



Arlinger John
Anders Rowell

18 April 2024

Minutes StanForD

2024-04-18

Place:	Vantaa/Teams, 2024-04-18, 12:00-14:00 EET
Participants:	John Arlinger, Secretary Johan J. Möller, Chairman Anders Rowell, Skogforsk Heikki Komulainen, Ponsse Tapio Räsänen, Metsäteho Timo Melkas, Metsäteho Jonas Gustafsson, Sveaskog Nicklas Nygård, LogSet Henrik Persson, LogMax Antti Korkiakoski, JohnDeere <u>Attending through Teams:</u> Asko Leivonmäki, Logset Bobby Szwajkowski, Tigercat Joshua Hildebrand, Tigercat Etienne Pilon, Logmax Canada Fredrik Tjernberg, Komatsu Hristo Hristoskov, EcoLog Ian Wilson, Forest PHD Jeremy Gibson, Forest PHD Jonas Thorsson, Dasa Lars-Erik Andersson, Dasa Juha-Antti Sorsa, Metsäteho Juha Korhonen, Technion Lars Henriksson, Biometria Lars-Erik Andersson, DASA Louis-Vincent Gagne, FP Innovation Magnus Hansson, SP Mike O'Shea, Coillte Janne Pursiainen, Ponsse

	Ian Wilson, PHD Forest Jeremy Gibson, PHD Forest
--	---

Decision summary

Decided to do the following schema updates:

- **Property in ogi.** It was decided to add new elements Property and PropertyDefinition to the ogi 4.1.
- **MINRegistrationLogLength.** It was decided to add MINRegistrationLogLengthButt and MINRegistrationLogLengthOther elements to the MachineHarvestedProductionType.
- **LogTransport in fpr.** It was decided to add a new LogTransport structure to the fpr-file where log identities and pile coordinates can be reported per partial load.
- **UserID description.** Decided to update the description of UserIDs according to the following text: *Identity per product defined by user of data, normally logging organisation Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, lt, gt, :, " (double quote), /, \, |, *, +.*
- **Version4.1.** It was decided to release version 4.1. during the fall of 2024
 - It was decided to clearly state at the web site that any one today developing support for version 4 should focus on 4.1 and thus avoid 4.0. Basically the 4.0 should be viewed as depreciated.
- **PresentationOrder.** It was decided to update rules 3.3 and 3.8. This means that re-sorting of Products and Deliverys in the machine does not trigger a new Key.

Conclusions from discussions:

- **RelativeLimitationMatrix in pin.** It was decided to postpone the decision to the next meeting when Ponsse has carried out some practical tests.
- **FeatureUserID.** It was decided to postpone the implementation of a FeatureUserID.
- **Incomplete stems/Sub-stems (stem forks).** To be further discussed at upcoming meetings.
- **Emission reporting.** No standard proposals. To continue to monitor this issue at upcoming meetings.
- **BarkFunctionCategory in previous versions.** Decided to let the secretary analyse the possibility of adding a new "Unknown" enumeration to the BarkFunctionCategory which means that future problems of this kind could be handled in a good way.

- **Multi-Tree measuring control in harvesters.** Proposal postponed.
- **Invalid XML characters.** GeoJSON allows some characters that will destroy an xml-structure such as &, < and >. The conclusion was that we should decide on a recommended method for handling these characters. It was thus decided to postpone this issue until the next meeting.

Action list:

- Recommend that unclassified products are reported per SpeciesGroup.
- Recommend that the diameter vector and DBH is adjusted if a short log (butt-off) is completely excluded from the hpr.
- Recommend that by default all butt-offs from single tree processed stems should be registered independent of length.

§1 Opening of meeting and agenda

Chairman Johan J Möller opened the meeting.

The agenda was accepted.

