is either out of range, unknown, or otherwise unavailable.

```
<field name="waterdepth" numberofbits="">
```

Structures (grouping)

Structures provide predefined groups of fields that may be repeated.

A proposed idea is to flatten the structures within a field. This would prepend the name of each structure into the name of the fields with a structure something like this in pseudo code:

```
structure pos2d
  field lat
  field lon
message foo
  include name=origin struct=pos2d
```

This would then pass through a flattener program and produce:

```
message foo
  field pos2d.lat
  field pos2d.lon
```

I believe this could also be done with XPointers. Have to try it out. Simpler is probably better.

Arrays

Arrays allow repeated instances of basic types or structures.

Fixed Length

When the number of objects will always be the same, a fixed length array can be used.

Variable Length

When the number of objects can change a variable that precedes the array can be used to specify the

number of objects.

Compression

Considering for text fields to allow them to be compressed with a standard (open and free) compression library. It would then be variable payload in that field depending on how well the bit stream compresses.

Lookup tables

Examples

St Lawrence Seaway

Marine Mammal Location

Open Issues

- Should there be an eFID and can it use FID of 63?
- Should define a basic geophysical report for MGG research ships with field like:
  - magnetic field declination, inclination, and intensity
  - gravity
  - depth
  - water temp, depth of measurement
  - xbt reports
  - etc

Acronyms and Terms - Glossary

FIX: flushout glossary

- AIS - Automatic Identification System for maritime messaging
- ASCII - American Standard Code for Information Interchange. Specified in ANSI X3.4-1986. http://www.ietf.org/rfc/rfc1345.txt
- DAC - Designated Area Code. Says what country the message is for
- ECDIS -
- ECN -
- eFID - for FID of 63, provide an "extended" FID with many more possible messages
- FID - 64 message types for a particular country (0..63)
- RFC - Internet Request for Comment http://www.faqs.org/rfcs
- VHF -
- XDR - eXternal Data Representation. Used for Unix RPC and NFS communication. http://www.faqs.org/rfcs/rfc1832.html
- XML - Extensible Markup Language http://en.wikipedia.org/wiki/Xml
- XSD - "XML Schema" Schema specifying the allowed format for an XML document

Attachments

- ais-binary-msgs.xml (9.6 kB) -"Sample XML definition file to illustrate what the specification files would look like", added by schwehr on 06/01/06 09:41:51.
- ais-binary-msgs.xsd (6.9 kB) -"Close to a working schema to validate the XML specifications", added by schwehr on 06/01/06 09:42:28.
- IEEE_754_FPArithmetic.pdf (84.1 kB) -"IEEE Float point specification", added by schwehr on 06/16/06 08:25:24.
- brc.comments.txt (4.8 kB) -"Comments from Brian Calder on the specification", added by schwehr on 06/16/06 08:26:49.
- MCO_report.pdf (346.3 kB) -"Mars Climate Orbiter failed because of a units problem", added by

schwehr on 07/06/06 12:21:25.
- XML AIS Spec - brief explain-v2.doc (21.5 kB) -"Short summary document from 06-Jul-2006", added by schwehr on 07/06/06 12:23:54.