Joint TSMAD18 & DIPWG1 MEETING

4th to 8th May 2009 (Ottawa, Canada)

Paper for Consideration by TSMAD
S-101 General Discussions

Submitted by:	S-101 Work Item Leader
Executive Summary:	This Paper is to set the stage for general overarching discussions relevant to S-101
Related Documents:	TSMAD18-16.3A_S-101Prod Spec
Related Projects:	None

Introduction / Background

In order to focus the discussion at the joint TSMAD/DIPWG, this paper will draw attention to specific areas of S-101 that need in-depth discussion to establish a way forward in the development of S-101.

Analysis/Discussion

The following table represents key areas of discussion for the upcoming TSMAD/DIPWG meeting. The first column notates either the specific section in the draft version of S-101 or if it is a general overarching discussion. The second column represents the action or section heading of S-101 and the third column represents the initial thoughts of the S-101 work item leaders and the TSMAD chair.

In addition their will be several other papers that go into more detail regarding certain aspects of S-101. These were created from the wiki discussion on S-101.

S-101	Action	
General	**Tying FCD new entries to FC**	Need to develop a strategy for tracking entries into the Feature Concept Dictionary and the whole package goes into the Feature Catalogue goes into the next version. This also needs to coordinate with the Portrayal Control Body. (Excel Spreadsheet/word)
General	**ENC 2.0**	ENC 2.0 – need to develop a strategic plan to differentiate between S-57 and S-101 (topic for Ottawa discussion)
General	**M_NSYS**	Need to rewrite rules to have multiple M_NSYS with different ORIENTS and need to figure out where it goes in S-101, the Feature Catalogue and encoding guide
General	**Feature Catalogue and Portrayal Catalgue updates**	Need to develop a mechanism for the delivery of new updates to the feature and portrayal catalogues. It would need separate encoding of XML. We need to make sure that the end-users will get either via the IHO website, OEM's or distributors.
2	**scope**	Tentatively think there are three scopes. • Root Scope would be common to everything. • Scope 1 – could be for scale dependent data • Scope 2 – could be for scale independent data. Question is this is not cross-referenced anywhere else in the document to call which scope or scopes are to be used. Is this in the encoding and does it go in the encoding as metadata and if scope is common where do you state what is

		common – ie CRS.
4.1	**application schema**	This provides the UML use cases for S-101, such as complex attributes, relations between features and spatial, information types, scale dependent and independent data and how the ENC is put together. If you try and put everything in the FC into here then the document maintenance would be nightmarish. Needs to just lay out the interconnections between everything.
4.3.3	**Themes**	Divide Groups into Themes based on INT 1. This would eliminate group 1 versus group 2, make encoding easier and define the actual themes in the Feature Catalogue. Possibly have an Annex listing the different themes and the features associated with that theme. Need to have a statement distinguishing between a theme and an aggregation type.
4.4	**Time Varying Features**	Needs sorting – is MAGVAR still needed. Tidal Working Group is to report to TSMAD
4.7.5	**Mandatory Attributes**	Need to revisit mandatory attributes. Some may not be necessary anymore. This goes in conjunction with the review of the Portrayal Catalogue and Feature Catalogue. Also, how to deal with unknown mandatory attributes and how S-58 will check for this.
4.15	**Display Scales**	.3mm rule. Would like to make it mandatory. Need to ask if all production systems can properly filter down.
4.16	**Geometry**	Should we fully describe each geometry type within S-101 or refer back to S-100
5.4	**Vertical and Sounding Datum**	Need to ensure that the changes in vertical and sounding datum need to be displayed to the mariner – Portrayal section needs to address it.
6	**Data Quality**	Most of this section goes to DQWG – need to have a discussion on up-to-datedness of the data. Does it belong in this section or in metadata.
9	**Portrayal**	Need to incorporate major portions of S-52 into S-101 to give better instructions on how to control portrayal. DIPWG would still control this section. S-52 still remains a live standard as long as S-57 is alive and well.
11.1.1	**Dataset Naming**	Need a discussion on dataset and support file naming.
11.1.7	**Updating**	Need a big discussion on how to tell the mariner that the update was done and what features have changed.

The following three discussion topics are contained in separate papers and will be discussed individually.

