John Arlinger 16 April 2013

Minutes StanForD2010

2013-04-10--11

Place:	Skogforsk, Uppsala 10:00-13:30 CET
Participants:	Mikael Eliasson, LogMax Teemu Paldanius, Ponsse Tapio Räsänen, Metsäteho Timo Melkas, Metsäteho Erik Kindlund, John Deere Nazmul Bhuiyan, Skogforsk Björn Hannrup, Skogforsk (§13-14) Niclas Eriksson, Komatsu Forest Uli Dietz, KWF Lars Henriksson, SDC Joakim Nygren, Haglöf (day 1) Antti Yli-Fossi, John Deere (phone) Lars-Erik Andersson, Rottne (phone) Jörgen Eriksson, Dasa (phone, day 1) Jonas Thorsson, Dasa (phone, day 2) Jean Plamondon FP Innovations (phone) Pasi Palonen, Parker (phone) John Arlinger, Secretary Johan J. Möller, Chairman

§ 1 Opening of meeting

Johan J Möller opened the meeting.

§ 2 Agenda

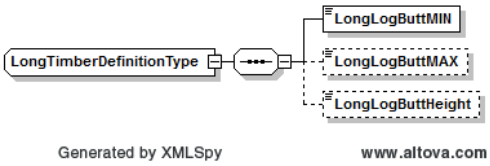
The agenda was accepted by the meeting.

John presented the roadmap for StanForD2010 that was decided at the last meeting (2012-11-14—15). John pointed out that all decisions at the meeting will be included in new 3.0 pre-release schemas.

§ 3 Clarifications

Long logs definition

Based on questions from Ian Wilson, Australia, 2012-12-14.



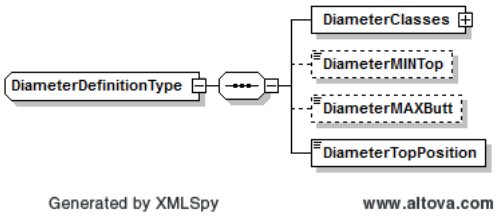
Ian suggested that we clarify how under vs on bark is handled for elements LongLogButtMAX and LongLogButtMIN (old pole matrix). These are to be dependent on element DiameterUnderBark (boolean) and LongLogButtHeight.

It was decided that the descriptions of the elements are updated according to the following table (new text underlined):

Name	Description	Unit	Type	OldVarNo
LongLogButtMIN	Lower limit for butt diameter per length class and product. Elements DiameterUnderBark and LongLogButtHeight defineas how to measure this diameter.	mm	nonnegative-Integer	v165_t1
LongLogButtMAX	Upper limit for butt diameter per length class product. Elements DiameterUnderBark and LongLogButtHeight defines how to measure this diameter.	mm	nonnegative-Integer	v166_t1

DiameterMAXButt definition

Based on question from Anders Nylund, Komatsu, 2012-12-17.



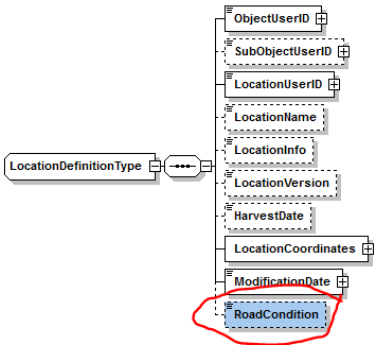
Element DiameterMAXButt has the definition "Maximum diameter in the large end of the log per product. (V134_t2)". We must clarify the description in order to specify at what position DiameterMAXButt is to be measured. We could either state that this is always to be measured at the butt end or we could add a new element DiameterButtPostion ("Position from butt end of log where butt end diameter is measured.")

It was decided that the description of DiameterMAXButt is updated according to the following (new text underlined):
Maximum diameter at the large end of logs, per product. This means the diameter at the physical butt end of the log which is identical with the maximum diameter of the whole log.

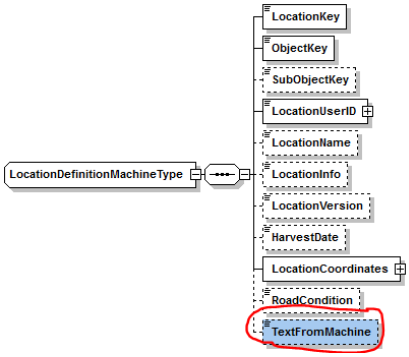
RoadCondition and TextToMachine in forwarder files

Based on discussion with Patrik Andersson, Södra, 2013-02-06.

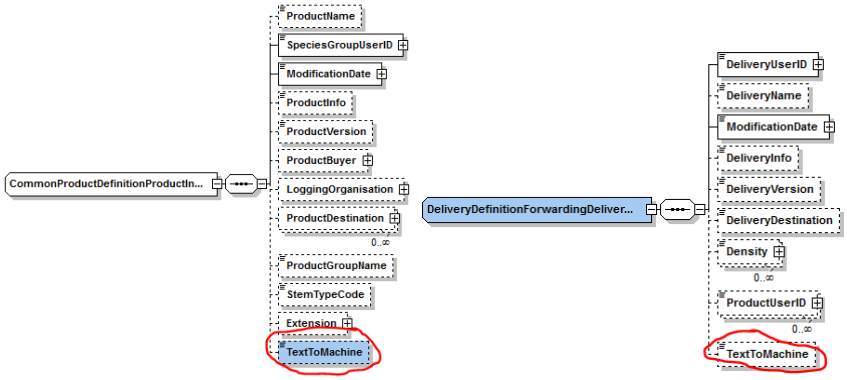
Södra is today sending out information about road condition on paper/pdf to the forwarder operator which is then registered manually by the operator in var441_t13. It is therefore was suggested that optional element RoadCondition is added also to LocationDefinition in Forwarding object instruction (foi-message), not only in the fpr-message. It was decided to postpone this proposal and that the possible use of UserDefined data should be analyzed.
The reason for this update is that the data in the fpr-message goes directly from SDC to the provider of transportation from forest to industry.

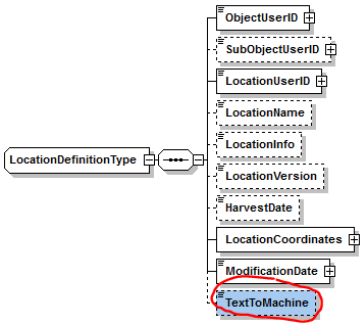


It was suggested that we add element TextFromMachine to LocationDefinition in fpr-message. This element is comparable with the present var441_t9 which is used by Södra today in order to communicate free-text information to truck driver.

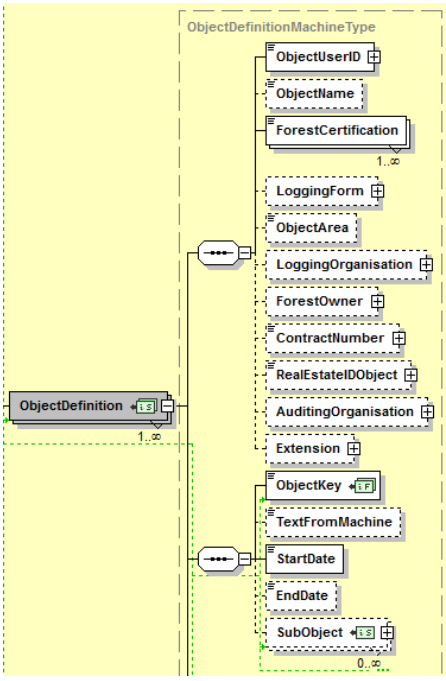


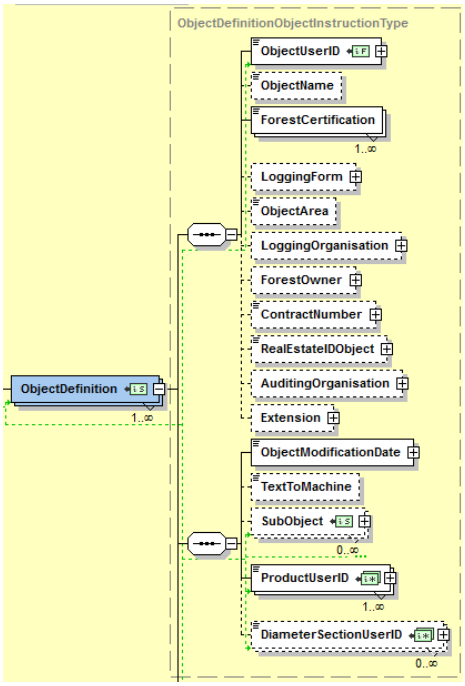
It was also suggested to add optional element TextToMachine to ProductDef, DeliveryDef and LocationDef in order to communicate "free text" information to operator.





It was discussed at the meeting whether elements TextFromMachine and TextToMachine are suitable. As illustrated below they do exist today for the ObjectDefinition:

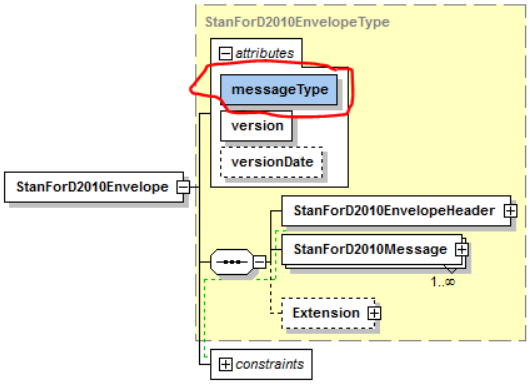




It was decided to postpone this issue, examples are needed as well as an analyses of whether it would be better to include a new element XxxxDescription or AdditionalText.

Attribute messageType in envelope

Antti Yli-Fossi, John Deere (2013-02-20), has noticed that unlike the rest of the messages, the root element in the envelope does not contain the attribute "messageType" of type "MessageCategoryType". The type itself contains "env". It was therefore decided to include this attribute in the env's root as well along with the version attribute. The rest of the attributes in MessageAttributeGroup are obviously not applicable.



Negative price values

Suggestions based on discussion at meeting between Skogforsk and Metsäteho, Helsinki (2012-12-19).

It was suggested to make Price a nonNegativeInteger instead of an integer since negative values are not relevant.

Niclas pointed out at the meeting that Komatsu can handle negative price values in MaxiXplorer. Anders Nylund, Komatsu, commented after the meeting that the issue of allowing negative values in var162 was discussed at the meeting 2000-04-10 (§2) when the limitation matrix was implemented. It was then decided that negative values are allowed.

The meeting decided to postpone the issue.

The secretary will propose that we change the data type to nonNegativeInteger at the next meeting as well as adding to the annotation for element Price that "Possible negative values originating from an apt-file must be converted to the value 0" as discussed with Anders Nylund after the meeting.

Negative measurement data

Suggestions based on discussion at meeting between Skogforsk and Metsäteho, Helsinki (2012-12-19).

Many measurement data elements have the data type positiveInteger or positiveDecimal (e.g. ReferenceDiameter, LogLength, LogDiameter) which means that the value 0 is not allowed.

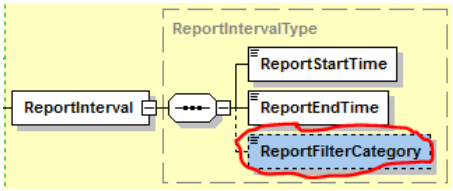
It has been noted that some other elements including measurement data have the data type nonNegativeInteger or nonNegativeDecimal which means that the value 0 is allowed. This is somewhat confusing. If for example length and diameter of a log is larger than 0 the volume of the log must be larger than 0. It was suggested that nonNegative data types are changed to Positive for elements DBH, LogVolume and StemVolume.

It was decided to reject the proposal and keep the data type as it is since for example a stem theoretically can be shorter than 120 cm, and in case of mult tree felling athe estimated volumes may in some cases be 0.

Mandatory ReportFilterCategory in mom

Element ReportFilterCategory was added at last meeting with the following enumerations: No filter, Object filter and Other filter.

Observe that "No filter" means that "All run (work) times, down times and meal breaks that occurred between ReportStartTime and ReportEndTime are included, with no exceptions. This means that any times not registered are to be considered to be unutilized." The value "Object filter" means that only times directly connected to a specific object are included.



It was decided that ReportFilterCategory is made mandatory in 3.0 in order to always make it possible to know if a mom-file is time oriented or not. It was also decided to exclude enumeration "Other filter" since this kind of enumeration will only make it possible to tell the receiver of data that the type of filtering is "unkown" and that the data is therefore more or less impossible to use.

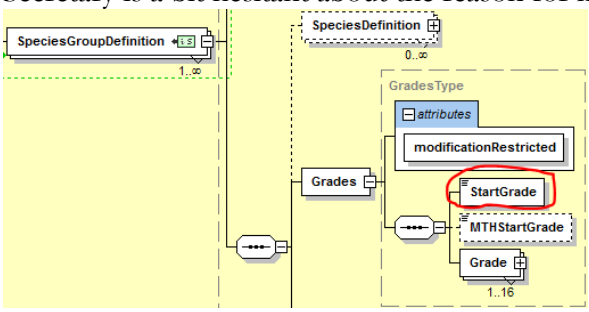
Grades

Suggestions based on discussion at meeting between Skogforsk and Metsäteho, Helsinki (2012-12-19).

Based on the experiences from the Apteri system it was discussed whether:

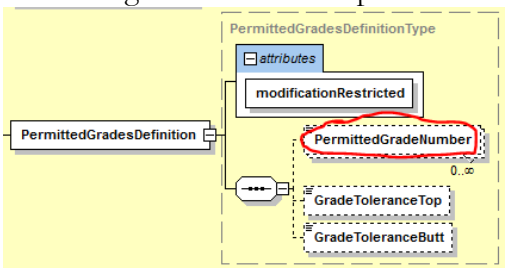
- StartGrade is to be made optional in spi
- PermittedGradesDef is to be made optional in pin

Secretary is a bit hesitant about the reason for making StartGrade optional:



How to interpret if PermittedGrades are totally excluded for a product (ProductDef in pin-file)? Normally this means that the product cannot be selected at all by the bucking optimization software. The operator must manually give the product at least one PermittedGrade value.
Since PermittedGradeNumber is already optional there is no real need for making

PermittedgradesDefinition optional:



It was decided not to change the present elements at all.

Declarations in root element of messages

It is stated in Rule 12.2 (page 39) in document "StanForD_2010_naming and design rulesV1p5":

> *StanForD 2010 messages MUST conform to the following physical layout as applicable:*
>
> *<!-- ======= XML Declaration======== -->*
> *<?xml version="1.0" encoding="UTF-8"?>*

It has been noted by Niclas Eriksson that attribute *encoding="UTF-8"* is optional according to the XML W3C standard, e.g. .Net does not print this to a message by default.

Most parsers interpret an xml-file in the following way:

1. If encoding exists then try to use if. If there is a mismatch with BOM information then the result is a "Fatal error"
2. If encoding is missing then use BOM to decide if it is UTF-8 or UTF-16
3. If encoding and BOM are missing assume UTF-8

BOM means "Byte Order Mark". Two bytes in the beginning of the files specifies the byte order which is needed if UTF-16 is used. Normally does not exist otherwise.

It is stated in "Rule 12.3" (page 39) in document "StanForD_2010_naming and design rulesV1p5":

> *The root element of each StanForD 2010 message MUST have following namespace declarations and schema location attribute:*
> ```
> xmlns:sfd="urn:skogforsk:stanford2010:<version>"
> xmlns="urn:skogforsk:stanford2010:<version>"
> xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
> xsi:schemaLocation="urn:skogforsk:stanford2010:1:draft <URI
> path to schema file>"
> ```

The issue of whether xmlns:sfd="urn:skogforsk:stanford2010:<version>" really needs to be mandatory needs to be addressed since it in many cases is not relevant according to several of the manufacturers. However, in SDCs present implementation for reading hpr this is necessary. A preliminary "fix" has been developed whereby this information is added to the file if it is missing.

Antti Yli-Fossi wrote the following comments (2013-02-14):

The HPR's default namespace is set to be "urn:skogforsk:stanford2010" (that's the xmlns="urn:skogforsk:stanford2010"). This implies that all the elements without a name prefix belong to that namespace. Unless the document contains elements prefixed with "sfd:", e.g. <sfd:HarvestedProductionHeader>, defining the sfd namespace is not necessary, and as every element (except for xsi namespace types, and that's just the schemaLocation attribute) is thought to belong to the "urn:skogforsk:stanford2010" namespace.

Practically this means: the component we use to write the final document recognizes that all elements (by convention attributes are not in a namespace, as they don't need to be) belong to the same namespace, and thus omits the sfd: prefix, marks the default namespace as "urn:skogforsk:stanford2010", and as there's no sfd: prefix, it also notices that we don't need to define xmlns:sfd at all and hence it doesn't get written.

It was thus decided that:

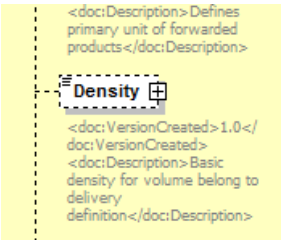
- rule 12.2 is changed in order to point out that encoding attribute is optional.
- rule 12.3 is changed in order to point out that namespace prefix sfd is optional in StanForD2010 messages.