§2 Presentation of participants

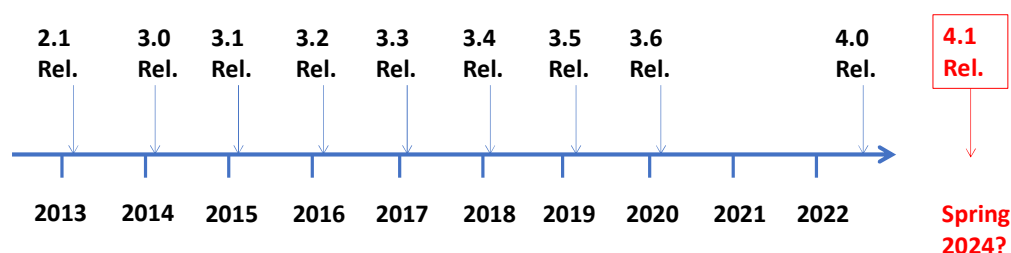
Presentation of meeting participants.

§3 Information about decisions and discussions at previous meeting

Main decisions and conclusions at last meeting (2024-02-07) were presented as described in appendix 1.

§4 Future frequency of StanForD updates

The figure below illustrates the timeline for publishing all versions since 2.1.



The secretary suggested that we decide on releasing version 4.1. during the fall of 2024 as we now have several significant updates proposed such as coding of features in the ogi and the new log positions and IDs in the fpr-file. The conclusion was therefore that we will probably have additional meetings in June and in September. The meeting accepted this schedule and the creation of a pre-release version of 4.1 that will be distributed after this meeting.

Heikki Koumolainen proposed that 4.1 doesn't have to be backwards compatible with 4.0, since anyone is still to implement it operationally.

According to the secretary there are a few test systems with 4.0, but nothing that is in operational use that we know of. He therefore proposed that we remove 4.0 from the website as a solution.

It was decided to clearly state at the web site that any one today developing support for version 4 should focus on 4.1 and thus avoid 4.0. Basically the 4.0 should be viewed as deprecated. From the secretary's point of view, it is preferable to keep backwards compatibility with 4.0 as there are no proposals that makes it worth the price of breaking the backwards compatibility.

§5 Reporting of issues related to version 4.0

No new errors or bugs to report on.

§6 Previously accepted proposals

The secretary presented all previously decided updates (decided between November 2022 and February 2024). No comments or objections were raised and no new decisions were thus made. All decisions described in appendix 1.

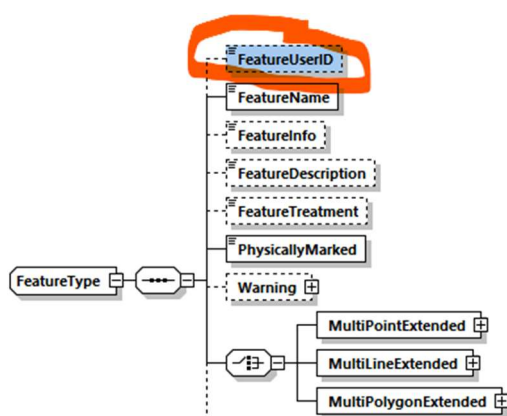
§7 New proposals

FeatureUserID in ogi/oGr

Johan Holmberg, ForestLink.

ForestLink has started to use ogi 4.0, this is also true for CGI. It has been realized that it would be convenient to have the possibility to reference a specific Feature not included inside the ogi, for example additional information describing a polygon not to be harvested due to nature conservation.

It is thus suggested that we add the new id element FeatureUserID under element Feature.



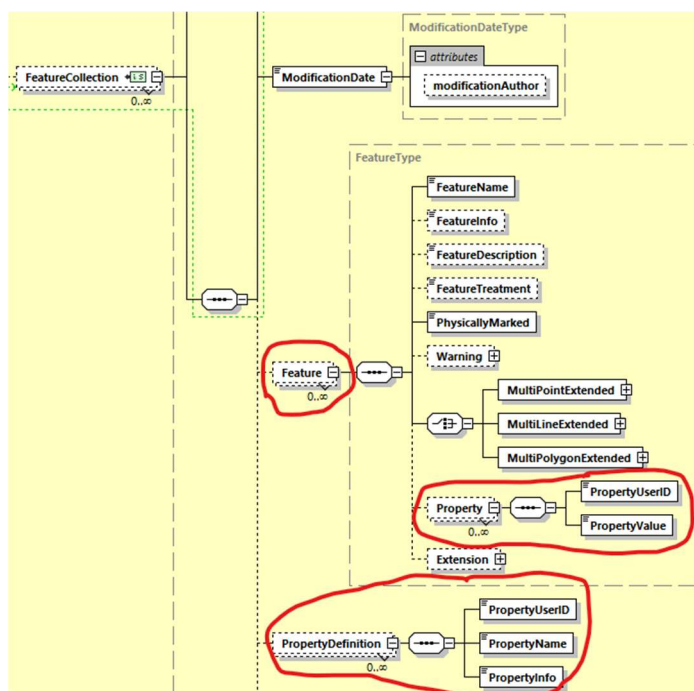
Name	Type	Unit	Description
FeatureUserID	string	Free text	Identity per feature defined by user of data, normally logging organisation. Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, !, gt, :, " (double quote), /, \, , *.