4.7.4	**Text Placement**	Need to have a discussion on text placement. Do we create a complex attribute of distance and orientation and let the OEMs and ENC software producers implement this on a system level. See associated paper.
4.13.2	**Scale Independent features**	Made a first cut in Capetown. Need to review the list again, taking into account scope and the fact that areas have scale. We may want to start this concept on a smaller scale and then expand as we test it out. Maybe use SCAMIN and SCAMAX and have one cell. See associated paper.
4.15	**Display scales**	Display and Chart Loading. See Associated Paper

Wiki Discussions :

Data Sets :

Four kinds of data sets may be produced :

- new data set : no ENC data has previously been produced for this area and for the same navigational purpose.
- update : changing some information in an existing data set.
- re-issue of a data set : including all the updates applied to the original data set up to the date of the re-issue. A re-issue does not contain any new information additional to that previously issued by updates.
- new edition of a data set : including new information which has not been previously distributed by updates.

DG - Re-issue should be removed as it is a part of the paper chart world.

EM: Don't necessary agree with DG's comment, HOs current release re-issues of data sets. Therefore we need to consider the consequences of removing this option. How will it impact distribution of data?

Each new data set or new edition is called a base cell file.

A data set containing updates to one base cell file is called an update cell file.

The issue date (the ISDT sub-field of the DSID field) is the date from which the data can be used.

The issue date (the ISDT sub-field of the DSID field) is the date on which the data has been released by the HO and therefore can be used.

The update date (the UADT sub-field of the DSID field) is the date on which an update has been applied to a dataset by the HO.

EM: suggest to add by the HO, to remove possibilities of confusion regarding the use of UADT.

CDOB comment on 15.3 Data Sets

Updating can either be done by file replacement or by incremental transactional updates. To do transactional updates the data needs to be specially structured to accommodate such updates. The ENC vector data is so structured. This goes right back to early decisions in ENC V.2. In particular the ENC data is structured at a low level of topology so that incremental changes do not cascade resulting in many derivative changes from one simple change. Incremental update should be restricted to only the base feature oriented vector ENC data and not any auxiliary files. There is only one important use case for incremental updates and this is satisfied by limiting incremental updates to this level. All other data should be updated by file replacement.

Because there are effectively two types of updates there is a need to retain "re-issue" of a data set. Re-issue is effectively file replacement update. This means we have:

- new data set : no ENC data has previously been produced for this area and for the same navigational purpose.
- update : changing some information in an existing data set. (Transactional update)
- re-issue of a data set : including all the updates applied to the original data set up to the date of the re-issue. A re-issue does not contain any new information additional to that previously issued by updates. (file replacement update)
- new edition of a data set : including new information which has not been previously distributed by updates.

Cells :

In order to facilitate the efficient processing of ENC data the geographic coverage of a given usage must be split into cells. Each cell of data must be contained in a physically separate, uniquely identified file on the transfer medium, known as a data set file (see clauses 5.4 and 5.6.3).

Cells must be rectangular (i.e. defined by 2 meridians and 2 parallels). No advantage to removing rectangular cells. Any data limits possible.

EM: Because of the difficulties maintaining boundaries between cells restricting cell shapes to rectangles makes sense.

EM: Should something be stated about polar regions and the 180 degree longitude?

The coordinates of the borders of the cell are encoded in decimal degrees in the 'Catalogue Directory' [CATD] field. Data extent will be described using MD_DatasetDiscoveryMetadata.boundingBox

Each cell must contain a minimum of one group 1 (skin of the earth) feature. Data coverage in the cell is defined by the exterior boundary/ies of one or more contiguous group 1 features. Areas of non data coverage are defined as those areas within a cell not populated with data. Remove- we should not define no data areas catcov = 2 has been removed from the standard.

Data in cells with the same display scale should not overlap. In areas of overlap the mariner should have the choice.

EM: Terminology must be defined (data limits vs. data coverage vs. cell limits vs. borders)

Cell file size.

Current 5MB limit is too small. Ex: hi-density ENCs, hi-res bathy

ENC maximum file size will be increased to 15Mb
Should there be a maximum file size for an update? 5MB?
EM: is there an exchange set size limit too? This must be discussed, since it is still unclear, at least to me, how gridded data will be distributed. As part of an S-101 exchange set, or as individual products? What about image files of for example a light house? We had a discussion on this internally and recommend that this chapter should not impose a limit on file size. Instead a statement that ECDIS systems should process files efficiently should be inserted. Future increasing of file size should be considered. For example, the ECDIS Stakeholder Forum could decide what the file size should be for the next two years. In addition the file size should be based on a detailed analyses and be on an analytic value.