Density

Suggestion from Teemu Paldanius, Ponsse (2013-04-02)

Element Density was made repeatable in fpr and fdi message at the meeting in November 2012.

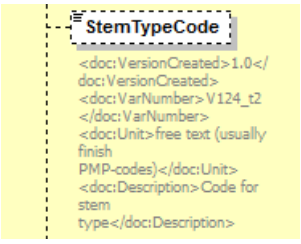
It was decided to make the element Density repeatable also in the fqc-message since it is presently not repeatable.



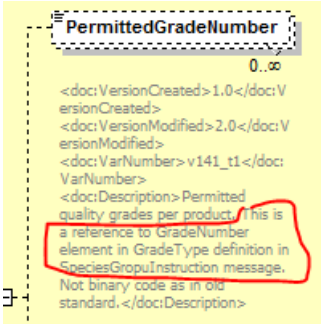
StemTypeCode

Suggestion from Teemu Paldanius, Ponsse (2013-04-02)

StemTypeCode is now defined as free text with description "Code for stem type" in ProductInstruction message.



Teemu thinks that it should be somehow pointed out in the definition that the code also must be included in the element StemTypeDefinition (SpeciesGroupInstruction message). This could be done in a similar way as PermittedGradeNumber is a reference to the GradeType definition in SpeciesGroupInstruction.

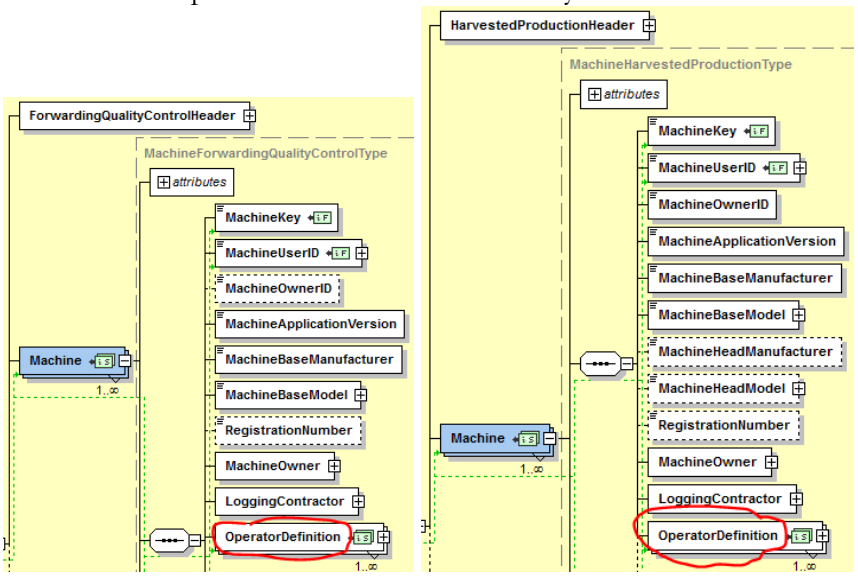


It was therefore decided that the annotation description of StemTypeCode in pin schema is to be updated according to the following text (new text underlined):
<doc:Description>Code for stem type. This is a reference to StemTypeCode element in StemTypeDefinition in SpeciesGropuInstruction message.</doc:Description>

Operator definition and "empty" production files

Based on discussion with Teemu Paldanius, Ponsse (2013-03-22).

The element OperatorDefinition is mandatory in all files.



It was pointed out by Teemu that many files, e.g. fpr, hpr, thp, fqc, may not include any production data at all. This may occur if we send a file including only the EndDate to indicate that logging is completed or if we send a production file before we have actually started any logging activities. After some e-mail discussions we came to the conclusion that we should probably keep OperatorDefinition mandatory and recommend that at least the "active operator" is registered in the OperatorDefinition in these cases.

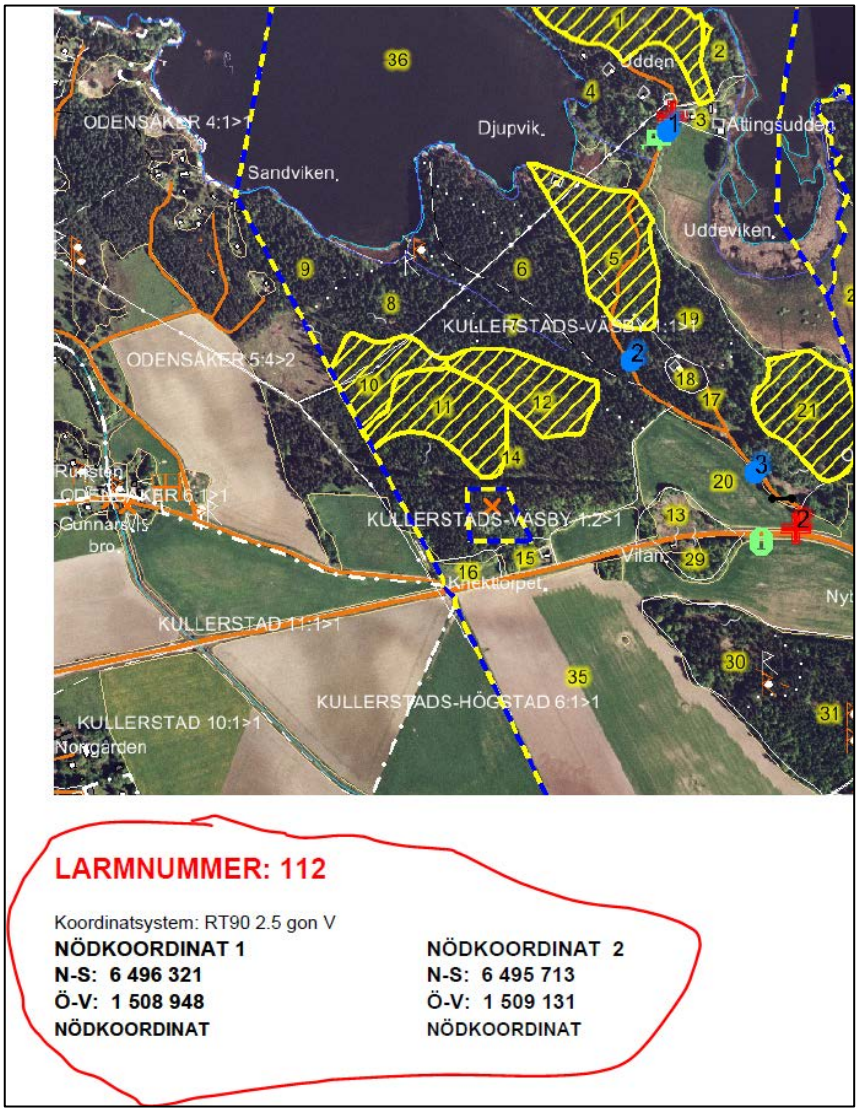
Observe that no decision was made concerning this issue.

§ 4 New elements and attributes

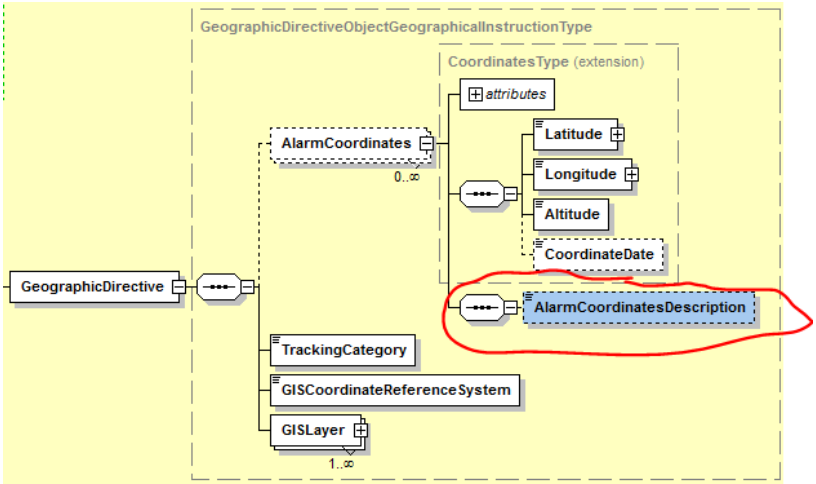
Alarm coordinates in ogi

Secretary has received some comments (LogMax, August 2012) regarding Alarm coordinates registered according to figure below.

It is common in Sweden that the alarm coordinates are printed in a pdf-instruction using some local reference system like for example SweRef99 as illustrated by the following figure from a pdf document.



A possible solution could be to include alarm coordinates as free text where any reference system might be used in order to communicate this information to the operator if he e.g. wants to tell the position over the phone.The suggestion was updated during meeting:



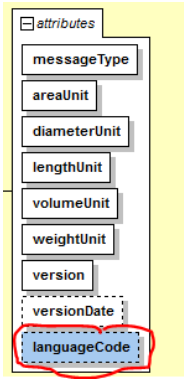
It was decided to postpone the issue in order to have more thorough discussions with Swedish forest companies. Both Johan and Erik pointed out that they have experienced situations where there have been problems in communication the alarm coordinates in emergency situations.

Language attribute in message header

Proposal from Tapio Räsänen (2012-09-27).

In the Finnish StanForD meeting (September 2012) a proposal to add a new attribute to "User defined data" for language information was discussed. The idea was that the attribute would tell the language of the message content: Finnish, Swedish, English and Portuguese etc. This would help in the control of the user interface. The attribute could be added to the root level together with messageType, areaUnit etc. and the enumerations should be according to ISO standard.

It was suggested to add optional attribute languageCode with the following definition to all schemas: *The ISO language code that describes the language that is used in message content. Eg. fin (Finnish), swe (Swedish), fra (French), deu (German), Unit is ISO 639-3 code*



The previous meeting decided to postpone this issue since there was no clear description of how to use it.

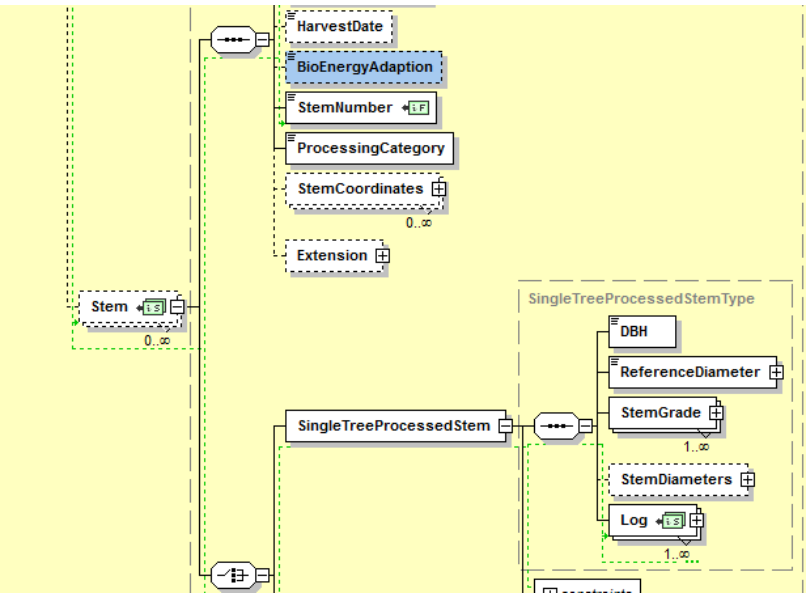
The present meeting decided to reject the proposal since there did not seem to be any obvious benefits.

Tree codes

The following issues were discussed at the two previous meetings.

Both German and Finnish participants mentioned during the last meeting their interest in being able to add "tree codes" for windthrows and beetle infected trees.

Presently we have the possibility for the operator to register whether or not harvesting was adapted for extracting forest energy assortments in the form of residues and stumps. This information is registered in the element BioEnergyAdaptation per stem. Allowed enumeration values are "None", "Logging residues", "Stumps" and "Logging residues and stumps".



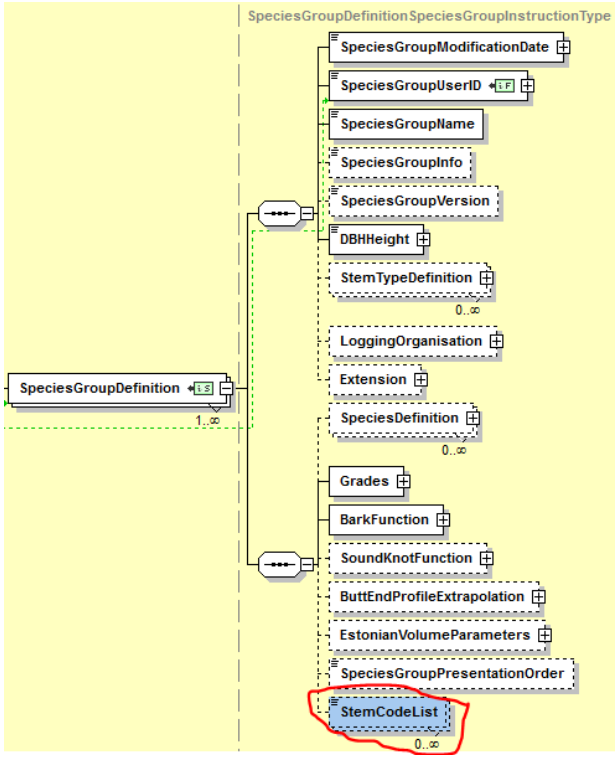
During the last year several forest company representatives in Sweden have requested additional possibilities for registering information about individual stems such as for example high stumps and stump treatment (fungicides).

High stumps

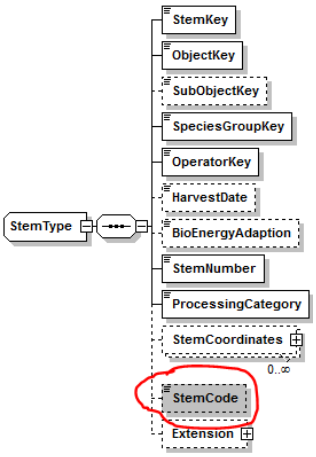
Due to forest certification (FSC) it is now very common that harvester operators are required to register the number of high stumps on each object or sub object. This is today done manually by paper or an old fashion "log counter".

It was therefore suggested that we look at the possibility of using hpr-message for registering some kind of code per stem that for example could be used for indicating that a certain stem was cut in order to leave a high stump (3-4 meter above the ground).

It was decided to add a repeatable element StemCodeList to the spi-schema in order to make it possible to define a set of codes that the operator can select manually in the harvester for each harvested stem:



It was also decided to add an element StemCode per stem to the hpr-schema in order to make it possible for the operator to manually select StemCode for each harvested stem based on the available StemCodeList in the spi-file:



Name	Description	Unit	Type
StemCodeList (spi)	List of stem codes to be registered manually by operator in harvester in element StemCode.	Free text	string
StemCode (hpr)	Stem codes registered manually by operator in harvester. Only codes included in StemCodeList can be selected by operator.	Free text	string

Examples

Spi:

```
<SpeciesGroupStemCodes>Högstubbe/High stump</SpeciesGroupStemCodes>
<SpeciesGroupStemCodes>Kulturstubbe/Cultural</SpeciesGroupStemCodes>
<SpeciesGroupStemCodes>Vindfälle/Wind throw</SpeciesGroupStemCodes>
<SpeciesGroupStemCodes>Barkborre/Bark beetle</SpeciesGroupStemCodes>
<SpeciesGroupStemCodes>Nära evighetsträd/Close to eternal tree</SpeciesGroupStemCodes>
```

Hpr:

```
<Stem>
        <StemKey>12301</StemKey>
        <ObjectKey>1</ObjectKey>
        <SpeciesGroupKey>1</SpeciesGroupKey>
        <OperatorKey>1</OperatorKey>
        <HarvestDate>2010-04-14T11:34:27.059+02:00</HarvestDate>
        <BioEnergyAdaption>None</BioEnergyAdaption>
        <StemNumber>1</StemNumber>
        <ProcessingCategory>SingleTreeProcessing</ProcessingCategory>
        <StemCode>Högstubbe/High stump</StemCode>
```

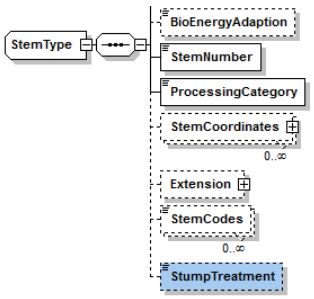
……

```
<Stem>
        <StemKey>12302</StemKey>
        <ObjectKey>1</ObjectKey>
        <SpeciesGroupKey>1</SpeciesGroupKey>
        <OperatorKey>1</OperatorKey>
        <HarvestDate>2010-04-14T11:34:27.059+02:00</HarvestDate>
        <BioEnergyAdaption>None</BioEnergyAdaption>
        <StemNumber>2</StemNumber>
        <ProcessingCategory>SingleTreeProcessing</ProcessingCategory>
        <StemCode>Barkborre/Bark beetle</StemCode>
```

It was discussed at the meeting whether StemCode should be repeatable or not but it was decided to not make it repeatable since it would make the user interface more complex. It will be possible for forest companies to define "aggregated" codes, e.g. including both "wind throw" and "bark beetle" into one single stem code.