Henrik Persson pointed out that this type of identity could probably be handled by using the proposed new Property structure as described in the following proposal.

It was thus decided to postpone the decision indefinitely and that the secretary will analyze the possibility to use Property for this type of identity.

Feature properties in ogi

Two new structures (Property and PropertyDefinition) were suggested in order to make it possible to include structured information per feature in the ogi message as illustrated below.



Name	Data type	Unit	Description
Property			Structure for register properties per feature
PropertyUserID	String	Free text	Identity per property defined by user of data, normally by logging organisation. Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, !t, gt, :, " (double quote), /, \, , *.
PropertyValue	String	Free text	Property value. Observe that the name and possible ID is found in PropertyDefinition wich is referenced using PropertyUserID.

Name	Data type	Unit	Description
FeaturePropertyDefinition			Structure for register property definitions per feature collection

Name	Data type	Unit	Description
PropertyUserID	String	Free text	Identity per property defined by user of data, normally by logging organisation. Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, !, lt, gt, :, " (double quote), /, \, , *.
PropertyName	String	Free text	Name of property
PropertyInfo	string	Free text	Free text identity information per property. Identity element normally to be used by logging organization. No constraints in standard concerning how to use this element.

Observe that there should be a constraint making certain that any Property.PropertyUserID references an existing PropertyDefinition.PropertyUserID.

Example of how these new structures could be used in order to convey machine readable information, in this case an estimated number of retention trees left after harvesting:

FeatureName	FeatureInfo	FeatureDescription	FeatureTreatment	PhysicallyMarked	MultiPointExtension	MultiPolygonExtension	Extension
1 Träsk	xw	-	Träsk	false	MultiPointExtension		
2 Kulturstubbe	e64	10 st enligt hpr	-	false	MultiPointExtension		
3 Högstubby	c64	9 st enligt hpr	-	false	MultiPointExtension		
4 Plockhuggen yta	ew2	105 st/ha lämnade enligt beräkning. TGL: 45-0-55		false		MultiPolygonExtension	Extension <Property> (3)
5 Avverkad yta	7364734	Avverkat tom 30/6	Hygge	false		MultiPolygonExtension	Extension

PropertyDefinition	PropertyUserID	PropertyName	PropertyInfo
1	Typ	Åtgärdstyp	
2	Mstm	Medelstamvolym	{ "forestand_element": "<ObsP_MeanStemVolumeSub>", "forestand_code": "<http://www.sis.se/ss637009/code_inventobjekt/E20>", "forestand_uom": "m3" }
3	ArH	Areal normal hänsyn	{ "forestand_code": "http://forestand.skogforsk.se/hansynsuppfoljning/hansynsniva/normal", "forestand_element": "<Place> <Area> <Place>", "forestand_uom": "ha" }
4	hstb	Högstubby per ha	{ "forestand_code": "http://forestand.skogforsk.se/stemcodes/hogstubby", "forestand_uom": "per_ha" }
5	gtlm	Grova lämnade tallar	{ "forestand_code": "http://forestand.skogforsk.se/hansynsuppfoljning/hansynsniva/normal", "forestand_element": "<ObsP_AreaStemNumbers>", "forestand_uom": "per_ha", "filter": { "dbhmin_mm": 150, "forestand_code": "http://forestand.skogforsk.se/species/E_1_1_1" } }
6	gim	Grova lämnade träd	{ "forestand_code": "http://forestand.skogforsk.se/hansynsuppfoljning/hansynsniva/normal", "forestand_element": "<ObsP_AreaStemNumbers>", "forestand_uom": "per_ha", "filter": { "dbhmin_mm": 150 } }