In order to facilitate the efficient processing of ENC data the geographic coverage of a given usage must be split into cells. Each cell of data must be contained in a physically separate, uniquely identified file on the transfer medium, known as a data set file (see clauses 5.4 and 5.6.3).

Cells must be rectangular (i.e. defined by 2 meridians and 2 parallels).

The coordinates of the borders of the cell are encoded in decimal degrees in the 'Catalogue Directory' [CATD] field.

Each cell must contain a minimum of one group 1 (skin of the earth) feature. Data coverage in the cell is defined by the exterior boundary/ies of one or more contiguous group 1 features. Areas of non data coverage are defined as those areas within a cell not populated with data.

Data in cells with the same display scale must not overlap.

BG:RENCs have adopted an unofficial policy of allowing a 5 metre "on the ground" overlap to allow for difference in coordinitate multiplication factors. I guess if we mandate the 10000000 this won't be an issue.

RF: The 5 metre on the ground overlap originates from item 11 in the IHO *Recommendations for Consistent ENC Encoding. E*ven with a mandated COMF value of 10000000 there will always problems achieving a perfect join where the data limits do not follow a meridian or parallel.

Each cell must be completely covered by meta objects of class **M_COVR**. The areas that contains data must be indicated by **M_COVR** objects with CATCOV = 1. Any other areas that do not contain-data must be covered by **M_COVR** objects with CATCOV = 2.

HA: We have had discussions in the past regarding whether M_COVR CATCOV=2 objects are really needed. Has a conclusion been reached if so can we document the justification for having them.

Cells with the same navigational purpose may overlap. However, data within the cells must not overlap. Therefore, in the area of overlap only one cell may contain data, all other cells must have a meta object M_COVR with CATCOV = 2 covering the overlap area. This rule applies even if several producers are involved.

HA: We have had discussions about changing the overlap rule to same scale band or radar range rather than purpose since purpose and scale do not seem consistent. Do we need to restrict overlap between producers? No overlap from the same producer makes sense but overlap between producers probably happens anyway so we might as well accommodate it properly. Perhaps if cells from different producers have overlapping coverage at the same scale the ECDIS user should be given the choice of which one to put on top. The ECDIS makers and users should be consulted about this I think. Currently some agencies are using complex territorial boundary shapes to crop their data making complicated and dense M_COVR geometries that perhaps are not really needed.

Point or line features objects which are at the boundary border of the data coverage (**M_COVR** with CATCOV = 1) of two cells with the same navigational purpose must be part of only one cell. They are put in the south or west cell (i.e. north and east borders of the cell are part of the cell, south and west borders are not).

When a feature object exists in several cells its geometry must be split at the cell boundaries and its complete attribute description must be repeated in each cell.

CDOB comment on 15.1.1 Cells

In a theoretical sense, and this is a question for S-100, there should exist both a tiling scheme and separately a reference into the tiling scheme. That is, the tiling scheme needs to be identified, and each tile needs to identify itself as a member of the tiling scheme. This is shown in the Imagery and Gridded Data part in the UML, but it is a general S-100 issue.

In S-101 it is therefore necessary to both describe the tiling scheme, and to index into it.

It is not sufficient to say simply that there are rectangular cells. "Cells must be rectangular (i.e. defined by 2 meridians and 2 parallels)". If one covers the whole world one also needs to cover the polar regions.

A tiling scheme is itself a discrete coverage (as described in the IGD section), and it is necessary to describe the boundary of the coverage and how it is tessellated. It can be a simple grid over most of the earth, but the cells change the amount covered as they approach the pole. There have been many plans to address this in other communities. For example DGIWG has the ARC grid structure which is a series of banded cells with polar stereographic over the poles. How will IHO address this? We need something more than what we have.

The tiling scheme used should be explicitly described in the S-101 product specification, probably in an annex.

Separately, each tile/cell needs to identify itself within the tiling scheme. I believe this should be done in an attribute, and not just in the filename. Filenames can be corrupted for many reasons and having a name within the file should be the primary identifier. It also allows one to fix corrupted filenames. The filename is external to the file. That is one knows the filename without opening the file and therefore it can be equivalent to, or be related to the tile name. Unfortunately the 8.3 file naming, and the associated automatic filename corruption process developed by several operating system vendors affects this. See comment on 15.4.

If one a plans to extend the tiling scheme to hi density ENCs and to re-use the same tiling scheme for auxiliary data such as bathy overlay, currents etc, then the tiling scheme identification becomes even more important and needs to be even more sophisticated. There needs to be different series at different densities.