Stump treatment

A different issue is the possibility to register if some kind of stump treatment with fungicides was used after felling or not. This is information that is more straight forward to standardize since it is "true or false" and can be registered automatically.



It was decided to reject the proposal. May be discussed again if there is a strong need from forest companies.

BioEnergyAdaption

Proposal based on discussion with Lars Henriksson, SDC (2013-04-08).

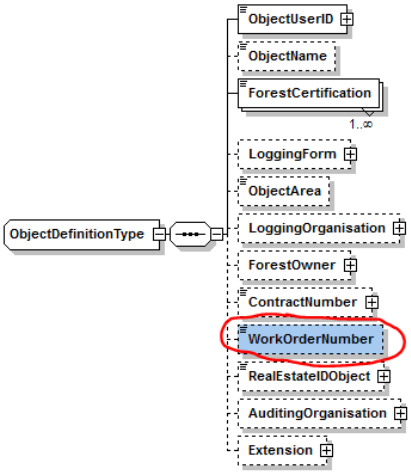
It was discussed during the meeting whether it is relevant to have enumeration "None" and to have the element optional. What is the difference between a non-existing element and the value "None"?

The meeting decided to keep element BioEnergyAdaption optional and to exclude enumeration "None".

WorkOrderNumber (Logging / Forwarding)

There is a Finnish need to give work order numbers (LoggingOrderNumber and ForwardingOrderNumber) in oin and foi and to send them back in hpr, fpr, thp and hqc.

It is suggested that we add new element WorkOrderNumber to ObjectDefintion:

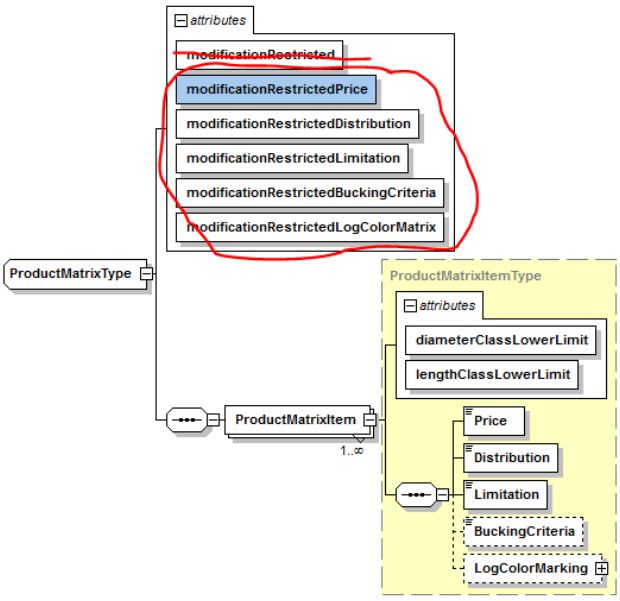


It was decided to postpone this issue to a telephone meeting in August 2013. The Finnish users will discuss the issue before that meeting and Metsäteho will write a description of how WorkOrderNumber is to be used.

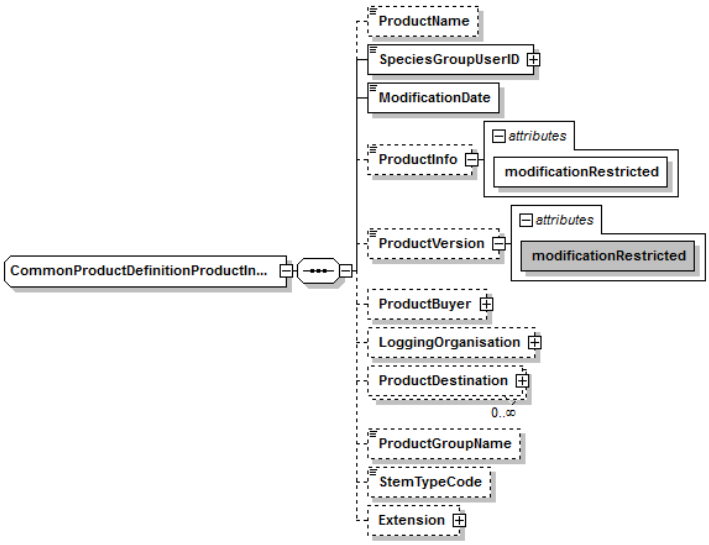
Updating modificationRestricted

Suggestions based on discussion at meeting between Skogforsk and Metsäteho, Helsinki (2012-12-19).

It was decided that modificationRestricted is divided up per matrix in version 3.0 (Price, Colour, Limitation, Distribution and BuckingCriteria) as illustrated below since it is often so that there is a need for not restricting colours but restricting the other ProductMatrixItem elements:



It was also decided to add modificationRestricted to XxInfo and XxVersion as illustrated below. The idea is to make it possible for forest companies to keep the given identification unchanged through the information chain.

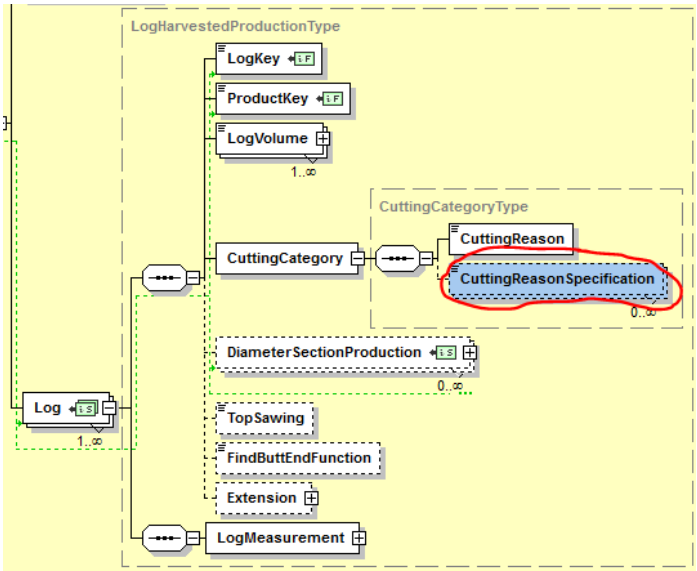


It was also decided that to re-write descriptions to make these id-elements more open (new text underlined):

Name	Description	Unit	Type	OldVarNo
XxInfo	Free text identity information about product. Identity element normally to be used by logging organization. No constraints in standard concerning how to use this element.	free text	string	V121_t2
XxVersion	Additional identity information ~~description~~ of product. Identity element normally to be used by logging organization. No constraints in standard concerning how to use this element.	free text	string	V121_t3

CuttingReasonSpecification

It was discussed at the last meeting whether CuttingReasonSpecification (enumerations Outside cutting window, Quality change or Unclassified log) is really needed since cutting window, unclassified logs and quality changes are known through other elements.



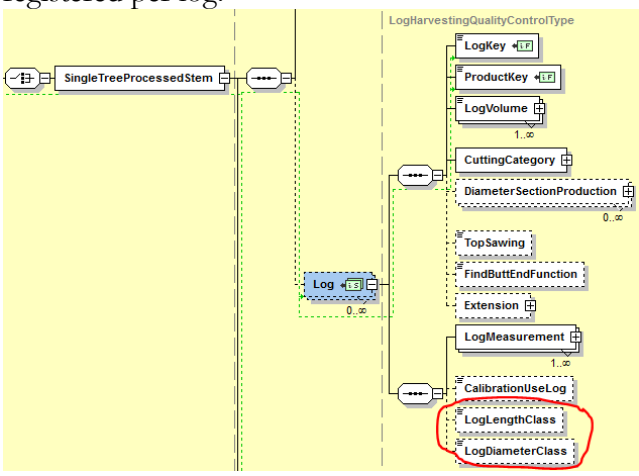
It was thus decided that the element CuttingReasonSpecification is excluded from the 3.0 version.

LogLength- and LogDiameterClass in hpr

Niclas Eriksson, Komatsu (2013-03-26).

It has been noted by Niclas that the length and diameter classes of a log is not registered for each log in an hpr-file.

If we look at the hqc-file it can be noted that both length and diameter class can be registered per log:

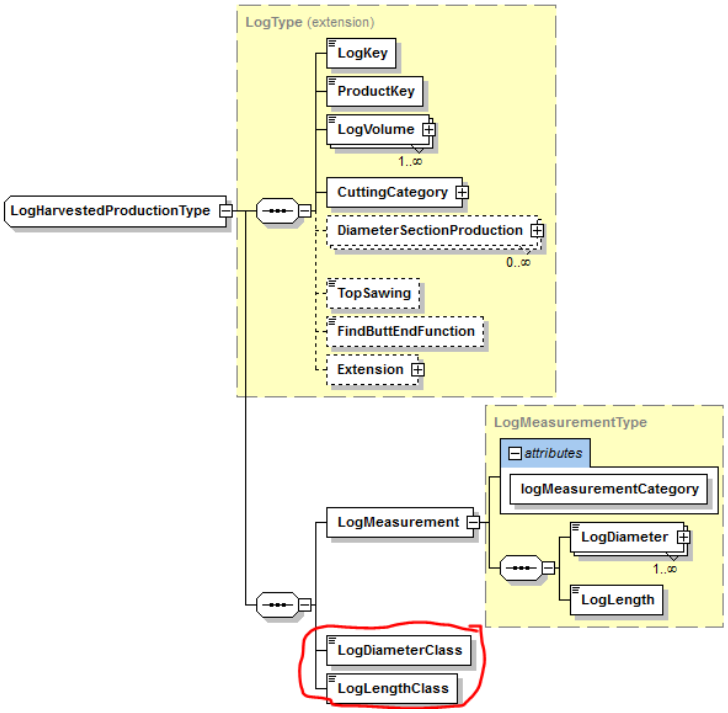


A large number of different parameters in the hpr-file must be considered in order to be able to calculate the length and diameter classes:

- LogLength and LogDiameter in LogMeasurement.

- DiameterClassLowerLimit and LenghtClassLowerLimit and several other elements under LengthDefintion and DiameterDefinition

It was thus suggested that we add to new optional elements (LogLengthClass and LogDiameterClass) per log to the hpr message:



Name	Description	Unit	Type
LogDiameterClass	Harvester registered diameter class of the log. Use value 0 in case of unclassified log.	mm	nonNegativeInteger
LogLengthClass	Harvester registered length class of the log. Use value 0 in case of unclassified log.	cm	nonNegativeInteger

There was a discussion during the meeting about the fact that this proposal means adding information that can be calculated using already existing variables. It was decided to postpone the issue to the next meeting in order to give the manufacturers more time to analyse the proposal.

§ 5 Updating design of some definitions

It was decided to implement the design modifications presented in appendix 1.The need for the new "levels" ProductDetails and SpeciesGroupDetails was discussed. It was suggested at the meeting that they are to be excluded since they create additional and perhaps unnecessary work. Since Juha-Antti Sorsa and Antti Yli-Fossi found a way of solving the issue of ProductDetails and SpeciesGroupDetails without any negative side effects the pre-release schemas published after the meeting were slightly modified when comparing with appendix 1.

Additional proposal from Teemu Paldanius (2013-04-05)

It was decided to change presentation order elements in the common part of the element structure in order to get the SpeciesGroupPresentation and ProductPresentationOrder information also included in fdi and fpr messages.

It was suggested to add new element DeliveryPresentationOrder, because this seems to be missing at the moment. With these changes it would be possible to sort tree species, deliveries and products to given order also in forwarder reports.

The meeting decided to add new element DeliveryPresentationOrder added to a DeliveryDefinitonType.

The issue of using SpeciesGroupUserID instead of SpeciesGroupKey when referencing from ProductDefinition in hpr and thp was brought up again by Teemu. The issue will be discussed again at the coming meeting

§ 6 Forwarding Object Instruction

LOCATIONDEF IN FOI

Original proposal

This issue is based on work done in the project ”Common planning service for contractors” (joint project for Stora Enso, Metsä Group and Metsähallitus). The issue was first raised in July 2012.

A timber sale contract can, in Finland, have several objects. An object is a logging area having the same logging form (final felling, first thinning etc.) and one single harvesting instruction. An object may consist of sub-objects which are normally the forest compartments of the forest plan. Sub-objects are needed to keep harvested volumes of different stands separated for planning purposes.

If objects are close to each other they have only one roadside landing. Piles for different assortments are located at the landing. Timber from all the objects can be forwarded into the same assortment-specific pile. That is why several objects can have one common location.

An example illustrating why one single landing may sometimes be used for several different objects is shown in figure 1. A landing is presently identical with a LocationDefinition in foi- and fpr-files.

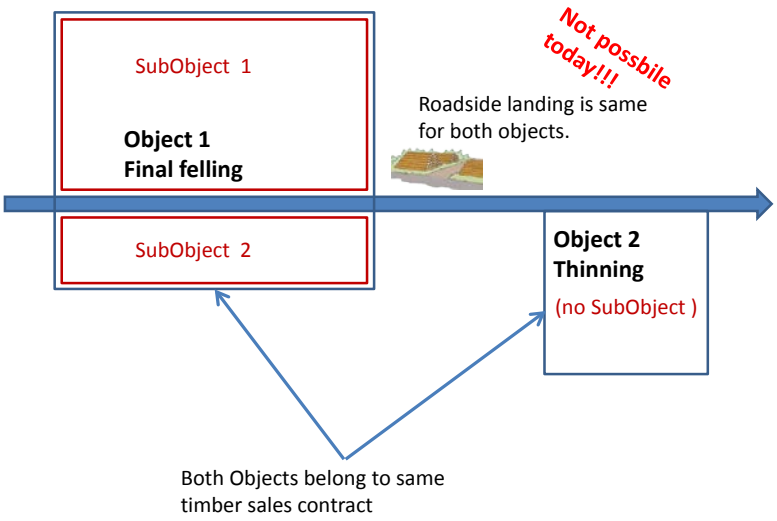


Figure 1. Example illustrating the need for using the same road side landing on several objects.

A problem with the present foi schema is that a specific Location (landing) can only be connected to one single object since ObjectUserID is not repeatable in LocationDefinition (figure 2).

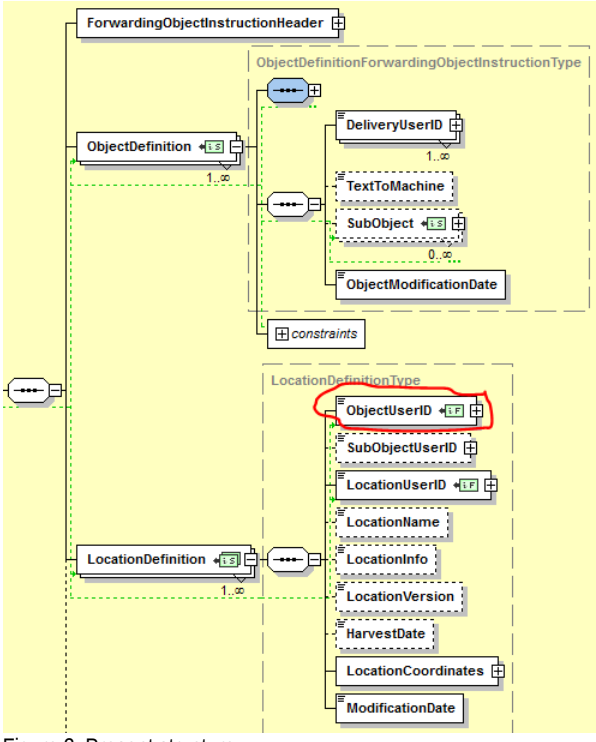


Figure 2. Present structure.

It was suggested that the structure is updated according to figure 2b.

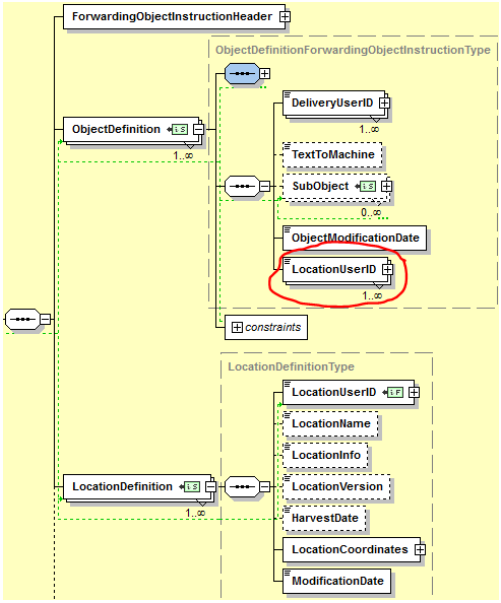


Figure 2b. Suggested new structure

Having the same Location connected two several different Objects means adjustments to fpr are also needed. A possible solution could be to add ObjectKey and SubObjectKey to the PartialLoad.