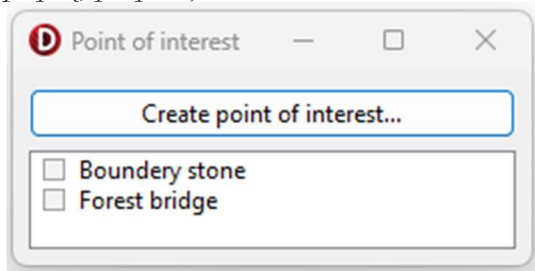
Property	PropertyUserID	PropertyValue
1	typ	Slutavverkning
2	Mstm	0.75
3	ArH	0.66
4	hstb	4
5	gim	11

Another example where the structures are used for registering point of interests in a machine:

Ogi-example illustrating empty FeatureCollection with a list of available properties:

FeatureCollection	Feature	PropertyDefinition
HarvestingFeedback	Null/Empty	"Boundary stone"; Code_68 "Forest bridge"; Code_23

Possible operator GUI for selecting a property of a point (it is possible to register more than one property per point):



Ogr-example illustrating a FeatureCollection with two points added by operator with a selected property:

FeatureCollection	Feature	Feature.Property	PropertyDefinition
HarvestingFeedback	Name: "Boundary stone"; GeoJson: Point X/Y/Z	UserID = Code_68 Value = 1	"Boundary stone"; Code_68
	Name: "Forest bridge"; GeoJson: Point X/Y/Z	UserID = Code_23 Value = 1	"Forest bridge"; Code_23

Jonas Thorsson pointed out that these new elements can be difficult to handle for a machine controller as the different codes in PropertyInfo are not standardized and therefore unknown. There might be cases where ogr-files from different sources are using the same code for different properties. It was pointed out by the secretary and Henrik Persson that it is a very powerful structure for several different reasons, e.g. making it possible to include information from not only a shp-file but also an old dbf-file. Another advantage is that structured information can be communicated between different actors as long as they have exchanged information about the codes in use. Basically this is a situation identical with the StemCodes. This is very important as the ogr messages will most certainly be processed in several steps before reaching a machine and it makes it possible for the operator to modify and add structured information related to individual features as long as a specified code is used.

It was decided to add these new structures to the ogr 4.1.

Handling Butt-offs in hpr

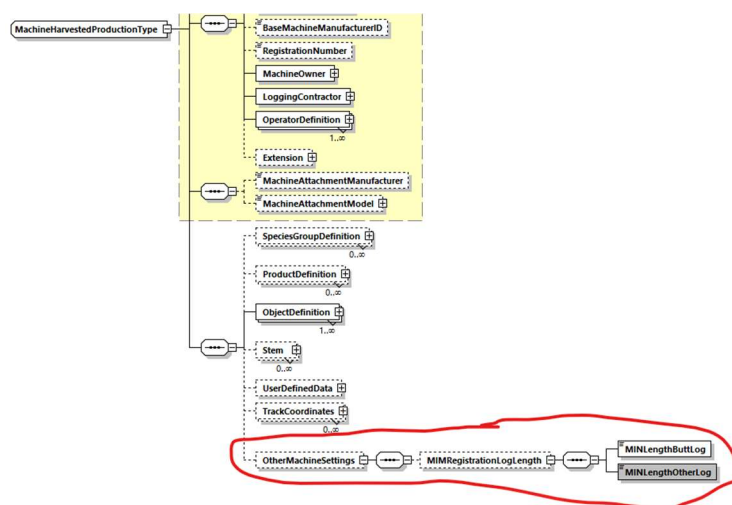
It is possible to completely discard "very short logs", which means that a certain part of the stem may not be reported in the hpr.