The concept of having an optional overlap area of 5 meters (or some other number) in each cell is reasonable, but it really just represents a more sophisticated tiling scheme. If there is overlap there needs to be a way to indicate what is the data that a cell indicates that it has the authority to describe and what it just caries along as "overlap" area. Is this the proper use of M-COVR ? One should always give precedence to the authoritative cell for an area, and never the overlap data.

If we are going to mix densities and share auxiliary files then it really becomes necessary to carefully describe the tiling scheme

Text and Picture Files :

If the file is referenced from a national attribute field [NATF] encoded using lexical level 2, the file name in that attribute field must also be in lexical level 2 (two byte encoding), using only the ISO 646 (the first seven bits) subset of ISO 10646. It is up to the decoder to convert the filename to the lexical level appropriate for the receiving file system. Text and picture files will be encoded using UTF-8.

JP: notes from meeting discussions stated to get rid of NATF mentions as the FC will handle it and specify UTF-8

HB: The lexical level describes the character repertoir not how it is encoded, so neither the name of a field nor any encoding details (how many bytes) should be mentioned here.

HA: Why limit to ISO 646 why not allow ISO 10646 or just say UTF-8 and get rid of the lexical level confusion.

The text and picture files must be named according to the specifications given below :

Can this go beyond eight characters?

EM:<== I don't see why not, new systems, new rules. It would also make it easier for HOs with many ENCs to name files in a way that makes them legible and make it easier to provide unique names.

HB: More than eight characters yes, but there must be some rules (allowed characters, do not store the actual information in the file name, ...)

- the first two characters identify the producer.

The extension is used to identify the type of the file.

The following are approved extensions for text and picture files: PNG, XML, HTML, TXT

EM: HTML and XML need to be accompanied with instructions for display (standard instruction set such as XSLT). It should be clarified whether or not HTML will be allowed to carry picture files. And conventions of locating pictures as content should be developed. Exchange sets should contain all information needed to render external (to the ENC dataset) files.

Need to provide specific rules regarding how OEM's will store text and picture files and how those files will be updated with new information.

EM: <== Why? It should be up to the OEM how the files are stored in the system, all TSMAD should state is that no files should ever be overwritten by files from other data sets. Text and picture files should only be overwritten or removed by update action to the dataset that the files originally belong to. Please keep in mind that the very first recommendation from the ECDIS Stakeholders' Workshop was; "S-101 – should

describe the product produced by HO – not the product delivered by Distributor**", so TSMAD should be careful specifying items that might cross that line.**

HA: Rules should be regarding transmittal and how files get updated or deleted and not how the files are stored in the system implementation.

Text and Picture files shall have a world-wide unique name.

Textual information

The attributes INFORM, NINFOM, and TXTDSC and NTXTDS should not be used when it is possible to encode the information by means of any other attribute.

INFORM and NINFOM contain information as text, whereas TXTDSC and NTXTDS encode the name of an external file.

The text contained in INFORM and NINFOM is ASCII text. Formatting characters are prohibited. INFORM and NINFOM should generally be used for short notes (e.g. caution notes from paper charts), or to transfer information which cannot be encoded by other attributes, or to give more detailed information about an object.

EM: NINFOM and NTXTDS should be removed because they are no longer valid. And ASCII text and formating characters should be discussed. At least to ensure there is agreement for UTF-8. Furthermore, some of this text belong in the encoding guide, such as instruction on when to use INFORM or TXTDSC.

EM: **<== MD8, 8.CI.1 states that there is a limit in INFORM (assume this also applines to NINFOM) of a 300 characters. Should this be maintained?**

HA: This question should go to ECDIS systems and perhaps users because other than the limit of a field size in ISO8211 I see no reason to restrict this.

B.G: My only concern is that some producers use this field to include what are in fact notes. I agree with Hugh that users should be consulted, particularly if they would prefer to read a long text string (scrolling) or a properly structured text file.

The text files referenced by TXTDSC and NTXTDS must be ASCII files, and may contain formatted text. -These files should generally be used for longer texts (e.g. longer chart notes, tables or paragraphs from sailing directions). It is up to the producing authority to determine the most suitable means of encoding a particular piece of text.