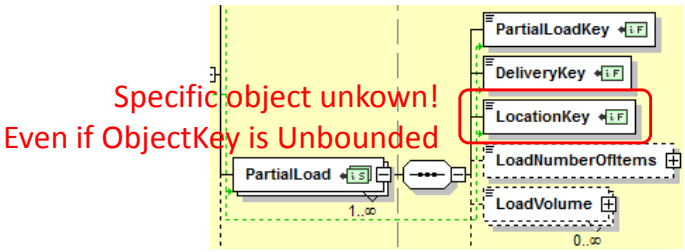


Figure 2c.Illustrating the problem with not knowing the true Object of each PartialLoad

This proposal means some fundamental changes to the data structures connected to forwarder production data and forwarder instructions. Locations will become a new totally independent feature, meaning that a "library" of locations must be introduced. This library must be maintained totally separately from the object. It is clear that at least some of the manufacturers having objections since this whole issue could be solved in an administrative company system using the present structure.

Additional consideration

An important issue to consider is whether there is support in the forwarders for several objects in one instruction and for more than one object per fpr-file. Including more than one object per instruction is possible according to the schemas since the element Object is repeatable. However, it is not known whether anyone has ever tested or developed any applications for doing this. So far it seems that no manufacturers support having several Objects in one single instruction (true for both harvesters and forwarders).

The secretary discussed this issue with Mikael Eliasson and Fredrik Tjernberg after the meeting in Helsinki. The issue was also discussed briefly with the other manufacturers. No one supports several objects in one instruction today and a clear majority seem to prefer that we take away the possibility of having more than one object per instruction.

Fredrik Tjernberg (2012-01-14) clearly stated in our discussion that Komatsu does not really like the idea of creating a separate "library" with LocationDefinitions totally separated from the Objects.

It would make the situation simpler if we clearly stated that we are only to have one object per instruction sent to both harvesters and forwarders. The following examples with two separate files per file type illustrates how to handle the needs describe above:

Example foi (instruction)

File	ObjectName	ObjectUserID	LocationUserID	Location-Info	Location-Coordinates
1	FinalFelling	101	1001	32453	60,12; 13,24
2	Thinning	102	1002	32453	60,12; 13,24

Example fpr (forwarded production)

File	ObjectName	ObjectUserID	ObjectKey	LocationUserID	Location-Key	Location-Info	Location-Coordinates
1	FinalFelling	101	1	1001	1	32453	60,12; 13,24
2	Thinning	102	2	1002	2	32453	60,12; 13,24

Observe that we get the same LocationInfo and LocationCoordinates in both fpr-files sent from the forwarder even if the Objects are different.

It was decided that we only allow one ObjectDefinition per instruction (oin, foi & ogi), means it will not be repeatable.

Observe that a consequence of this decision is that the suggested update (previous section) of foi- and fpr-structures is not relevant any more.

It was also decided that secretary is to write a document illustrating how to create foi based on the fact that ObejctDefinition is not repeatable and that one landing is used on several objects.

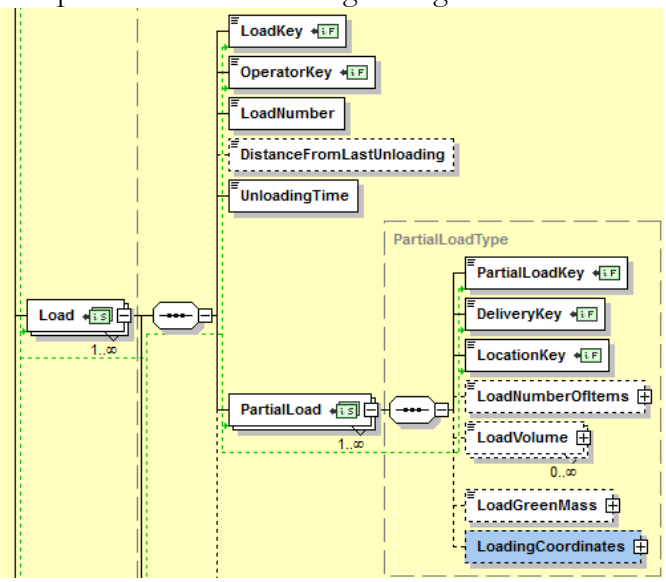
One situation (discussed after the meeting) to consider is if two objects Final felling (1) and Thinning (2) are next to each other and forwarding from both of them is done at the same time and to one location. It might be so that one single load is collected from both of the objects. How can the system in forwarder handle this? Are two separate fpr's made for both objects? It feels unnecessary from practical point of view. Are ObjectKey and SubObjectKey needed for Load or PartialLoad? Good examples and discussions with forest companies are needed before final decisions

§ 7 Product migrations during forwarding

The issue of how to handle product migrations (sortimentsvandring in Swedish) in forwarded production message (fpr) was first discussed during a meeting in Helsinki 2012-10-14. The issue was raised by Tuomo Vuorenpää, Metsähallitus.

It was pointed out at the previous meeting by both Erik Kindlund (John Deere) and Urban Folkesson (Ponsse) that there is an operator need for making adjustments of the forwarded volumes, for example if there has been a manual error adding 100 m3 instead of 10 m3 of a certain product. The conclusion of the secretary was that the simplest solution to both product migrations as well as operator errors is to allow negative volumes in "LoadVolume".

The present structure for registering forwarded volumes is illustrated below:



A possible solution to the problem of handling adjustments/correction/migrations could be to make it possible to also register a negative volume or mass.

Fredrik Tjernberg (discussion 2013-01-11 with secretary) prefers a simple solution where the only modification done is to open LoadVolume and LoadGreenMass for negative values (today only positive values). This means that we can easily handle the most important problem of operator corrections without having to do any structural changes. UnloadingTime should be the time of correction/adjustment.

The following two DeliveryDefinitions are used in the examples below:

Delivery-Name	Delivery-Key	Delivery-Info
Saw logs	11	0110
Pulp wood	12	9010

In case of a simple correction of an error (100 m^3 registered instead of 10 m^3) we could register the following information if done immediately after the first incorrect registration:

LoadKey	Delivery-Name	Delivery-Key	LoadVolume	DistanceFrom-LastUnloading	Loading-Coordinates
1	Saw logs	11	100	943 m	60.12; 15.12
2	Saw logs	11	-90	0 m	60.12; 15.12

If it is a question of products being moved from a Delivery for saw logs to a Delivery for pulp wood we could register the following information:

LoadKey	Delivery-Name	Delivery-Key	LoadVolume	DistanceFrom-LastUnloading	Loading-Coordinates
12	Saw logs	11	-11	0 m	60.12; 15.12
13	Pulp wood	12	11	72 m	60.13; 15.12

If it is a question of products being moved from a Delivery for pulp logs to a Delivery for forest owner fuel wood we could register the following information:

LoadKey	Delivery-Name	Delivery-Key	LoadVolume	DistanceFrom-LastUnloading	Loading-Coordinates
23	Pulp wood	12	-9	0 m	60.13; 15.13
24	Forest owner fuel wood	14	9	410 m	60.09; 15.07

This is a robust and simple solution that does not mean much of an update to the existing systems. If this is not deemed to be enough in the future we could very well come back and try to come up with some more sophisticated solution.

It was <u>decided</u> to allow negative values for LoadVolume and LoadGreenMass (new data data type Decimal).

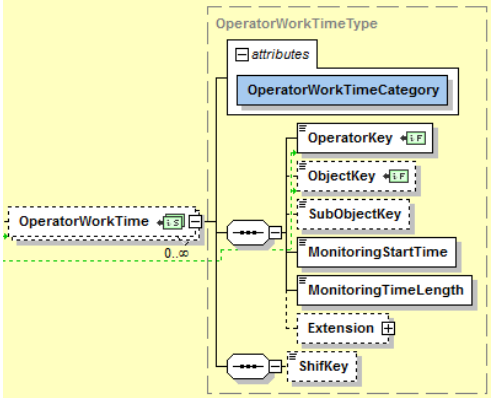
§ 8 Updating Mom-enumerations

OperatorWorkTime enumerations

Based on discussion with Jonas Thorsson, Dasa, 2013-02-05.

OperatorWorkTimeCategory has the following enumerations:

- "MachineRunTime"
- "Waiting for repair"
- "Trailer transportation"
- "Meal break"
- "Planning outside machine"
- "Travelling to work outside machine"
- "Other work outside machine"



The originally existing enumerations Terrain travel, Road travel, Other work, Repair, Maintenance and Disturbance were excluded and MachineRunTime added on 2010-04-14.

The only down times included are “Waiting for repair” and “Trailer transportation” (“OtherMachineDownTimes”). The following types of down times are not included:

- ”Repair”
- “Maintenance”
- "Miscellaneous / other" (Disturbance)
- "Planning/follow up" (Disturbance)
- "Machine stuck" (Disturbance)
- "Weather" (Disturbance)
- "Ordered stop" (Disturbance)

- "Unproductive terrain work" (Other down times)
- "Other" (Other down times)

It was decided that we change the enumerations of OperatorWorkTimeCategory to:

- ~~"MachineRunTime"~~
- "MachineWorkTime"
- ~~"Waiting for repair"~~
- ~~"Trailer transportation"~~
- "Meal break"
- "Planning outside machine"
- "Travelling to work outside machine"
- "Other work outside machine"

The new enumeration list means that we only specify times in OperatorWorkTimes that are not specified IndividualMachineWorkTime or CombinedMachineWorkTime.

MachineDownTime enumerations

Based on discussion with Mikael Eliasson, Log Max (2013-04-03).

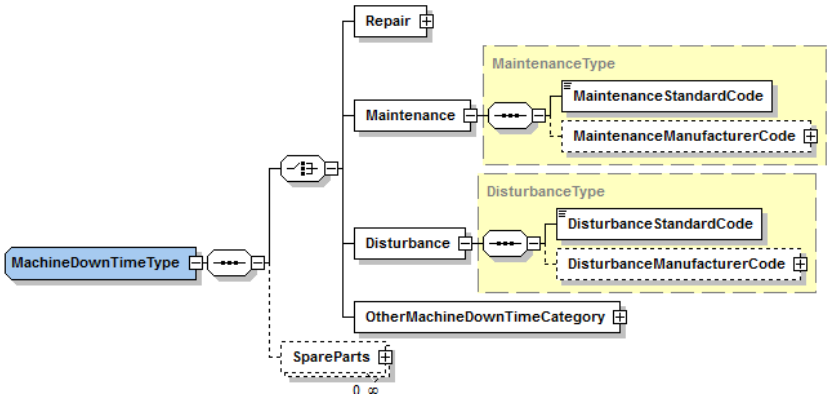
Mikael has noted that the old code 2000 (Administration, telephone) in var330_t3 has not been included in DisturbanceStandardCode. The issue was discovered by a logging team owned by Sveaskog.

The secretary noted that the following maintenance codes from var330_t3 (implemented 2011) are also missing in MaintenanceStandardCode:

- Preventive maintenance 1400
- Periodic maintenance 1500
- Machine wash 1600
- Out of fuel 1700

It was thus decided that we add the following enumerations:

- "Administration, telephone" to DisturbanceStandardCode
- "Preventive maintenance" to MaintenanceStandardCode
- "Periodic maintenance" to MaintenanceStandardCode
- "Machine wash" to MaintenanceStandardCode
- "Out of fuel" to MaintenanceStandardCode



§ 9 Identification issues

Skogforsk and SDC have started testing reporting of hpr-files from a limited number of harvesters during the spring (LogMax, Komatsu, JohnDeere). The basic idea is to send in both hpr and pri/prd in order to make certain that SDCs system for analysing hpr-files gives the same results as the old pri-system.

Skogforsk therefore analysed whether the identity mapping between old and new standard is done according to our recommendations. Unfortunately these mappings were not controlled during the harvester test 2012.

The following mappings of identities were controlled since they are very important from an SDC perspective if we are going to test to send in both pri- and hpr-files druing the spring of 2013:

Structure (Def)	New element	Swedish GUI name	Comments	Old var
MachineDefinition	MachineOwnerID	Ägarens maskin id	Id defined by machine owner	Var003_t1
MachineDefinition	MachineUserId	Maskin id enl	Id defined by logging organisation	var003_t2
ObjectDefinition	ObjectUserId	Objekt id	Id defined by logging organization	Var021_t1
ProductDefinition	ProductBuyer. BusinessID	Köpare, Kod	Used by SDC	var032_t1
ProductDefinition	ProductDestination. BusinessID	Mottagningsplats, Kod	Used by SDC	var032_t2
ObjectDefinition	ContractNo	Virkesordernr	Today normally virkesordernr in Sweden, used by SDC.	var35_t2
SpeciesGroup-Definition	SpeciesGroupVersion	Trädslag, version		var120_t3
SpeciesGroup-Definition	SpeciesGroupInfo	Trädslagskod		Order of species in file (eg stem code 1, var266 in pri-file))
ProductDefinition	ProductVersion	Produktversion	Used by SDC.	var121_t3
ProductDefinition	ProductInfo	Produktkod	Used by SDC. Today normally assortment code (SSTE)	Var121_t2

Observe that these mappings will also be important in the future in all cases of mixing old and new standard (e.g. send apt to machine and reporting hpr from machine).

Secretary presented the mapping document from January 2012 (**appendix 2**) in order to start a discussion about this issue.

It was pointed out by Antti Yli-Fossi (2013-02-14) that we must consider the following issues when discussing identities:

1) Very little or preferably no change at all between Stanford2010 versions in existing mappings. For example if v2p0 mapped old variable varXXX_tY to some element, v2p1 or v3p0 would maintain that instead of moving it under some other element. The standard itself of course evolves, but perhaps that should only affect things that are sf2010-specific only, or add new mappings if really deemed necessary.

2) No ambiguous or redundant mapping. In practice, one variable under one element, with no conditions (e.g. "if this is HPR, SomeUserID is that, but if this is some other file, the same type's same SomeUserID is something else") and no structuring

inside an element's or attribute's value (e.g. <SomeElement>var123_1 var123_2</SomeElement>). If we need to have several old variables in some context, make one element or attribute for each.

Manufacturer identification possibilities

It was also discovered that the implementation of MachineUserID in at least one case differed from our expectations when analysing the identity mapping between old and new standard. Instead of being settable in the machine it is, in this case, used for a hardcoded identity of the machine (e.g. chassis number). This means that it cannot be used by the logging company as was planned by Skogforsk.

Skogforsk fully understand the need for this kind of manufacturer controlled identity of the machine based on discussion with Niclas Ericsson, (2013-02-01). It was decided that a new optional string-element "BaseMachineManufacturerID" with the following description is added to all MachineTypes:

Name	Description	Unit	Type
BaseMachine-ManufacturerID	Machine identity set by the manufacturer of the machine. This is an identity to be used by manufacturer to identify specific machine, for example using chassis number. Not to be modified by operator, machine owner or logging organization.	Free text	String

Observe that the difference from the MachineKey is that MachineManufacturerID can be constant over the lifetime of a machine even if you have a complete computer breakdown whereby the old computer is totally replaced. A new MachineKey must probably be created under these circumstances.

§ 10 A Finnish default spi (including stem types)

Tapio presented the work on a preliminary Finnish default spi.

It was decided in the last Finnish StanForD meeting that Metsäteho is to prepare a default spi-file. First version has been prepared, but not delivered to be commentated yet. A 3.0 version will also prepared with new changes like StemCodeList.