Why is there a need to discard anything in the first place? One important cause is when you have trees with thick branches and defects where a stem e.g. might be completely delimbed before you process the stem and a butt-off must be done to "re-set" length measuring. The length of this butt-off has very low accuracy (might be a negative value) and it is thus logical not to register this short log. There are also some markets that require that a butt-off is not registered at all.

It was suggested at the previous meeting to implement a new element MINRegistrationLength. However, the last meeting decided to let the secretary differentiate the proposal based on butt logs and other cuts until the next meeting.

The previous meeting also decided to strongly recommend that the diameter vector is adjusted if a short log is completely excluded.

The following new elements was therefore proposed:



Name	Type	Unit	Description
OtherMachine-Settings			Structure for registering machine settings not communicated through any StanForD instructions.
MINRegistration-LogLength			Structure for minimum length for logs registered in production file as set in the harvester controller.
MINRegistration-LogLengthButt	nonNegativeDecimal	cm	Butt logs shorter than this length are not reported in production file. Set in the harvester controller. This setting is primarily relevant for single tree processed stems.
MINRegistration-LogLengthOther	nonNegativeDecimal	cm	Other logs (not butt) shorter than this length are not reported in production file. Set in the harvester controller. This setting is primarily relevant for single tree processed stems.

It was decided to add these new elements to the MachineHarvestedProductionType and to recommend that by default all butt-offs from single tree processed stems should be registered independent of length. It is also recommended that, as stated in previous minutes, the diameter vector and DBH shall be adjusted in cases where butt-offs are completely discarded.

Rules for Updating PresentationOrder

Proposal originally based on question from Heikki Kuomolainen, Ponsse.

PresentationOrder is one of those messy concepts that looked simple enough when added to the standard, but turned out to be quite complex and often of limited use. When sent from the forest company in a pin it is viewed as an indication of how the product could be sorted. The PresentationOrder is not too difficult to handle in case of SpeciesGroups which are static and quite few. However, in the case of products you often find that a big company may have more than 100 pins in their database (e.g. Sveaskog) and they do not know exactly which combinations of products are used together so it is very difficult to set up PresentationOrder elements in a good way.

It was therefore decided to update the following rules 3.3 and 3.8 in the “rules document”:

R3.3	Key and ModificationDate MUST be updated when modifying a Definition in the machine, e.g. updating prices and length classes of a ProductDefinition.	ModificationDate not to be updated if ProductKey is updated only because SpecieGroupDefinition was modified and SpeciesGroupKey was updated. <u>Another exception to this rule is if Product- or DeliveryPresentationOrder is updated in the machine, no new Key and ModificationDate must be created in this case. This means that re-sorting of Products and Deliverys in the machine does not trigger a new Key</u>
R3.8	It MUST be possible for operator to edit products and species groups during production if attribute modificationRestricted is false. ProductKey and SpeciesGroupKey MUST be updated if product or species group is modified.	<u>An exception to this rule is if Product- or DeliveryPresentationOrder is updated, no new Key must be created in this case. This means that re-sorting of Products and Deliverys in the machine does not trigger a new Key</u>

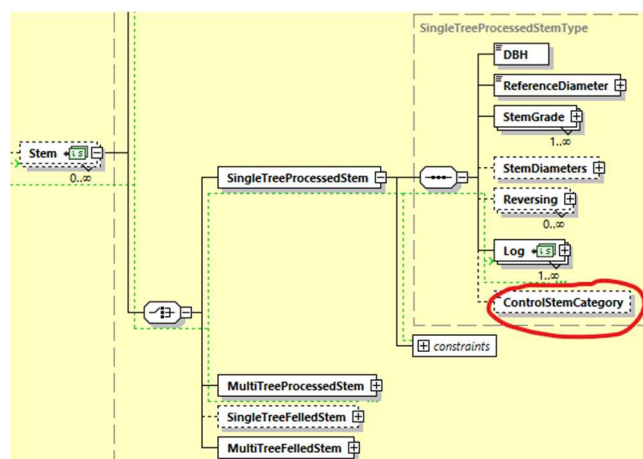
Multi-Tree measuring control in harvesters

There is a need for controlling and calibrating the estimated volumes registered for felled and multi-tree-processed stems. Skogforsk has therefore developed and tested a method for control and possible correction of these volumes. A consequence of this work is a proposal for updating hpr- and hqc-files as described below in order to report control data where a multi-tree-felled bunch is control measured by processing the individual stems.

