S100WG01 8.17

Paper for consideration by S-100 WG

Inheritance and super/sub-types in feature catalogues

Submitted by:	Jeppesen
Executive Summary:	Clarification on super-types and sub-types in feature catalogues.
Related Documents:	(1) Part 5 FC Supertypes - proposal
Related Projects:	(1) S-100 2.1.0

1 Introduction/Background

This document discusses the issue of inheritance in feature catalogues and a clarification on the discussion in S-100 Part 5 on super-types and sub-types in feature catalogues.

2 Analysis/Discussion

Feature catalogues are expected to be consistent with the application schema's specification of attributes, relationships, and constraints for feature and information types. One way to achieve this is to replicate the inheritance hierarchy in an application schema in the feature catalogue. On the other hand, S-100-based applications at present use only instantiable (non-abstract) features and information types, and including an inheritance hierarchy in feature catalogues means extra processing because properties defined in super-types must be inherited by sub-types in application logic, sometimes with constraints that are specific to the sub-type (e.g., allowed value lists for enumeration attributes).

Inheritance is permitted in S-100-based feature catalogues, but S-100 Edition 2.0.0 clause 5-4.2.2 concludes "It is always good design practice to keep the depth of an inheritance tree as shallow as possible." In principle this could be done by reducing the depth of inheritance trees to zero, i.e., having no super-type/sub-type relationships in a feature catalogue. Even if there are inheritance relationships in an application schema, they could be excluded from the feature catalogue by copying attributes, associations, and constraints to sub-types at all levels.

The advantage of excluding inheritance from feature catalogues is mainly structural simplification (and consequently simpler processing) since abstract types and inheritance hierarchies need not be implemented; also in S-100-based product specifications, inherited enumerated attributes can have different lists of allowed values for different sub-types. The disadvantages include (probable) increases in the volume of the feature catalogue especially if many instantiable features or information types have common attributes or associations, and increased complexity for maintenance (an update to an attribute nominally bound to a super-type would have to be made to each sub-type at all levels, and this would have to be checked before the feature catalogue is released). Also, inheritance is a common paradigm in object-oriented programming.

In general the need for inheritance increases with increasing numbers of closely related feature or information types, or as more characteristics are shared between similar types, or even if several different types share some characteristics. For example, in S-101, all geographic feature types have information bindings to the information type **SupplementaryInformation** and feature bindings to the cartographic feature **TextAssociation** and since there are approximately 170 features in S-101 there is a good case for a common super-type to bind these associations instead of repeating them in every feature type.

There is no universal answer as to whether or not to use inheritance; it is an information modeling issue and the matter should be left to project teams, considering factors such as application schema and feature catalogue complexity, maintenance, etc. A complete analysis of when and where to use inheritance would be quite complex and would not result in more than guidelines for product specification project teams, and will therefore be foregone at this point. The general guidelines mentioned above are included in the proposal.

Part 5 of S-100 implicitly permits the use of inheritance and this should continue to be the case. A clarification to the discussion of inheritance in S-100 Part 5 may be useful and the accompanying proposal contains draft language.

3 Recommendations

Add a clarification to the discussion of inheritance in S-100 Part 5 and leave determinations of whether or not to use inheritance to the product specification project teams.

4 Impacts

No change to the current situation for super-types/sub-types in feature catalogues. Some clarifying material added in S-100 Part 5.

5 Actions Requested

The S-100WG is invited to:-

- discuss this paper;
- adopt the accompanying clarifying update to Part 5.