Common stem type codes and names for all users in Finland has been included

- PMP stem type codes
- pine, spruce and birch are practically same with all users
- the fourth species varies: alder, aspen, other broadleaved, other tree species …

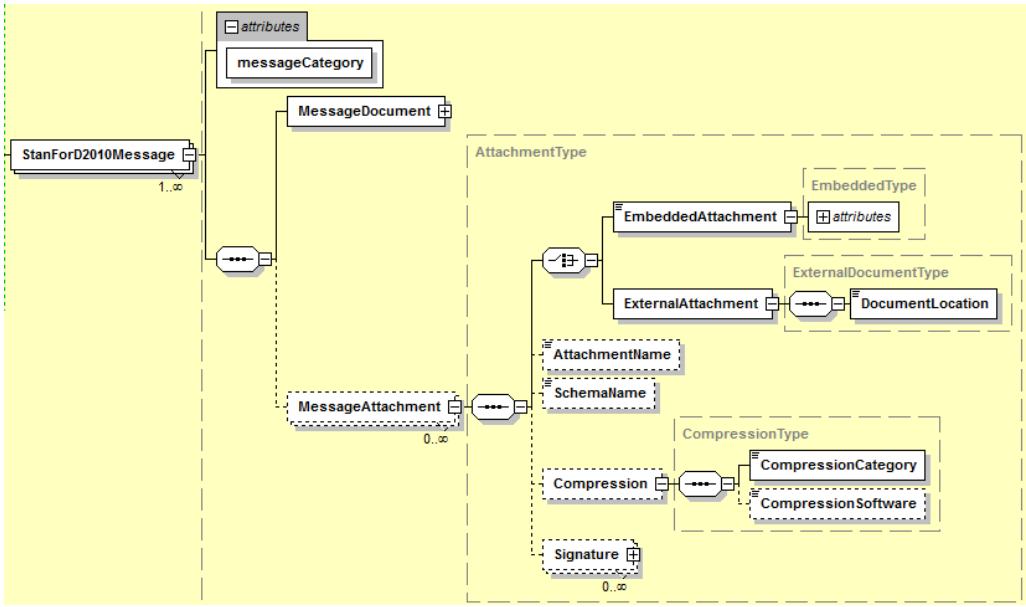
Butt end profile functions and tables according to timber measurement act (Poikela 2007) included. barkFunctionCategory is set to 'None'

§ 11 Compression and envelopes

Based on discussion with Antti Korkiakosko and Antti Yli-Fossi (2013-03-05)

- Do we plan to have ENVz files (zipped ENV) like MOMz and HPRz?
  If we are going to use ENVz the schema enables double compression since the

"EmbeddedDocument" can be compressed (zipped). When trying to compress the already compressed content, the outcome is not significantly smaller than original (actually, in some cases it can be larger). The schema also allows that there can be both zipped EmbeddedDocuments and not zipped EmbeddedDocuments in the same ENV. A suggestion is that the "Compression" element is to be located under "External document" so that the content of the ENV (without external documents) is compressable.
Present structure for attaching non StanForD files in MessageAttachment is illustrated below:



- There is also a question regarding element "DocumentLocation" under "ExternalAttachment" since it allows web-addresses. This may cause internet security issues and unpredictable big data transfer bills. Let's take an example with a machine where there is only GPRS modem (or satellite receiver in Russia) and there is a 15MB map file linked to the document as an external document. The operator starts a site using that ENV. Should the machine create internet connection and start to down-load the file (takes a lot of time and can be really expensive via satellite, but is the easiest way for the operator)? Automatically? How is this planned to work, or is it machine manufacturers' own decision how they should handle external documents in their systems?

From a development standpoint the compression inside the env file is a bit more complicated to implement, and not supporting envz makes an exception to a rule general to all other documents, so deliberately not supporting envz gives more work and an artificial limitation.

Another point is that if we have several pin or spi files inside an env file, compressing them individually and then storing those as Base64 is not as efficient as just letting them be plain text and compressing the whole env file, since in the latter case the dictionary created by the DEFLATE algorithm is made from everything contained in the env file and thus every "<ApplicationVersionModified>" is just replaced with one reference to the one dictionary.

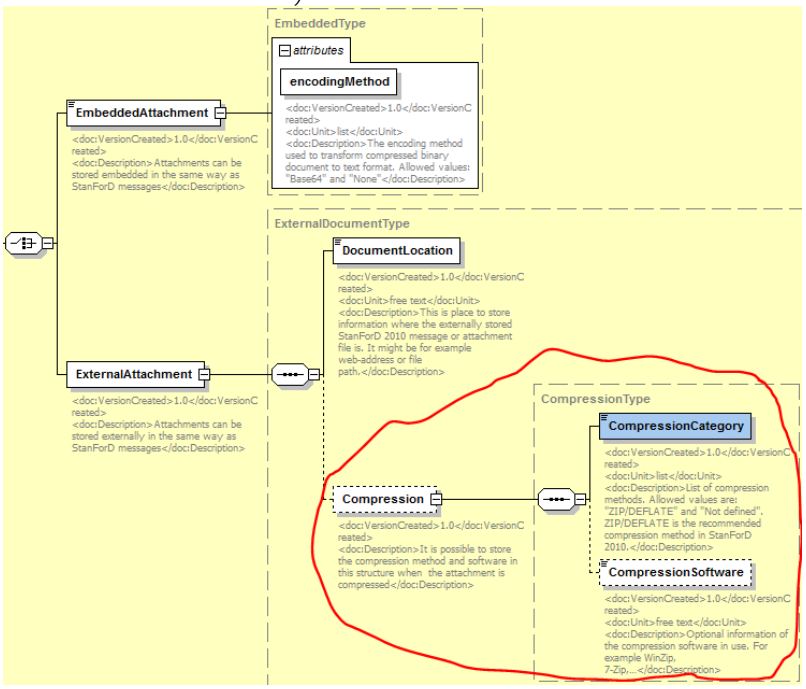
Third point regarding this is that if we compress the internal EmbeddedDocuments, they have to be Base64 encoded. This along with the compression itself makes it difficult to at least try to see what the internal files look like in case there's some ad hoc need to do that e.g. on the field, as opposed to an envz file that can just be unzipped and then inspected with any text editor, given that the encoding method in the embedded documents is "none".

Observe that Base64 encoding is to be used when including binary files in an envelope as illustrated in the example below where a geographical instruction including an ogi and a jpg is merged into one single envelope:

```
<StanForD2010Message messageCategory="ogi">
  <MessageDocument>
    <EmbeddedDocument encodingMethod="None">&lt;?xml version="1.0" encoding="UTF-8"?&gt;…
    <DocumentName>1033.ghd</DocumentName>
  <MessageDocument>
  <MessageAttachment>
    <EmbeddedDocument encodingMethod="Base64???">/9j/4AAQSkZJRgABAQEAYABgAAD/2wBD…
    <DocumentName>AerialPhoto.jpg</DocumentName>
  <MessageAttachment>
```

It was thus decided that:

- Individual messages included in an envelope are not to be compressed. This means that element Compression is only to be used for external files (ExternalAttachment and ExternalDocument) as described below.



EncodingMethod will remain for EmbeddedAttachment and EmbeddedDocument in order to handle binary files.

- The complete envelope is to be compressed into an ENVz-file if compression is needed.

§ 12 Updated Swedish reporting recommendations

John presented the updated Swedish recommendations which are included appendix 3.

A conclusion drawn from the discussion was that it must be possible to generate object oriented hpr-files where only data from one single object is included.

§ 13 A new instruction film: "How to do control measuring?"

Björn Hannrup presented a preliminary instruction film created by Maria Nordström, Skogforsk. The film will be finished and made available for everyone by May.

§ 14 Skogforsk projects:

Development of calibration methods

Björn Hannrup presented a calibration project that will be carried out by Skogforsk starting this spring. The project consists of the following parts:

1. Data collection for calibration
2. Success factors for improved measurements and effects of the quality assurance system
3. Operator support system for analysis of measurements quality and need of calibration
4. Guide lines for control measurements/ calibration

Calibration of MTH volumes

Johan Möller presented the first work done in this project.

§ 15 Development of SilviA for StanForD 2010

John informed the meeting that SilviA 7 was released in January 2013. A new version with bug fixes available right now!

Contact regarding SilviA7 is infoforest.se@logica.com

§ 16 Information material regarding StanForD2010

Johan presented the printed brochures that are also available at www.skogforsk.se/stanford2010 as pdf-files. Presently in Swedish and English but soon also in German thanks to translation work done by KWF.

§ 17 SDC testing hpr

Status today:

Hpr:

- Six harvesters send hpr to SDC. Same production is sent in pri- or prd-formats.
- A separate version of SDC Sender for StanForD2010-messages has been created.

- A good test material to check the mapping between the two formats and our analyses of hpr.

PRI:

- Test of converting pri to hpr and compare the results from reading of the two formats.
- There will be a number of files which can´t be converted. Most common problem is wrong stem numbering in partial reported pri (some older software versions).

Test version available for some customers in the beginning of May

Hpr:

- Automatic reading and presentation of the results from a small number of harvesters. The results will be available for some of our customers in a separate test environment.
- The harvesters in the test still need to send both hpr and pri/prd from same objects.

Release in the end of May

Timing depends on the test results.

Hpr from a limited number of harvesters from the test will be moved to the new production environment.

The reading of pri/prd in the old environment will be stopped for these machines.

Gradually more and more pri machines will be moved to the new environment. The definite plan for this is not decided.

Hpr from new application versions will first be checked in the test environment.

The change from prd to pri

Today over 2/3 (~1050) of the harvesters send pri instead of prd. The plan is to stop the receiving and reading of prd from 2013-06-01. SDC will not close prd before we are ready for hpr

§ 16 Discussion about updating the old standard and length of change period.

Decided that unless there are proposals for next meeting we will stop including this issue on the agenda

§ 17 Other issues

No other issues.

§18 Next meetings

2013-08-29	Phone meeting at 09:00 CET. Issues: WorkOrderNumber, LocationDefinitions in foi
2013-11-06—07	Regular StanForD meeting in Uppsala .

§ 19 End of meeting

Johan thanked everyone for their participation.

Juha-Antti Sorsa
John Arlinger

6 March 2013

Proposal new basic designs

Design of ClassifiedProductDefinition

The implementation of classified product definition structure in StanForD 2010 schemas is not optimal if we want to support reusability and avoid redundancy in data structures. The following table gathers all types that define classified product specific information and in which messages they are used. The first three types are defined in StanForD2010CommonDefinitions file. All the other types are message specific. We can almost immediately see that same information is defined in many places. For example the common definition type is nearly identical and it can be found in three different files: one in StanForD2010CommonDefinitions file and the others in fdi and pin messages.

Type/Message	Fpr	fdi	hpr	hqc	pin	thp
CommonProductDefinitionType	X		X	X		X
CommonProductDefinitionMachineType	X		X	X		X
ClassifiedProductDefinitionMachineType				X		X
ClassifiedProductDefinitionMachineHarvestedProductionType			X			
CommonProductDefinitionProductInstructionType					X	
ClassifiedProductDefinitionProductInstructionType					X	
CommonProductDefinitionForwardingDeliveryInstructionType		X				
ClassifiedProductDefinitionForwardingDeliveryInstructionType		X				

This means that if we want to for example update CommonProductDefinitionType in Stan-ForD2010CommonDefinitions file we have to update also types in both pin and fdi messages. This kind of data redundancy is not desirable. This is not only schema management problem be-cause applications that are processing messages have to be updated also in many places. Therefore we suggest that we reorganise product structures in schemas so that redundant parts can be removed as much as possible.

Important thing is to analyse what are actually the differences between classified product definitions in separate messages? There is naturally common part that is equal in each message. The distinguishing parts (elements and structures behind them) have been presented in following table.

Element/Message	fpr	fdi	hpr	hqc	pin	thp
ProductUserID	X		X	X		X
SpeciesGroupUserID		X			X	

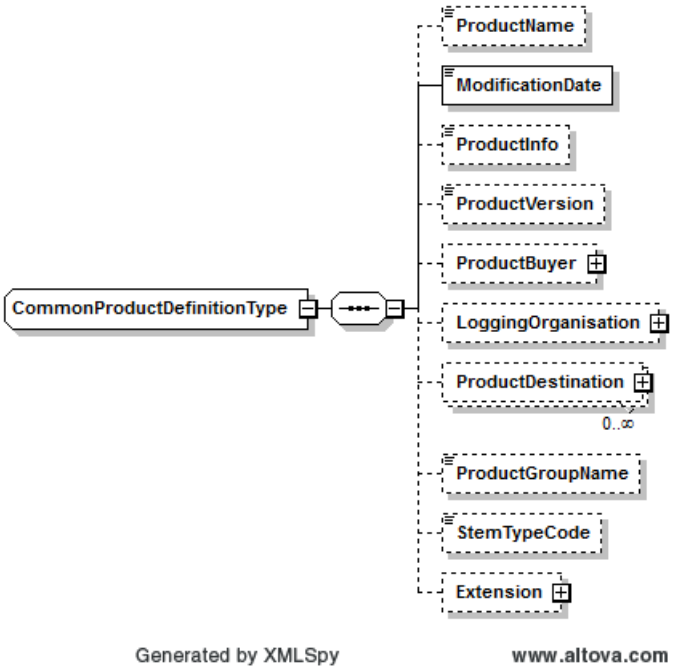
SpeciesGroupKey	X		X	X		X
DiameterDefinition			X	X	X	X
LengthDefinition			X	X	X	X
PriceDefinition			X	X	X	X
ProductCreationDate		X	X		X	
ProductPresentationOrder		X	X		X	
ManualFreeBuck			X		X	
TopLogFreeBuck			X		X	
ProductCondition			X		X	
CuttingWindowDefinition			X		X	
PermittedGradesDefinition			X		X	
LengthDistributionDefinition			X		X	
LimitationDefinition			X		X	
ProductMatrixes			X		X	
LongLogButtHeight			X		X	

In table we can see that identification information varies quite a much between messages. However there is two set of elements that are included in multiple messages. Elements inside dark red rectangle are in four messages and ones inside light red rectangles are in two messages. We can use this information to provide reusable structures that are defined only once in Stan-ForD2010CommonDefinitions file.

Therefore we suggest that we reorganise product structures in schemas so that most of the redun-dant parts can be removed.

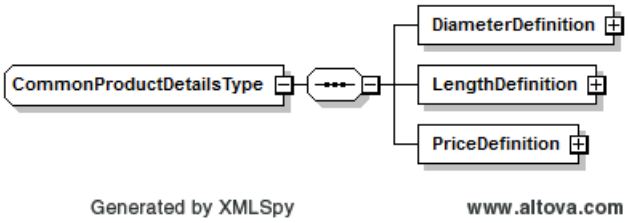
The suggestion is following:

1) We provide one type definition where we gather all information that is common for every classified product. That kind of type we have already in StanForD2010CommonDefinitions that is CommonProductDefinitionType.

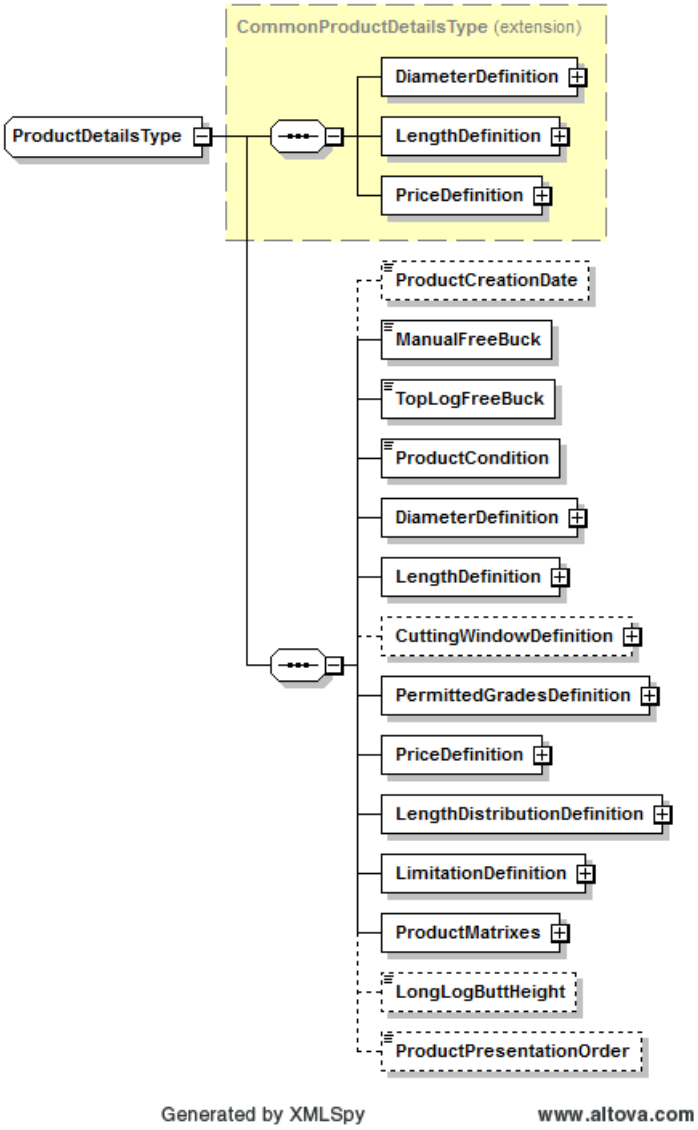


2) Then we provide subtypes for each message that has classified product structure. All those subtypes inherit CommonProductDefinitionType and additionally they define few ele-ments that are varying. Elements that have to be defined in each message are the ones that have no color background in table above. Additionally we add one mandatory element ProductDetails for those subtypes (messages) that contain elements that are colored in table above. Those new element structures are described in 3) and 4). These subtypes are message specific and therefore they are defined in schema files of corresponding messages.