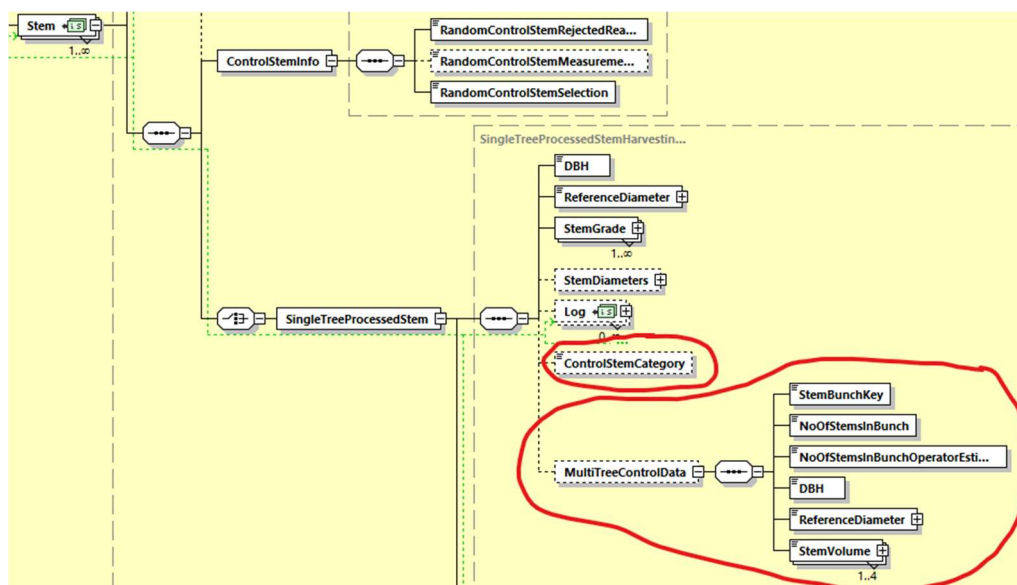
It was proposed to add a new optional element ControlStemCategory to the hpr- and hqc-schemas with one of the following enumeration values as this would make it possible to easily identify hpr stems selected for control:

- *Random SingleTree Control*
- *Operator SingleTree Control*
- *Random MultiTree Control*
- *Operator MultiTree Control*

A suggested update to hpr-schema illustrated below:



Several suggested updates to hqc-schema illustrated below:



Observe that an hqc with MultiTreeControlData would typically only include log measurements from the machine (M1). It is not intended to be sent to a caliper but directly used for control and possible calibration. It could therefore be described as an “M1-file” that is never used for M2 or M3 measurements.

Name	Type	Unit	Description
ControlStemCategory*	string	enumeration	Type of control stem. Enumeration values are: "Random SingleTree control", "Operator SingleTree control", "Random MultiTree Control", "Operator MultiTree Control"
MultiTreeControlData			Structure for MultiTree control data.
StemBunchKey	Integer		A number set in machine for unambiguously separate stem bunches. Only to be reset when starting on a new harvesting object. One single counter for StemBunchKey in both MTP and MTF stem bunches should be implemented in order to avoid confusion. It does not have to be a sequential number always incremented with one for each stem.
NoOfStemsInBunch	Positive-Integer	No	Number of stems in bunch as registered automatically by harvester controller.
NoOfStemsInBunch-OperatorEstimated	Positive-Integer	No	Number of stems in bunch as manually registered by operator. Makes it possible to check if machine stem counter is correct by comparing with NoOfStemsInBunch.
DBH	nonnegative-Integer	Mm	DBH (on bark) measured at height according to DBHHeight (in SpeciesGroupDefinition). Same diameter used in optimisation and volume calculation. DBH is based on butt-end extrapolation function (old SPP) if such function is used at DBH position.