Annex A. Detailed Comments

Clause / sub-clause / annex	Paragraph / Figure / Table / Note	Type	Comment – justification for change	Proposed change	Observations
all		te	Needs to be updated in conformance with ISO 19115-1:2014. This action would ideally be synchronized with the XML schema implementation of metadata, ISO/TS 19115-3 but the latter has not yet been released by ISO. As of January 12 2016 it is in DTS stage and a mature version of the schemas is available though not the text.	Depends on the status and ISO plans for ISO 19115-3. Alternatives: 1) Proceed with updating the text (only) of S-100 Part 4A. Update the XML schema implementation ASAP when ISO 19115-3 is released. 2) Hold back on updating the text of S-100 Parts 4A-C until ISO 19115-3 is released.	
Overview	Figure 4A-D.4	te	ISO 19115 already defines a citation type CI_Citation type which can be used instead of S100_SupportFileSpecification and S100_ProductSpecification. CI_Citation has attributes title, edition, and editionDate as substitutes for name/version/date	1) Delete S100_SupportFileProductSpecification and S100_ProductSpecification classes 2) Change the 3 attributes using them as follows: S100_DatasetDiscoveryMetadata.productSpecification : CI_Citation [1] S100_ExchangeCatalogue.productSpecification : CI_Citation [0..1] S100_SupportFileDiscoveryMetadata.supportFileSpecification [0..1] (mandatory if and only if dataType value is not Text). 3) Add attribute locale: PT_Locale [0..1] or characterSet: MD_CharacterSetCode [0..1] to S100_SupportFileDiscoveryMetadata for the character encoding.	

Overview	Figure 4A-D.4	te	S100_DatasetDiscoveryMetadata.dataCoverage and S100_DataCoverage.boundingBox are both mandatory in this diagram, implying that a geographic bounding box is required in metadata. But there may be S-100 products where a bounding box is not defined, e.g., “Norwegian ports”. In the original, bounding box/polygon were optional. ISO 19115 defines dataType EX_Extent which allows use of EX_BoundingPolygon, EX_GeographicBoundingBox and a text attribute called “description”.	Add attribute extent: EX_Extent [0..*] to S100_DatasetDiscoveryMetadata for the dataset bounding box. Make dataCoverage optional (mult. 0..*) in S100_DatasetDiscoveryMetadata. Change S100_DataCoverage replacing attributes boundingBox & boundingPolygon with attribute extent: EX_Extent [0..*] with a constraint requiring (either?) EX_GeographicBoundingBox or EX_BoundingPolygon. (Also, extent.description can replace ID.)	
S100_SupportFileDiscoveryMetadata		te	checksum and digital signatures are different, a file might have a checksum but not be signed	restore the checksum attribute but make it optional (multiplicity 0..1)	
S100_SupportFileFormat	literal ASCII	te	S-100 text is in UTF8, not ASCII	Revert literal from ASCII to Text	
S100_CatalogueMetadata	attributes	te	Why do fileName, fileLocation, versionNumber, issueDate, and productSpecification have multiplicities of 1..* instead of 1, 0..1? It will need some complex constraints to match the numbers of these attributes. If it is to allow for multiple catalogues, an upper bound of * on the role multiplicity in figure 4a-D.3 should be able to handle it. Alternatively, a complex attribute to group the info for each catalogue should be introduced.	multiplicity of all attributes should be [1] or [0..1] multiplicity at appropriate association end in figure 4a-D.3 should be 1..* or 0..*	
S100_CatalogueMetadata	productSpecification	te	CI_Citation can be used instead	change type to CI_Citation	
S100_CatalogueMetadata		te	catalogues might be in different languages and character sets	Add as attributes either: locale: PT_Locale OR, both of: language: ISO 639-2 & characterSet: MD_CharacterSetCode.	

S100_CatalogueMetadata	digitalSignatureReference & digitalSignatureValue	te	What is being signed here, why, and who will sign it? If for signing catalogues – will it be the IHB or each producing agency? Will we end up with all producers having to sign all portrayal and feature catalogues? Will a system end up with multiple copies of catalogues differing only in who signed them? If some need to be signed, does signing have to be mandatory for all? At present it is not clear why digital signatures are needed here, the same is goes for the implications of requiring signatures here. Checksums might suffice to guard against transmission or other corruption.	The remarks column should describe what is being signed. Analyse the implications of signing and explain them in a new clause. Make digitalSignatureReference and digitalSignaturevalue optional.	