3) We include in StanForD2010CommonDefinitions file a totally new type CommonPro-ductDetailsType that has three elements the ones that are dark red in table

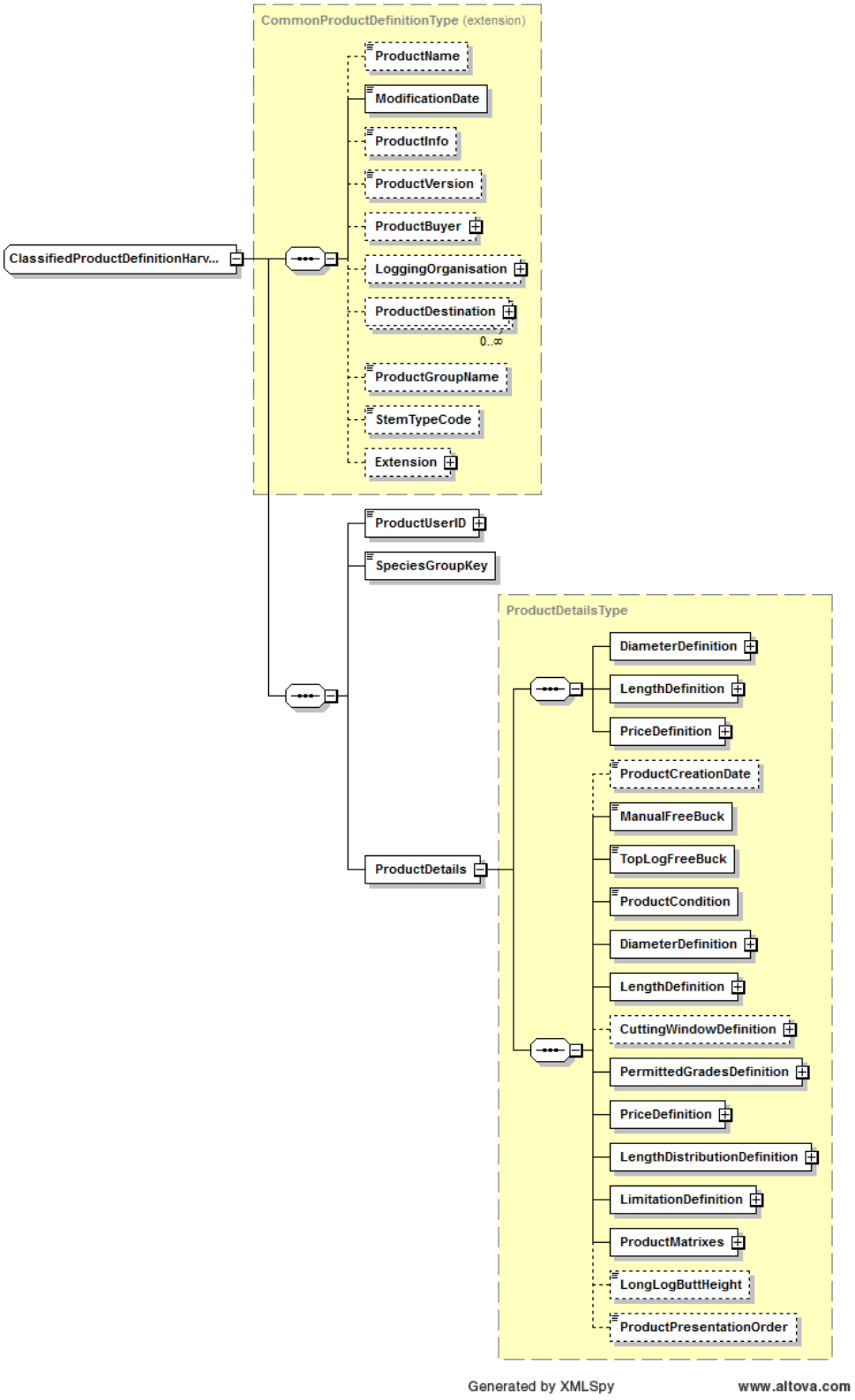


4) Then we include in StanForD2010CommonDefinitions file a new subtype ProductDe-tailsType that inherit CommonProductDetailsType.



Types defined in 3) and 4) are used in message specific classified product structures as needed. For example the ClassifiedProductDefintionHarvestedProductionType in hpr

schema would be following:



Design of SpeciesGroupDefinition

When we analyzed the actual differences between species group definitions in separate messages we can see the result in the following table.

Element/Message	fdi	fpr	hpr	Hqc	thp	mom	spi
SpeciesGroupModificationDate	X	X	X	X	X	X	X
SpeciesGroupUserID	X	X	X	X	X	X	X
SpeciesGroupName	X	X	X	X	X	X	X
SpeciesGroupInfo	X	X	X	X	X	X	X
SpeciesGroupVersion	X	X	X	X	X	X	X
StemTypeDefinition	X	X	X	X	X	X	X
LoggingOrganisation	X	X	X	X	X	X	X
Extension	X	X	X	X	X	X	X
DBHHeight			X	X	X	X	X
Grades			X	X	X	X	X
BarkFunction			X	X	X	X	X
SoundKnotFunction			X	X	X	X	X
ButtEndProfileExtrapolation			X	X	X	X	X
EstonianVolumeParameters			X	X	X	X	X
SpeciesGroupPresentationOrder			X	X	X	X	X
SpeciesGroupKey		X	X	X	X	X	
SpeciesDefinition							X

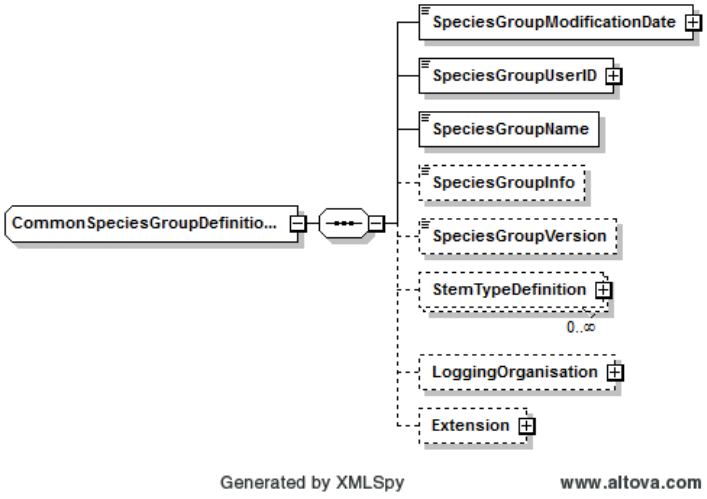
We can see that there are three set of elements that are included in multiple messages. We can use this information to provide reusable structures that are defined only once in StanForD2010CommonDefinitions file.

Therefore we suggest that we reorganize species group structures in schemas so that most of the redundant parts can be removed.

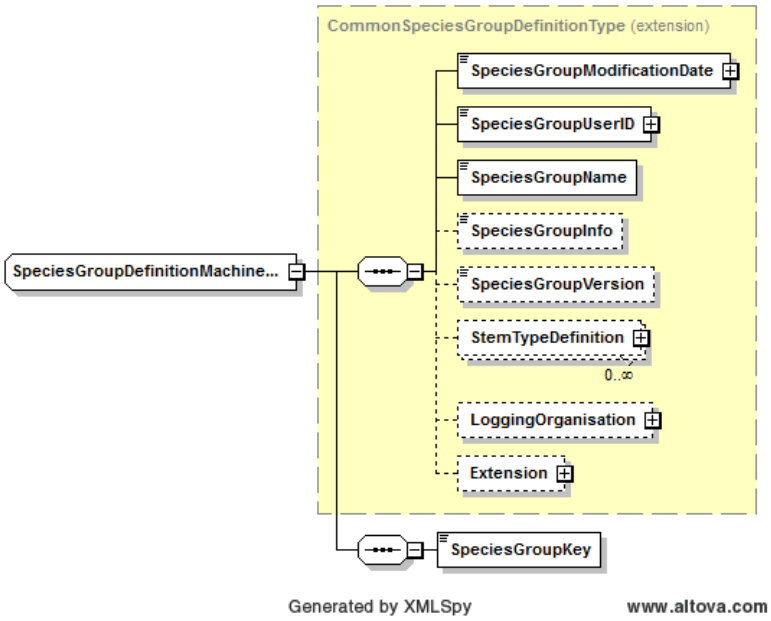
The suggestion is following:

1. We provide one type definition where we gather all information that is common for every classified product (green rectangle in table). We name this type

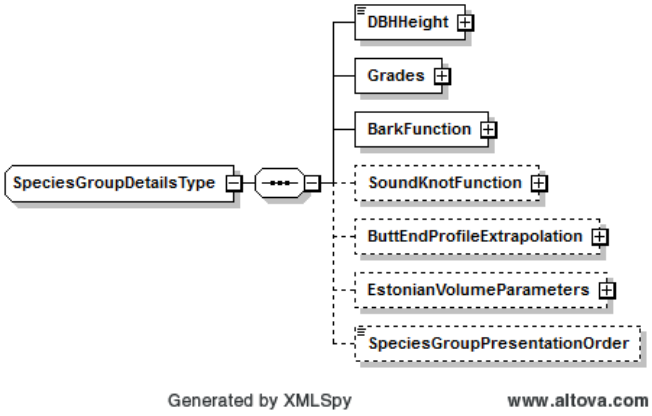
`CommonSpeciesGroupDefinitionType`.



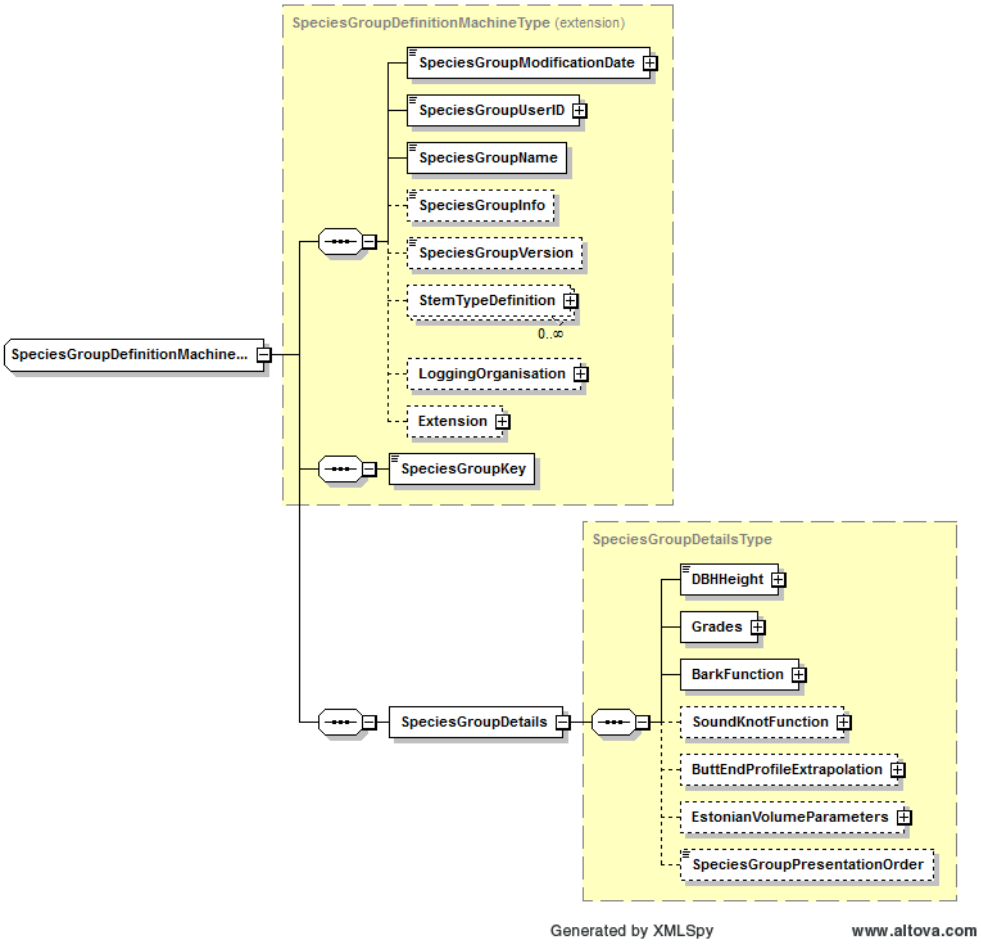
2. Then we provide a new type SpeciesGroupDefinitionMachineType that extends CommonSpeciesGroupDefintionType with SpeciesGroupKey



3. We include a totally new type `SpeciesGroupDetailsType` that contain elements that are colored light blue in table.



4. And finally we provide a new type SpeciesGroupDefinitionMachineWithDetailsType that gather all information into one type



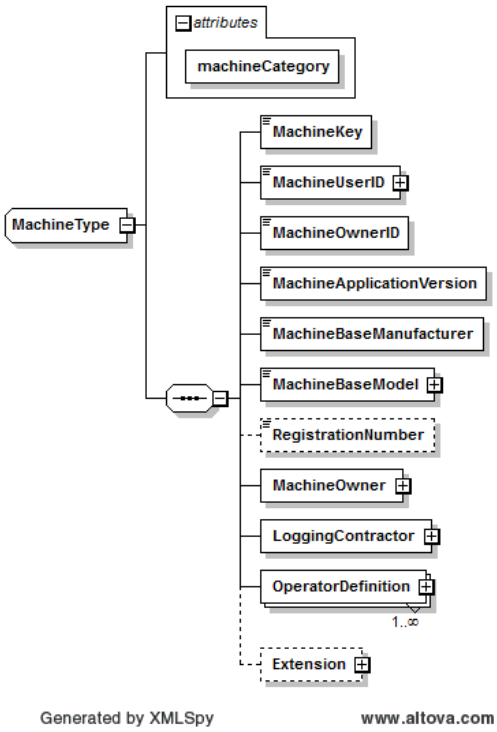
Using these new types we fulfill almost all message specific needs for species group definition. Only in "spi" message we have to implement message specific definition because there is need for SpeciesDefinition structure.

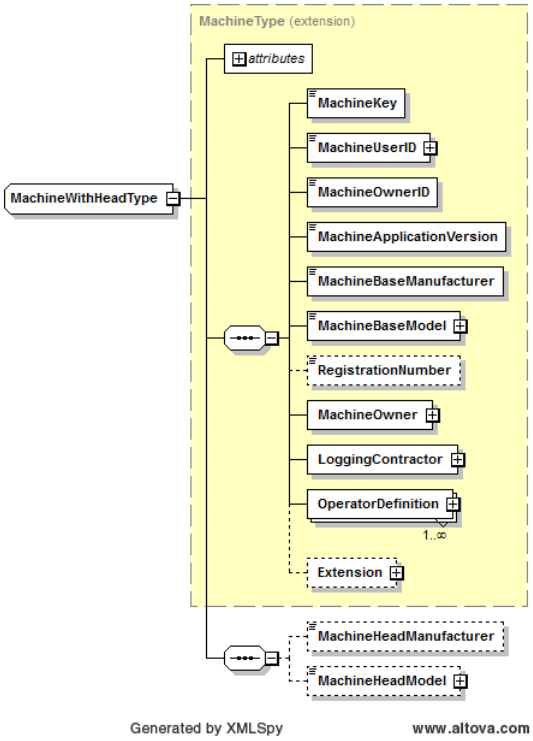
Design of Machine structure

We have also analysed the Machine structure in messages in the same way as SpeciesGroupDefinition and ClassifiedProductDefinition. In this case we have found two sets of elements that are common for messages (table below)

Element/Message	fqc	fpr	hpr	hqc	thp	mom	ogr
machineCategory	X	X	X	X	X	X	X
MachineKey	X	X	X	X	X	X	X
MachineUserID	X	X	X	X	X	X	X
MachineOwnerID	X	X	X	X	X	X	X
MachineApplicationVersion	X	X	X	X	X	X	X
MachineBaseManufacturer	X	X	X	X	X	X	X
MachineBaseModel	X	X	X	X	X	X	X
RegistrationNumber	X	X	X	X	X	X	X
MachineOwner	X	X	X	X	X	X	X
LoggingContractor	X	X	X	X	X	X	X
OperatorDefinition	X	X	X	X	X	X	X
Extension	X	X	X	X	X	X	X
MachineHeadManufacturer			X	X	X	X	X
MachineHeadModel			X	X	X	X	X

In this case we have provided two types: MachineType and MachineWithHeadType that are presented below. These types are used in messages as a base type for message specific version of machine structure.



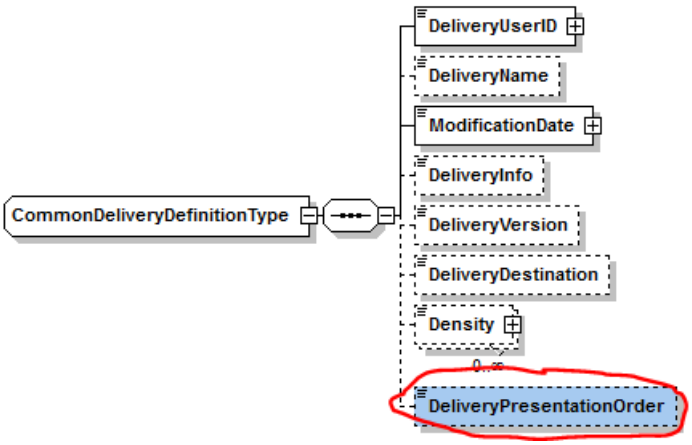


Based on suggestion from Teemu (2013-04-08)

There is not common definition for delivery at the moment. We think that a common base class for delivery should be introduced because almost the same information is now included in two messages :

- DeliveryDefinitionType in fpr
- DeliveryDefinitionForwardingDeliveryInstructionType in fdi

A suggested new CommonDeliveryDefinitionType to be used in both fpr and fdi:



John Arlinger
Lars Henriksson

19 January 2012

Mapping of StanForD Id elements

It was decided at the November (2010-11-10) meeting that John Arlinger is to write a proposal for how to map and convert between the two standard versions with a focus on relevant identities.