Name	Type	Unit	Description
ReferenceDiameter	nonnegative-Integer	Mm	Diameter at position used for predicting DBH in multi tree harvesting. Normally the first measured diameter. Not to be an extrapolated diameter (e.g. extrapolated using butt-end extrapolation function, old SPP). Can be used when calculating estimated volumes based on volume of single tree processed stems.
StemVolume	Positive-Decimal	M3	Volume of stem

*This element also registered in hpr

How and where to use the control data, calibration method and sampling method was discussed at the meeting. The method for calculating the volume of multiprocessed stems was also discussed. It was pointed out by the secretary that there is no strict standard regarding how to estimate the volumes in case of felling or multi-tree.processing.

It was also pointed out that the proposal might have a consequences on the feller-buncher quality control schema (Bqc). It was noted, by the secretary, after the meeting that the suggested procedure is only relevant for harvester and therefore of little relevance to the bq-schema.

It was decided to postpone this suggestion in order to let the secretary further develop the proposal. A new proposal should also include suggestions regarding calibration and sampling method. It is probably a good idea to consider a completely new structure that handles the fact that the felled stems as estimated by the harvester may differ from the number of single tree processed stems.

CDATA in GeoJSON (non valid characters)

Proposal based on discussion with Fredrik Tjernberg, Komatsu.

GeoJSON allows some characters that will destroy an xml-structure such as &, < and >. This means that we run the risk that the recently added elements MultiPointGeoJSON, MultiPolygonGeoJSON and MultiLineGeoJSON might cause problems. The following example would make the whole ogi-file in-valid since it includes the "&" character:

```
<MultiPointGeoJSON> {
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [-90.071532, 29.951065]
  },
  "properties": {
    "name": "Problems & solutions"
  }
}
</MultiPointGeoJSON>
```

One important reason for considering this issue is that the GeoJSON string is often generated by an external unknown application.

A common solution for this problem is to enclose a string within a so called CDATA section. This means that the following example is valid:

```
<MultiPointGeoJSON><![CDATA[{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [-90.071532, 29.951065]
  },
  "properties": {
    "name": "Problems & solutions"
  }
}]>
</MultiPointGeoJSON>
```

All characters enclosed by a CDATA section are interpreted as characters, not markup or entity references. Every character is taken literally, the only exception being the `]]>` sequence of characters.

It was therefore suggested that we updated the description for elements where GeoJSON strings are enclosed as illustrated below:

Name	Description
MultiPointGeoJSON	Multi point string according to GeoJSON. <u>This string is to be enclosed within a CDATA section in order to avoid problems with invalid characters.</u> This example includes the required tags: <pre><![CDATA[{ "type": "MultiPoint", "coordinates": [[10, 40], [40, 30], [20, 20], [30, 10]] }]></pre>
MultiLineGeoJSON	Multi line string according to GeoJSON. <u>This string is to be enclosed within a CDATA section in order to avoid problems with invalid characters.</u> This example includes the required tags: <pre><![CDATA[{ "type": "MultiLineString", "coordinates": [[[10.0, 10.0], [20.0, 20.0], [10.0, 40.0]], [[40.0, 40.0], [30.0, 30.0], [40.0, 20.0], [30.0, 10.0]]] }]></pre>
MultiPolygonGeoJSON	Multi polygon string according to GeoJSON. <u>This string is to be enclosed within a CDATA section in order to avoid problems with invalid characters.</u> This example includes the required tags: <pre><![CDATA[{ "type": "MultiPolygon", "coordinates": [[[[40.0, 40.0], [20.0, 45.0], [45.0, 30.0], [40.0, 40.0]]], [[[20.0, 35.0], [10.0, 30.0], [10.0, 10.0], [30.0, 5.0], [45.0, 20.0], [20.0, 35.0]], [[30.0, 20.0], [20.0, 15.0], [20.0, 25.0], [30.0, 20.0]]]] }]></pre>

The elements in the table above are located in MultiPointType, MultiLineType and MultiPolygonType.