HB: **Don't use the term ASCII here !!!!. ASCII is a character repertoir and only a few languages on this planet can be expressed with that characters. Both attributes and text files contains FREE TEXT according to the specified lexical level.**

HA: Again why not use UTF-8

HA: Also if we want to support formated text, tables etc we should really decide on what can be supported such as RTF, HTML or XML

EEM: HTML 4.01 on the grounds of wide acceptability, but XHTML (XML) could be considered, as it's technically superior, and S-101 is scheduled for release in 2012, acceptability will likely increase for XHTML (XML) by then. See http://www.w3.org/TR/xhtml1/#diffs for difference between HTML 4 and XHTML. <== Have discussed this with one of our HTML/XHMTL experts, which has informed me that there are several dialects within both HTML 4.01 and XHTML 1.0, so their recommendation was to specify that it must be HTML 4.01 strict and XHTML 1.0 strict.

JP: Decision from TSMAD 14 is that we needed to define the character repertoire and the encoding of the character must be in the meta-data.

DG: Agreed. HTML and X

RF: Do we need to introduce versioning information for both text and picture files at the moment updated/replacement file can only be distinguished by the creation date

EM: Agreed, instructions on how text, picture files and any other external file is updated is necessary. Should be done as soon as S-100 is released.

CDOB comment on 15.4.4 Text and picture files

If a file is referenced from the national attribute field (NATF) is should be encoded in UTF-8 (as should all text), but it should be restricted to match the file naming conventions. (see comment on File naming conventions.) These may be limited to Microsoft 8.3 FAT file limitations or possibly they may be relaxed a little to be a somewhat longer string, but the string should still be uppercase A-Z, 0-9 ASCII with certain excluded special characters. Anything else will not necessarily work on all implementations across platforms.

If there is a desire to have longer more descriptive identifiers in the NATF attribute field then the reference could be indirect. The actual externally referenced file names could be held in a table and the NATF attributes could reference into the table with a more descriptive name.

There should be maximum freedom to the manufacturer to produce the most cost effective implementation. The standard should only define the information content and how it is encoded at the exchange fro the supplier to the consumer. Who the consumer is, is part of the discussion. If the HO is selling directly to end users, then the HO must be concerned with the end user's system as supplied by the manufacturer. One can not assume all data will pass through the manufacturer. The interface point must be from the HO to the end user to allow maximum freedom. This does not say how the OEM will store text and pictures but it will say how the files look across the exchange interface.

Textual information

The text in INFORM and NINFORM should be UTF-8. This implies that there be a clean 8-bit implementation. That is, the UTF-8 depends upon the high order bit for code extension.

When one says "Formatting characters are prohibited, what does this mean? are Carriage Return (NL) characters prohibited. This seems too restrictive. The specification should say that this is unformatted text and that the only formatting characters permitted are Carriage Return and SPACE.

If there is a desire for properly structured text then there should be a simple XML or HTML syntax permitted. This MUST be restricted or we would end up with imbedded style sheets and all sorts of fancy, but incompatible, stuff. Yes formats like XHTML is gaining popularity, but HTML is a constantly moving target. Keep it simple. Some simple markup of bold, underline, and some layout may be desirable. We don't want animated GIFs on charts. Determine the requirement and then just provide that. Don't open the door to abuse.

The text in TXTDSC and NTXTDS should be UTF-8.

Updating :

In order to ensure that updates are incorporated into the SENC in the correct sequence without any omission, the file extension and a number of subfields in the 'Data Set Identification' [DSID] field are used in the following way :

file extension every new data set, re-issue or new edition must have a ‘000’ extension. For update cell files the extension is the number of the update, ranging from ‘001’ to ‘999’. These numbers must be used sequentially, without omission. Number ‘001’ is the first update after a new data set or a new edition, but not after a re-issue. The update sequence is not interrupted by a re-issue. After a re-issue, subsequent updates may be incorporated into the SENC created from this re-issue or to the SENC created from the original data and kept continuously updated.

edition number when a data set is initially created, the edition number 1 is assigned to it. The edition number is increased by 1 at each new edition. Where a cell is cancelled and its name is re-used at a later date, the edition number of the new cell must be one higher than the final edition number of the cell that it has replaced. Edition number remains the same for a re-issue.

update number update number 0 is assigned to a new data set. The first update cell file associated with this new data set must have update number 1. The update number must be increased by one for each consecutive update, until a new edition is released. The new edition must have update number 0. A re-issue of a data set must have the update number of the last update applied to the data set. In the case of an update cell file the file extension is the same as the update number.

update application this date is only used for the base cell files (i.e. new data sets, re-issue and new edition), not update cell files. All updates dated on or before this date must have been applied by the producer.

issue date date on which the data can be used.