Conversions may according to the manufacturers occur in two directions:

- StanForD2010 →StanForD
  For example, new StanForD2010 instructions (oin/pin/spi) are sent to harvester and old drf-file is reported from machine.
- StanForD →StanForD2010
  For example, old StanForD apt-file is sent to harvester and new hpr-file reported from harvester.

Background

Three major issues concerning mapping was discussed at the telephone meeting 2010-12-10:

- Is a one-to-one relationship really needed? Do we need a couple of new elements?
- Why is not UserId elements included in the mapping table?
- Is apt-file to be converted into oin/pin/spi before being used in a harvester adopted to StanForD2010?

ONE-TO-ONE RELATIONSHIP

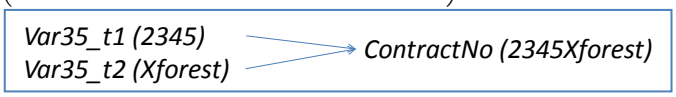
Conversions will become significantly more complex if we do not have a one-to-one relationship between StanForD variables and StanForD2010 elements.

A simple example of this problem (having a one-to-many relationship) can be illustrated by comparing variable 35 (contract number) and element ContractNo. Var35 consists of two types with no clear difference which means that we may have a on-to-two relationship.

When StanForD2010 structure is converted to StanForD (for ex. oin → drf) we could copy the same data to both variables:

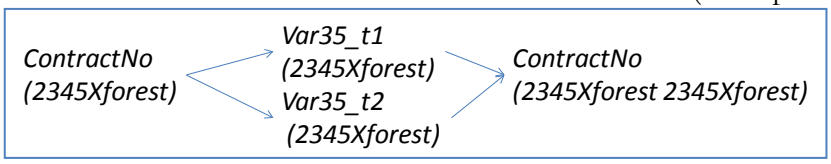
ContractNo(2345Xforest) → *Var35_t1 (2345Xforest)*
ContractNo(2345Xforest) → *Var35_t2 (2345Xforest)*

When a StanForD structure is converted to StanForD2010 (for ex. apt → hpr) a solution could be that we merge the data for these two variables into a single element ("2345"+"Xforest"="2345Xforest"):

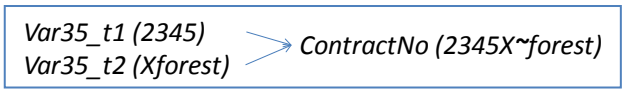


One problem with this solution is that we do not know whether the whole string came from one or both (var35_t1 and var35_t2) StanForD variables.

Another problem with one-to-many relationships is that we may end up with a mess if the conversion is done more than once as illustrated below (oin→pri→hpr):



One way to handle this kind of one-to-many relationship could be to use a separation character like ~ between the different StanForD variables included in one element as illustrated below:



However this would mean that we must have a separate parser for certain elements when they are based on converted structures.

Consequently Skogforsk do prefer staying with some kind of strict one-to-one relationship.

USERIDS

The UserIDs have a dual purpose in StanForD2010 which does not exist in StanForD:

- Forest companies or other users of data use *UserId* for identification purposes. *UserIds* may be used for harvesting objects, products, tree species, operators, transport objects and machines.
- UserIds are also used to link for example products in oin and pin together which means that they in some cases (for example oin) must be unique. This purpose is further described below.

The basic idea is that only one instance of a specific *UserId* should be available to use in a machine at a time. An old product or species group is for example to be replaced if a new product or species group with the same *UserId* is received by a machine. This means that for example only one instance of *ProductUserId* can be selected and used at a given time

The following figures (1-2) illustrates an example of how *UserIds*, *ModificationDates* and *Keys* are handled when updating and using object instructions. Observe that both *UserId* and *ModificationDate* are to be considered when updating the list of objects. An existing object is not to be replaced by a new object sent to the harvester if they have identical *UserId* and

ModificationDate. The operator always must have the possibility to deny the replacement of an existing object as well as an update of the active object presently in use.

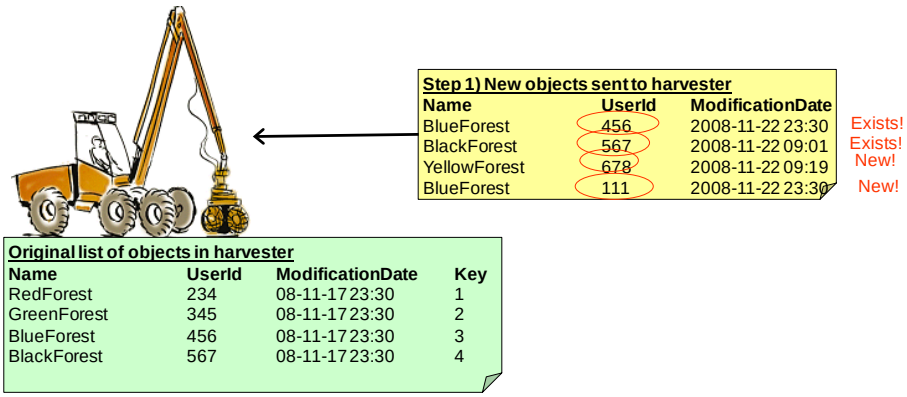


Figure 1. Five new objects (oin) are sent to a harvester. The UserId of the first (456) and the second product (567) already exists among the object while the third and forth objects are new.

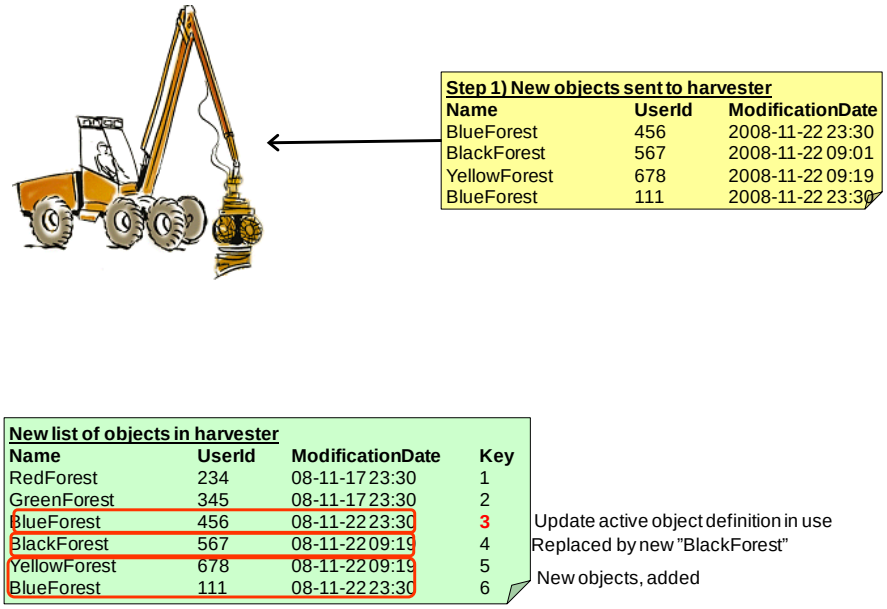


Figure 2. The list of objects is updated. Observe that a second object named BlueForest now exist since the ObjectUserIds are different .

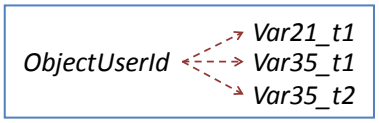
Unique identities in the old StanForD are today registered in different variables depending on market and companies involved. Sometimes a single variable is used sometimes several variables are used together. The following examples illustrate the fact that this is quite complex today:

- The unique id of a harvesting object (same as ObjectUserId) is often stored in var21_t1, var35_t1 or var35_t2.
  Södra skogsägarna use a combination of var21_t1 and var35_t1
  Sveaskog use var35_t2 (most common way in Sweden)

Finland (as far as I have understood) use var35_t1

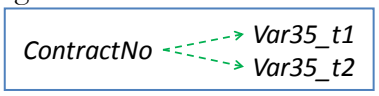
- ProductUserId stored in var32_t2, var121_t2, var121_t3 or var121_t6 in present StanForD

The following figure illustrates that ObjectUserId would have to be mapped against a number of different StanForD variables (example oin → drf):



Observe that we would have to have different rules for mapping ObjectUserId on different markets and for different forest companies.

To make the situation even more complex there are other elements that would be mapped against the same variables as for example ContractNo:



A rule would be needed to define whether ObjectUserId should over ride ContractNo or not.

Suggested solution

After some discussions (phone meeting 2011-01-21 and 2012-01-31) it was decided that we make the exact mapping settable in the machine according to the following table:

Elements	Alternative mapping variables set in machine
ObjectUserId	Var21_t1 OR var35_t1 OR var35_t2
ContractNumber	Var35_t1 OR var35_t2 Or no mapping
RealEstateIDObject	Var21_t1 OR var35_t1 OR var35_t2 Or no mapping
SubObjectUserId	Var21_t1 Or Var21_t2 Or no mapping
ProductUserId	Var120_t1 AND Var121_t1 AND Var121_t2 AND var121_t3 AND var121_t5
ProductVersion (old ProductIdInfo)	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductInfo	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductDestination.BusinessID	var32_t2
SpeciesGroupUserId	var120_t1 AND var120_t3
SpeciesGroupName	var120_t1 (var120_t2 in case of stm/ktr)
SpeciesGroupInfo	Order of species in file (eg stem code 1, var266 in pri-file)

Red and underlined means default mapping in Sweden. Necessary to follow if file is sent to SDC

Observe that ProductUserId and SpeciesGroupUserId are never to be converted from StanForD2010 to StanForD (for example from hpr to prd).

In this case it is not necessary to add any new id elements.

This solution may in theory give as a problem if for example var21_t1, var35_t1 or var35_t2 are empty (no ObjectUserId created) or if duplicate UserIds (for example ProductUserId) are created within a message. The following rule must be implemented when doing conversions:

- Conversion of apt to oin/pin/spi not allowed if UserIds in file are not unique (for example ProducUserId).

Converting apt-file to StanForD2010

Is an apt-file to be converted into oin/pin/spi before being used in a StanForD2010-harvester or not? How is it going to be used if we generate StanForD2010 files?

It is suggested that an apt-file is normally not converted to oin/pin/spi before being imported to ”product database”/”bucking application” in harvesters adapted to StanForD2010. All “Keys” are then set when apt is imported and products created in product database.

Below is an example based on the suggestions above (mapping set in machine):

Apt →

Var	Data
v21_t1	XForest
v35_t1	123417
v35_t2	xY
v120_t1	Pine
V121_t1	P_Saw
v121_t2	11_Q1
v121_t3	Mill_1Y_ Special
v32_t2	Mill_1Y

Hpr →

Element	Data
ObjUserId	XForest
RealEstateIdObject	123417
ContractNo	xY
SpeciesGroupName	Pine
ProductName	P_Saw
ProductInfo	11_Q1
ProductVersion	Mill_1Y_special
ProdDest. BusinessID	Mill_1Y
ProdUserId	PineP_Saw 11_Q1 Mill_1Y_Special
ObjectKey	1
ProdKey	1

Pri

Var	Data
v21_t1	XForest
v35_t1	123417
v35_t2	xY
v120_t1	Pine
V121_t1	P_Saw
v121_t2	11_Q1
v121_t3	Mill_1Y_ Special
v32_t2	Mill_1Y

This alternative means that a pin-product is not to be used for updating apt-products!

Mapping table

The following table describes a suggested mapping for important id variables:

Structure (Def)	New element	Old variable
MachineDefinition	MachineOwnerID	Var003_t1
MachineDefinition	MachineUserId	var003_t2
MachineDefinition	MachineCategory	Var003_t3
MachineDefinition	MachienBaseManufacturer	Var003_t5
MachineDefinition	MachienBaseModel	Var003_t6
MachineDefinition	MachienHeadManufacturer	Var003_t7
MachineDefinition	MachienHeadModel	Var003_t8
MessageHeader	ApplicationVersionCreation	var005_t1
MessageHeader	ApplicationVersionCreation	var005_t2

Structure (Def)	New element	Old variable
MessageHeader	ApplicationVersionModification	var005_t4 (caliper id in ktr)
MessageHeader	CountryCode	Var006_t1
MessageHeader	CreationDate	var012_t4
ProductDefinition	ProductCreationDate	Var013_t4
ObjectDefinition	StartDate	Var016_t4
ObjectDefinition	EndDate	Var017_t4
ObjectDefinition	ObjectUserId	Var021_t1 OR var35_t1 OR var35_t2
ObjectDefnition	RealEstateIdObject	Var021_t1 OR var35_t1 OR var35_t2 Or no mapping
SubObjectDef	SubObjectName	Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
SubObjectDef	SubObjectUserId	Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
SubObjectDef	RealEstateIdSubObject	Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
ObjectDefinition	ForestCertification	Var021_t5
ObjectDefinition	LoggingFormCode	Var023_t1
ObjectDefinition	LoggingFormDescription	Var023_t2
ObjectDefinition	ObjectArea	Var023_t3
ObjectDefinition	LoggingOrganisation.BusinessId	var031_t1
ObjectDefinition	LoggingOrganisation.BusinessName	var031_t2
ObjectDefinition	LoggingOrganisation.District	var031_t3
ObjectDefinition	LoggingOrganisation.Team	var031_t4
ObjectDefinition	LoggingOrganization.LastName	var031_t6
ObjectDefinition	LoggingOrganization.Address	var031_t7
ObjectDefinition	LoggingOrganization.Email	var031_t8
ObjectDefinition	LoggingOrganization.Phone	var031_t9
ProductDefinition	ProductBuyer.BusinessID	var032_t1
ProductDefinition	ProductDestination.BusinessID	var032_t2
ObjectDefinition	ForestOwner.BusinessName	Var033_t1
ObjectDefinition	ForestOwner.BusinessID	Var033_t2
ObjectDefinition	ForestOwner.Name	Var033_t3
ObjectDefinition	ForestOwner.Adress	Var033_t4
ObjectDefinition	ForestOwner.Email	Var033_t5
ObjectDefinition	ForestOwner.Phone	Var033_t6
MachineDefinition	MachineOwner.BussinesName	Var034_t1
MachineDefinition	LoggingContractor.BussinesName	Var034_t1
MachineDefinition	MachineOwner.BussinesID	Var034_t2
MachineDefinition	LoggingContractor.BussinesID	Var034_t2
MachineDefinition	MachineOwner.LastName	Var034_t3
MachineDefinition	LoggingContractor.LastName	Var034_t3
MachineDefinition	LoggingContractor.Adress	Var034_t4
MachineDefinition	LoggingContractor.Email	Var034_t5
MachineDefinition	LoggingContractor.Phone	Var034_t6
ObjectDefinition	ContractNo	Var35_t1 OR var35_t2 Or no mapping
Stem	SpeciesGroupKey	var110_t0 (stm/ktr)
SpeciesGroupDefinition	SpeciesGroupUserId	var120_t1 AND var120_t3
SpeciesGroupDefinition	SpeciesGroupName	var120_t1
SpeciesGroupDefinition	SpeciesGroupName	var120_t2 (stm/ktr)
SpeciesGroupDefinition	SpeciesGroupVersion	var120_t3
SpeciesGroupDefinition	SpeciesGroupInfo	Order of species in file (eg stem code 1,

Structure (Def)	New element	Old variable
		var266 in pri-file))
ProductDefinition	ProductName	Var121_t1
ProductDefinition	ProductUserId	Var120_t1 AND Var121_t1 AND Var121_t2 AND var121_t3 AND var121_t5
ProductDefinition	ProductVersion	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductDefinition	ProductInfo	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductDefinition	ProductCreationDate.	Var12_t4
ProductDefinition	ProductGroupName	var127_t1
DiameterDef	DiameterClassLowerLimit	var131_t1
DiameterDef	DiameterClassName	var131_t2
SpeciesGroupDefinition	GradeName	var143_t1
ObjectDefinition	ObjectTextToMachine	Var200_t2
OperatorDefinition	OperatorUserId	Var212_t1
DeliveryDefinition	DeliveryInfo	Var441_t2

Red and underlined means default mapping in Sweden. Necessary to follow if file is sent to SDC

SWEDISH TRANSLATIONS

Below is a table with Skogforsk and SDC recommendations regarding Swedish translations of certain important elements.