EM: We should include more explanations on how UADT and ISDT correlate and which drive what.

The table below gives examples of the way to manage the file extension, the 'Edition Number' [EDTN], the 'Update Number' [UPDN], the 'Update Application Date' [UADT] and the 'Issue Date' [ISDT] subfields.

Event	File extension	EDTN	UPDN	UADT	ISDT
New data set	.000	1	0	19950104	19950104
Update 1	.001	1	1	prohibited	19950121
Update 2	.002	1	2	prohibited	19950225
...					
Update 31	.031	1	31	prohibited	19950905
Re-issue of a data set	.000	1	31	19950905	19950910
Update 32	.032	1	32	prohibited	19951023
...					
Update 45	.045	1	45	prohibited	19951112

New edition	.000	2	0	19951201	19951201
Update 1 to edition 2	.001	2	1	prohibited	19960429
...					

EM: Table should contain the termination/cancellation of edition 1.

This example table relates to the specifications given in S-52 App 1, *'Guidance on Updating the Electronic Navigational Chart'*, in the following way:

- The update information encoded in each individual cell file is called a sequential update.
- The collection of the update information encoded in the update cell files which have been issued since the last new data set, the last re-issue of a data set or since the last update was applied to the SENC is called a cumulative update. In the example, the cumulative update for the new data set starts with update number 1. The cumulative update for the re-issue of a data set starts with update number 32. The cumulative update for a data set to which update number n has been applied starts with update number n+1.
- The update information which has been incorporated in a re-issue of a data set is called a compilation update.

Each re-issue or new edition of a data set must have the same name as the base cell file which it replaces.

In order to delete a data set, an update cell file is created, containing only the Data Set General Information record with the Data Set Identifier [DSID] field. The Edition Number [EDTN] subfield must be set to 0. The [DSSI] field must not be included in this update message. This message is only used to cancel a base cell file.

EM: Harmonize use of delete and cancel. Furthermore, there is no mention of what happens to the updates and any external files.

To inform the mariner that a new edition is available, an update cell file may be created, containing only the Data Set General Information record with the "Data Set Identifier" [DSID] field. The "Data Set Structure Information" [DSSI] field is not required." The "Edition Number" [EDTN] subfield must contain a value one higher than the current edition number.

In order to modify a text, picture or application file, a new file with the same name is created.

EM: this should be harmonized with 15.4.4

When an object pointing to a text, picture or application file is deleted or updated so that it no longer references the file, the ECDIS software should check to see whether any other object reference the same file, before that file is deleted.

EM: are we sure we want the ECDIS to manage this? I think it should be done on the producer side of things. If we set up strict rules on how text, picture and application files are deleted and updated, this should not be a problem for the producer. I assume application files are stylesheets and the like?

An exchange set may contain base cell files and update cell files for the same cells. Under these circumstances the update cell files must follow on in the correct sequential order from the last update applied to the base cell file.

EM: Paragraph above is not very clear.

The record version of each feature or vector record is indicated in the Record Version [RVER] subfield of the Feature Record Identifier [FRID] field or the Vector Record Identifier [VRID] field. At each update (ER) of a record, this version number is incremented by 1. For all new cells and new editions, the RVER subfield of each feature or vector record must be reset to 1.

Up-to-dateness information

Up-to-dateness information must be given in the cell file name extension and in the "Edition Number" [EDTN], "Update Number" [UPDN], "Update Application Date" [UADT] and "Issue Date" [ISDT] subfields of the "Data Set Identification" [DSID] field.

EM: Paragraph above is not very clear.

Issuing updates in advance

Under certain conditions, it may be necessary for a data producer to issue update information in advance. For example, a change in a traffic routeing system must be made public before the new situation is implemented. Within an ER data set there is no means of indicating the date at which an update must be applied. Therefore, when an ER data set is received by an end user, it must be applied immediately. To avoid situations where update information would cause target data to reflect a situation that does not yet exist, the following coding rules must be followed:

a) If the advance update information contained in the update message involves the addition of objects to the existing data (e.g. a new lighthouse), the attribute DATSTA on the new objects must contain the date at which the update becomes active.

b) If the advance update information contained in the update message involves the modification of existing objects (e.g. a change in a traffic routeing system), it must be treated as a deletion of the existing objects and replacement with new objects. See a) above and c) below.

c) If the advance update information contained in the update message involves the deletion of existing objects (e.g. the removal of a buoy), the update message must set the DATEND attribute of all objects to be deleted to the date at which the update becomes active. **NB**. This update message does not actually delete the objects from the data set, it simply indicates that on the date held in DATEND they become obsolete. A further update to actually delete the obsolete objects from the data set should be sent at the time that the change in the real world occurs.

d) To highlight to the mariner that the advance update information contained in an update message will take place in the future, it is recommended that a caution area object (**CTNARE**) be created covering the location at which the future changes will take place. A warning note specifying, in plain language, the nature of the future change should be encoded, either in the attribute INFORM or in a file referenced by the TXTDSC attribute of the **CTNARE**. The attribute DATEND on the **CTNARE** must be set to the date at which the change described in the update becomes active.

Changes to DATSTA and DATEND cannot be applied to spatial objects. Therefore, a change to the geometry of a real world object (e.g. the relocation of a buoy) to be applied in the future can only be achieved by updating all of the feature and spatial objects involved.

As a consequence of issuing advance information updates, more than one instance of a particular real world object could exist in the data set.

DG - Issuing updates in advance – does this belong in this section or should it reside in a use of the object catalogue type document. Remove from this section and place in encoding guide section.

EM- Fully agree with DG, also think that instructions for temporary updates should be included with this.

CDOB comment on 15.5 Updating

The more complex the file system, and the larger number of linked external files, then the more complex an update process there is.

Eivind Mong raises a valid question about what happens to associated files when a cell is updated. This can be a fragile pyramid of linked files, especially if we start to have multi-levels of resolution all linked to the same set of associated auxiliary files.

Possibly update can be restricted to only specific ENC vector data files and all other changes must be by file replacement. This would make things much easier to manage. One needs to look at the end use requirements. Available telecommunications facilities have changed significantly since S-57 was conceived. There is a use case for incremental update over narrow bandwidth communications channels for particular elements of the base ENC vector data while a vessel is at sea with poor communications channels available. For all other types of data it is better to do file replacements. The updating mechanism should recognize this.

Essentially we should have the old incremental update for select changes to the base vector ENC data, and file update for everything else.

I agree that the rules for issuing updated in advance should be in the use of the object catalogue, but the updating section should indicate that there is a validity interval defined by DATSTA and DATEND and that an incremental update may supersede the interval. That is, if an object is scheduled to change status, but an update has changed the object, the scheduled event must not occur. Updates take precedence over scheduled events.

Dataset Naming :

"All file names in S-57 are restricted to the limits described in ISO 9660, level 1; that is, file names may be composed of the upper case alphanumeric characters A to Z, digits 0 to 9 and the special character _ (underscore). The file name may be up to 8 characters long with an extension of 3 characters. The separator must be the character . (period). Directory names may be up to 8 characters long." (See also

In order to conform with ISO 9660 level 1 file names must be composed of only upper case characters A to Z, digits 0 to 9 and _ (underscore). The filename may be up to 8 characters long with an extension of 3 characters. The separator must be the character . (period). Directory names may be up to 8 characters long.

The producing authority of the ENC must be given in the cell file name and in the “Producing Agency” [AGEN] subfield of the “Data Set Identification” [DSID] field.

JP: Is this still valid? Can the ENC file name be longer than 8 characters, especially in light of the fact that the first two are already pre-designated. In addition, the third character is reserved for the usage band designator, however, since the structure has changed is this still valid. I realize there needs to be a limit but what is the limit.

EM: after internal discussion on this, it is Jeppesen's recommendation to retain upper case, underscore, alphanumeric characters and digits rules. We are not too concerned about length of file name and extension. But legacy systems need to be considered before changing the file naming conventions.

CDOB comment on 15.4 File Naming

This clause indicates that file names in S-101 must comply with ISO 9660 level 1 file naming conventions. This is an old, and very constraining, convention resulting from the original limitations of Microsoft DOS (and from even earlier DEC file naming rules). The difficulty is that the constraints of Microsoft 8.3 DOS file naming and ISO 9660 CD Rom file naming still continue today because the "backward compatibility" software built into the Microsoft, and Apple operating systems still can truncate filename to the original short file name conventions without the user realizing that this has happened. If the ENC file names were changed to a short file name form because the ECDIS operating system inadvertently called on the "backward compatibility" file name changing software, there could be significant confusion. To me this is a poor type of one way "backward compatibility" because it continues to limit systems in the future. In a way it is an automatic file name corruption capability.