Structure (Def)	New element	Swedish GUI name	Comments	Old var
MachineDefinition	MachineKey	Maskinnyckel		
MachineDefinition	MachineOwnerID	Ägarens maskin id	Id defined by machine owner	Var003_t1
MachineDefinition	MachineUserId	Maskin id enl	Id defined by logging organisation	var003_t2
MachineDefinition	MachineCategory	Maskintyp		Var003_t3
MachineDefinition	MachienBase-Manufacturer	Basmaskin, märke		Var003_t5
MachineDefinition	MachienBase-Model	Basmaskin, modell		Var003_t6
MachineDefinition	MachienHead-Manufacturer	Aggregat, märke		Var003_t7
MachineDefinition	MachienHeadModel	Aggregat, modell		Var003_t8
MessageHeader	ApplicationVersion-Creation	Version när skapad???		var005_t1
MessageHeader	ApplicationVersion-Modification	Version när ändrad		var005_t4
MessageHeader	CountryCode	Landskod		Var006_t1
MessageHeader	CreationDate	Skapad:		var012_t4
ProductDefinition	ProductCreationDate	Ändrad:		Var013_t4
ObjectDefinition	StartDate	Startdatum		Var016_t4
ObjectDefinition	EndDate	Slutdatum		Var017_t4
ObjectDefinition	ObjectName	Objektnamn		
ObjectDefinition	ObjectUserId	Objekt id	Id defined by logging organization	Var021_t1 OR var35_t1 OR var35_t2
ObjectDefnition	RealEstateIdObject	Fastighets nr		Var021_t1 OR var35_t1

Structure (Def)	New element	Swedish GUI name	Comments	Old var
				OR var35_t2 Or no mapping
SubObjectDef	SubObjectName	Del-objektnamn		Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
SubObjectDef	SubObjectUserId	Del-objekt id	Id defined by logging organisation	Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
SubObjectDef	RealEstateIdSubObject	Del-objekt, Fastighets nr		Var21_t2 OR Var21_t3 OR Var21_t4 Or no mapping
ObjectDefinition	ForestCertification	Skogs-certifiering		Var021_t5
ObjectDefinition	LoggingFormCode	Avverkningstyp		Var023_t1
ObjectDefinition	LoggingFormDescr.	Avverkningstyp, beskrivning		Var023_t2
ObjectDefinition	ObjectArea	Areal objekt		Var023_t3
ObjectDefinition	LoggingOrganisation. BusinessId	Organisation/skogsföretag, Id företag		var031_t1
ObjectDefinition	LoggingOrganisation. BusinessName	Organisation/skogsföretag, Företagsnamn		var031_t2
ObjectDefinition	LoggingOrganisation. District	Organisation/skogsföretag, Region		var031_t3
ObjectDefinition	LoggingOrganisation. Team	Organisation/skogsföretag, Distrikt		var031_t4
ObjectDefinition	LoggingOrganization. LastName	Organisation/skogsföretag, Efternamn		var031_t6
ObjectDefinition	LoggingOrganization. Address	Organisation/skogsföretag, Adress		var031_t7
ObjectDefinition	LoggingOrganization. Email	Organisation/skogsföretag, E-post /Mäjl		var031_t8
ObjectDefinition	LoggingOrganization. Phone	Organisation/skogsföretag, Tele		var031_t9
ProductDefinition	ProductBuyer. BusinessID	Köpare, Kod	Used by SDC	var032_t1
ProductDefinition	ProductDestination. BusinessID	Mottagningsplats, Kod	Used by SDC	var032_t2
ObjectDefinition	ForestOwner. BusinessName	Skogsägare, Företagsnamn		Var033_t1
ObjectDefinition	ForestOwner. BusinessID	Skogsägare, Kod		Var033_t2
MachineDefinition	MachineOwner. BussinesName	Maskinägare, företagsnamn		Var034_t1
MachineDefinition	LoggingContractor. BussinesName	Skogsentreprenör, företagsnamn		Var034_t1
MachineDefinition	MachineOwner. BussinesID	Maskinägare, Företagskod		Var034_t2
MachineDefinition	LoggingContractor. BussinesID	Skogsentreprenör, Företagskod		Var034_t2

Structure (Def)	New element	Swedish GUI name	Comments	Old var
ObjectDefinition	ContractNo	~~Kontraktsnr~~ Virkesordernr	Today normally virkesordernummer in Sweden, used by SDC.	Var35_t1 OR var35_t2 Or no mapping
ObjectDefinition	ContractCategory	Kontraktstyp		
SpeciesGroupDefinition	SpeciesGroupUserId	Trädslag id	Id defined by logging organisation	
SpeciesGroupDefinition	SpeciesGroupName	Trädslag, namn		var120_t1
SpeciesGroupDefinition	SpeciesGroupVersion	Trädslag, version		var120_t3
SpeciesGroupDefinition	SpeciesGroupInfo	Trädslagskod		Order of species in file (eg stem code 1, var266 in pri-file))
ProductDefinition	ProductName	Produktnamn		Var121_t1
ProductDefinition	ProductUserId	Produkt id	Id defined by logging organisation	
ProductDefinition	ProductVersion	Produktversion	Used by SDC.	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductDefinition	ProductInfo	Produktkod	Used by SDC. Today normally assortment code (SSTE)	Var121_t2 OR var121_t3 OR var121_t4 OR var121_t5 OR no mapping
ProductDefinition	ProductCreationDate.			Var12_t4
ProductDefinition	ProductGroupName	Produktgruppsnamn		var127_t1
OperatorDefinition	OperatorUserId	Förar id	Id defined by machine owner	Var212_t1

John Arlinger
Isabelle Bergkvist
Johan J Möller

2012-12-21

Reporting of StanForD2010 in forest machines

NEW SWEDISH REQUIREMENTS

Objectives

The main objective of this this document is to clearly specify the Swedish users requirements on machine manufacturers concerning how standardized data, according to StanForD, should be stored in harvesters and forwarders. This document replaces an older Swedish recommendation (Arlinger, J. & Möller, J. J., 2011, "Reporting of hpr-files").

Hopefully machine owners/users will be able to use the document as a basis for discussions with manufacturers regarding how to retrieve data from forest machines in an efficient way.

The document has been presented to both manufacturers and users during three different open meetings during the autumn of 2012 ("User group wood value", "TSG manufacturer seminar" and StanForD-meeting).

After one telephone meeting (2012-10-01) and some written communication (December 2012) a group consisting of the following user representatives has endorsed the requirements in this document:

Mats Larsson, Holmen Skog
Jonas Skyttmo, Skyttmos Skogsentreprenad
Jon Mattson, StoraEnso
Urban Nordmark och Jonas Gustafsson, Sveaskog
Patrik Andersson, Södra
Magnus Tjärnskog, SCA
Kalle Bramås, Mellanskog

Background

It is of great importance for that generation, storing and reporting of data:

- Is simple for the operators.
- Does not mean significant differences between different manufacturers. Receiving system must be able to handle reporting from all harvesters.

There were several discussions regarding how to store production data (hpr) during the development of StanForD2010. The manufacturers were in many cases hesitant about including too detailed standardized rules concerning this issue into StanForD2010

REQUIREMENTS ACCORDING TO THE STANDARD

The only requirement concerning reporting that has been agreed on is that all manufacturers must implement a function for reporting the total (aggregated) production for a harvesting object. Total reporting means that all harvested stems at an object are included in all reports (hpr-files) even if they have been reported previously. Total reporting is illustrated below in figure 1.

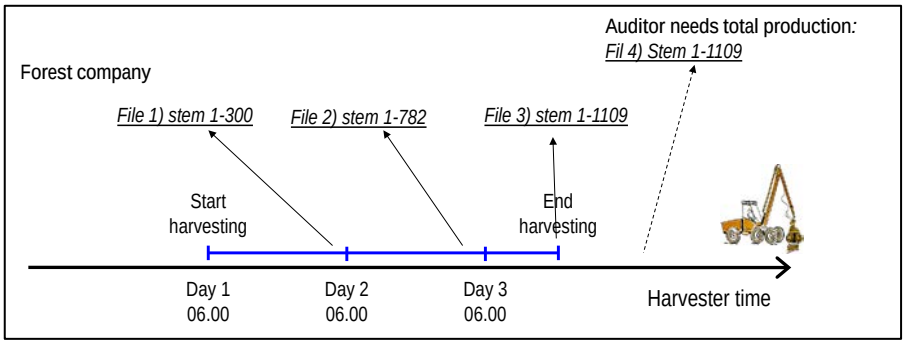


Figure 1. Illustration of StanForD requirement that it must be possible to report total production. In this case sending data to two separate organizations. This means that stem 1-300 will be included in all four files.

A clear disadvantage with total reporting is that the same stem data is often sent repeatedly, which means redundancy in the system.

Another standard requirement is that external (3rd party) applications used for transmitting files are responsible for deleting files from the machine system. It is also recommended that these external applications should have a setting for deleting files after transmitting files (StanForD-meeting 2012-04-18).

Swedish user requirements

PARTIAL REPORTING

Since Swedish users need the possibility to also do partial reporting of production data this separate Swedish document was written. A partial reporting mean that stems not previously having been reported are included in a new message. Since StanForD2010-files will have a relatively large size and require significant calculation capacity, partial reporting is deemed to be important. An example of partial reporting is illustrated in figure 2.

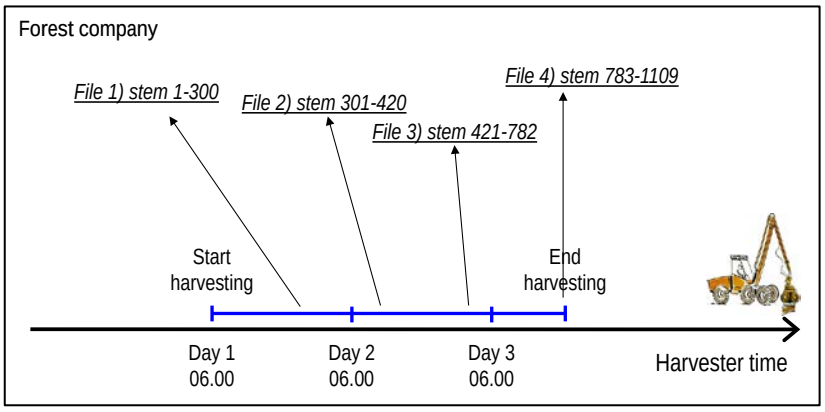


Figure 2. Illustration of partial reporting of four hpr-files.

The following Swedish requirement is therefore included:

- Functionality for partial reporting of production data must be implemented

AUTOMATIC CREATION OF STANFORD FILES

A majority of the Swedish forest machines today only report data through one single organization, normally SDC (figure 5). However, in some cases the reporting is done directly to a forest company system.

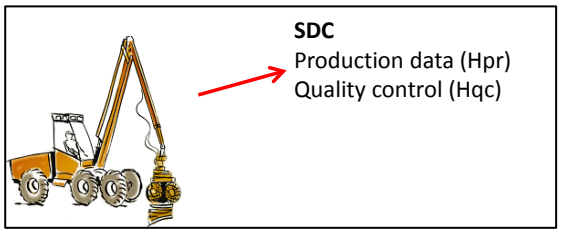


Figure 3. Illustration of the fact that certain data today normally is only sent to one single party, e.g. SDC or company specific system. This is the normal situation today.

An increasing interest has been noted recently, from several different actors that want to be able to send StanForD-data directly from the machines to their own system (figure 4). An example of this could be operational monitoring data used in different systems for different purposes as e.g. payment of operators, machine bench marking and comparisons of different harvesting objects.

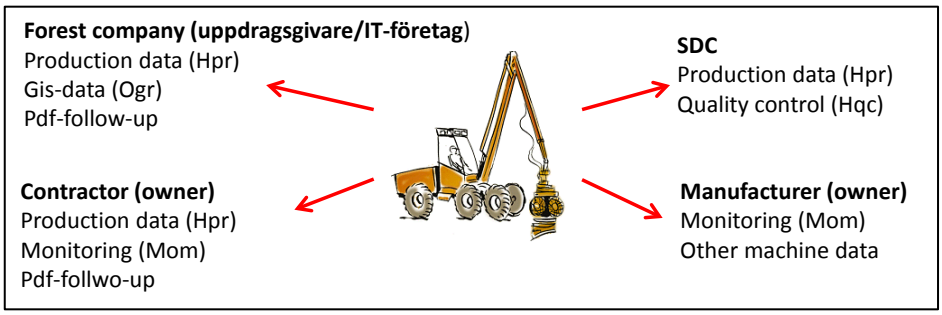


Figure 4. Several different parties will most certainly be interested in retrieving data directly from the machines in the near future. This may very likely cause conflicts where different transmission software try to get the same files.

The fact that we will most certainly see a number of new different "interested parties" requiring higher reporting frequency and simpler user interface for the operators mean that the following requirements has to be fulfilled for both harvesters and forwarders:

- No conflicts between different transmission software for sending files to a receiving system.
  Same type of report (mom, hpr,hqc etc) must be possible to store in different parallel folders. The folders must be constant over time or object unless the operator manually modifies these settings.

- The machine system must not automatically delete any StanForD files. The transmission software must be responsible for deleting old files from the machine system once the files have been transmitted.

- Automatic generation of reports must be implemented.
  This means that the operator should not have to press a button every time a report is to be created, for example once an hour or at the end of each harvesting object. Automatic report generation can be based on both events and time.

- A higher reporting frequency will be required, this means that we will go in the direction of on-line reporting. The minimum requirement is to be able to report automatically once an hour. Production data must also be possible to report automatically at the end of an harvesting object.

The requirements can be illustrated according to figure 5 and 6.

File types	Time-interval	Event	Reciever 1	Reciever 2	Reciever 3
Production (Hpr)	1 h	Obj finnished	../SDC/OutBox/	../Logica/hpr/	../SMF_Fleet/
Time oriented monitoring (Mom)	6 h	New operator	../SMF_Fleet/	../Manufacturer/	**entreprenor@ maskin.se**
Quality control (Hqc)	-	Control measuring	../SDC/OutBox/		
GIS (Ogr)	-	Obj finnished	../Logica/ogr/	**entreprenor@ maskin.se**	

X Always generate partial hpr reports

Figur 5. Simplified example of machine settings for automatic reporting. The objective is to illustrate the functionality, not the user interface.

File types	Time interval	Event	Reciever
Production (Hpr)	4 h	Object finnished	../SDC/OutBox/
Production (Hpr)	1 h	Object finnished	../Logica/hpr/
Production (Hpr)		Object finnished	**entreprenor@maskin.se**

Figur 6. Example of a more detailed setting per data receiver. This would be a better solution if different receivers have different needs regarding reporting frequency but it is probably a significantly more complex solution to implement and this functionality is therefore not required.

Possible events that could trigger an automatic generation of files are:

- End of harvesting at an object (applies to operational monitoring and production data, i.e. hpr, thp, fpr and mom)
- New operator (applies to operational monitoring and production data, i.e. hpr, thp, fpr and mom)
- Quality control measuring (applies to quality control data in hqc and fqc)
- Unloading of forwarder load (applies to forwarder production data in fpr)

Observe that the only event that must be implemented according to the Swedish users is "end of harvesting object" automatically triggering creation of production files.

All requirements

All requirements included in the document are compiled in the list below.

Requirements according to StanForD decisions 2011-12

- It must be possible to do total (aggregated) production reporting per harvesting object.
- External (3rd party) applications used for transmitting files are responsible for deleting files from the machine system.

Additional requirements from Swedish users:

- Functionality for partial reporting of production data must be implemented.
- Same types of reports (mom, hpr, hqc, fpr etc) must be possible to store in different parallel folders. The folders must be constant over time or object unless the operator manually modifies these settings if not changed manually by operators.
- The machine system must not automatically delete any StanForD files. The transmission software must be responsible for deleting old files from the machine system once the files have been transmitted.
- Automatic generation of reports must be implemented.
  This means that the operator should not have to press a button every time a report is to be created, for example once an hour or at the end of each harvesting object. Automatic report generation can be based on both events and time.
- It must be possible during automatic generation of reports to create individual files per object (hpr, thp and fpr). This means only one single ObectDefinition per production file is generated in the machines.
- A higher reporting frequency will be required, this means that we will go in the direction of on-line reporting. The minimum requirement is to be able to report automatically once an hour. Production data must also be possible to report automatically at the end of a harvesting object.

Manufacturers

The manufacturers have many different customers and markets to consider, it is therefore important that the Swedish users try to formulate a common recommendation. The Swedish users must respect the manufacturers need for a significant degree of freedom when it comes to developing the user interface. The users must also understand that the manufacturers must find suitable compromises between the requirements of different markets. The users must also give the manufacturers the freedom to further develop the functionality of their systems.

Future devlopment

A future possibility raised by some forest company representative could be to make it possible for the machine system to tell the transmission software that some new StanForD files have been automatically created. The transmission software would then not be forced to "look for" available files at a predefined time interval. An even more advanced solution could be that the transmission software tells the machine system to generate certain StanForD-files including data from a certain time interval. However, both of these two solutions mean a significantly deeper integration between the machine system and the transmission software. Such a deeper integration could perhaps be of interest on certain markets but not on other markets.

Another change could be a requirement on "true" on-line reporting, meaning after each harvested stem or each unloaded forwarded load. It would probably be necessary to create new "light versions" of hpr- and fpr-files since both file types would have a relatively large headers when sending individual stems or load.