The problem exists in case the Microsoft file system encounters a device which only supports the older FAT file structure, and there are still many of these devices about including some USB key drives. The current file structure for Microsoft systems is the "Microsoft NT File System" (NTFS), the details of which are proprietary Microsoft information. There have been many versions and the latest version for the Microsoft Visa operation

system behaves a little differently than the previous Microsoft XP version and each version before that going back to the early 1980s. NTFS supports up to 255 UTF16 (2-byte) characters.

If the filename follows the original old Microsoft 8.3 DOS file naming then this filename is put in the longer NTFS filename field without any conversion (and is a stable name). If the old rules are followed the automatic filename conversion capability is never invoked. However if the filename is longer or includes a mix of upper and lower case characters, and/or "special" characters, then Microsoft automatically generates a second hidden FAT filename. A complex conversion algorithm maps to the 8.3 filename structure, mapping lowercase to upper case, mapping special characters and using a hash of the original filename. The process is not even repeatable in different versions of the Microsoft operating system, and is certainly different mapping from the Microsoft NTFS to the Apple Macintosh 31 byte file name limitation or the Unix file name limitations (which are primarily special character limitations).

Other limitations in the file naming system also exist, such as the overall limitation of a file record including the metadata such as the creation date, hidden status, etc, and the full path through the directory system. This means that the concatenated string of all nested directory names and the filename are also limited differently on some media and versions of the operating system, resulting in a truncated string.

ENC data MUST be independent of the hardware systems upon which they are implemented. There needs to be freedom for manufacturers to be able to use any platform to cost effectively implement an ECDIS system. The difficulty is that some of the legacy limitations of the media and operating systems severely limit what can be done.

The safe approach is to continue to impose the limitation that ENC file names be limited to :

8 uppercase characters (A-Z and 0-9), followed by a period, plus
3 uppercase character extension, and
to disallow the characters " * / : < > ? \ | and
to disallow SPACE except as a trailing padding character

with directory names limited to 8 uppercase characters.

If this is too restrictive, then S101 should limit characters to 31 characters (so as not to bring in similar Mac issues), resulting in:

27 uppercase characters (A-Z and 0-9), followed by a period, plus
3 uppercase character extension, and
to disallow the characters " * / : < > ? \ | and
to disallow SPACE except as a trailing padding character

with directory names limited to 31 uppercase characters.

This approach still can invoke the worst part of the automatic filename algorithm, the replacement with a hash of the original filename.

Note that using all of the possible combinations in an 8 character field will possibly generate politically incorrect words in different languages. Filtering out "bad" words even more severely restricts the available choices.

Recognizing that at times this longer filename might invoke a filename mapping corrupting the ENC tile naming convention, it is recommended that the full filename for an ENC file also be placed as metadata within a file so that it would be possible to recover from such a file name corruption on some systems.

Text Coding

Text coding has evolved significantly since the definition of the lexical levels in old S-57. In those days the choice was between the single character set ASCII (technically ISO 646 International Reference Version), ISO 8859-1 (so called 8-bit ASCII), an accented character form ISO 6937 which created non spacing combined characters for use on teletypes, and the 2 byte coding of ISO 10646 (Unicode). S-57 introduced the lexical level concept so that one could separate the character repertoire needed from whichever coding approach was being used.

The concept of lexical levels are still valid, but the encoding issues have virtually been solved. The internet has standardized on the encoding UTF-8, which was not even available when S-57 was first developed. This is an expandable encoding which uses 1, 2, or more bytes of data (up to 6) to represent all possible characters. However, the first "page" of characters, exactly matches ASCII. This means that almost all English based text is represented in single byte characters, and accents are handled by an extension to a second "page" of characters in 2 bytes. Iconographic character sets, such as Japanese Kanji, can be handled in 2, 3 or 4 bytes. UTF-8 is unique. There is only one way to represent each character. This means that UTF-8 should be the only character encoding chosen for S-101.

The lexical level concept should remain at the information content level and indicate what a particular data field contains, either a machine readable character field such as a file name, or text that is intended to be read by humans. A lexical level distinction may be made between text that contains only western characters, or the iconographic character sets of some eastern languages, because the iconographic character sets may not be implemented in some systems. This is not the way identifying character repertoires tends to be done in many systems, but it is a simple distinction that is well know in IHO.

Therefore, S101 should only use UTF-8 for encoding and have three lexical levels: Machine Readable ASCII only, Western phonetic characters, and iconographic characters.