A second alternative is to use XML escaping where the characters are “converted” as illustrated below:

- & to &
- < to <
- > to >
- " to "
- ' to '

This means that the following text is valid:

```
<MultiPointGeoJSON>{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [-90.071532, 29.951065]
  },
  "properties": {
    "name": "Problems &amp; solutions"
  }
}
</MultiPointGeoJSON>
```

The meeting discussed the advantages and disadvantages with the different methods. The conclusion was that we should decide on one of the methods as recommended and that all members should consider this issue until the next meeting. It was thus decided to postpone this issue until the next meeting.

The secretary did some tests after the meeting and noticed that his two different xml parsers and XML Spy automatically applies the XML escaping. The secretary would therefore prefer XML-escaping. It was also tested by the secretary to enclose an xml-escaped ogi-file inside an envelope. This caused no problems when extracting the files from the envelope.

§8 Items for discussion

LOG IDENTITIES IN FPR AND IMPROVED FORWARDER REPORTING

Forwarder reporting should be improved in order to give more precise information about forwarded logs at the roadside. An important driving force is also to facilitate the development of interfaces, making it simple and easy for forwarder operators to report volumes with less systematical errors. This would make it possible to give better instructions to the truck drivers and in the long term build a complete digital information chain from forest to industry.

It is proposed to include a new structure in the fpr-file (figure 5, 6) whereby it is possible to describe what logs were transported with a load and approximate positions where the logs have been unloaded.

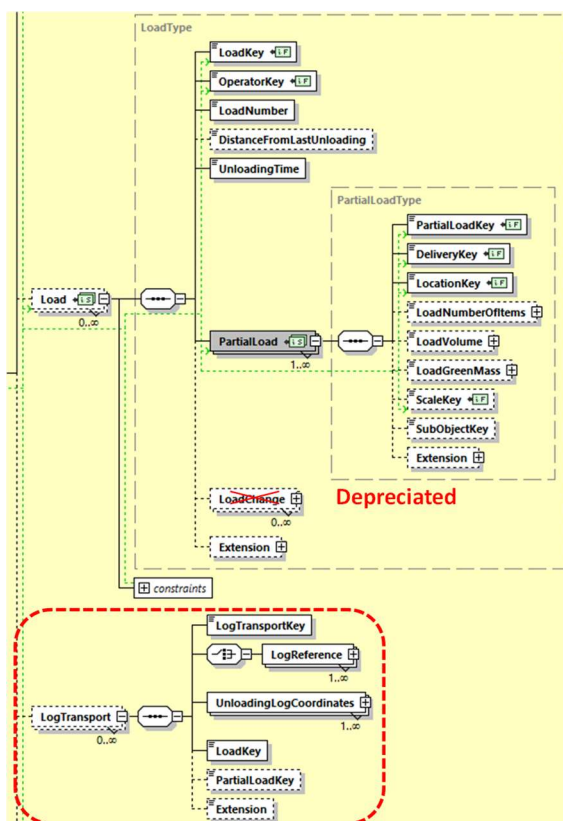


Figure 5. New structure LogTransported in fpr-schema.

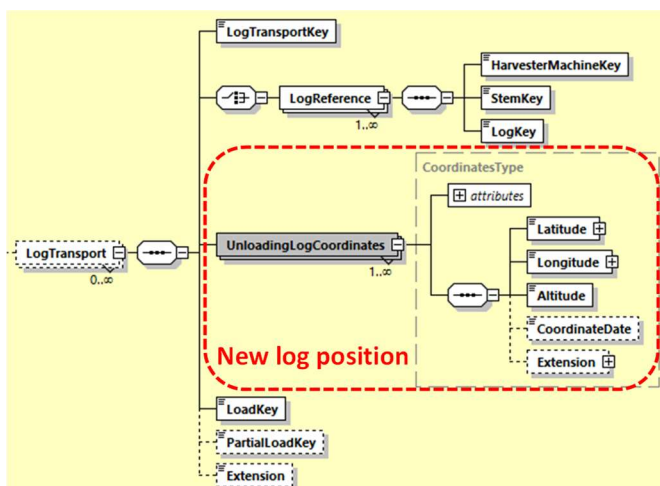


Figure 6. Detailed illustration of the new LogTransport structure, including positions where logs are unloaded and identities of the logs. This structure could be viewed as a "log tracking structure".

Tabel 1. Description of suggested new elements

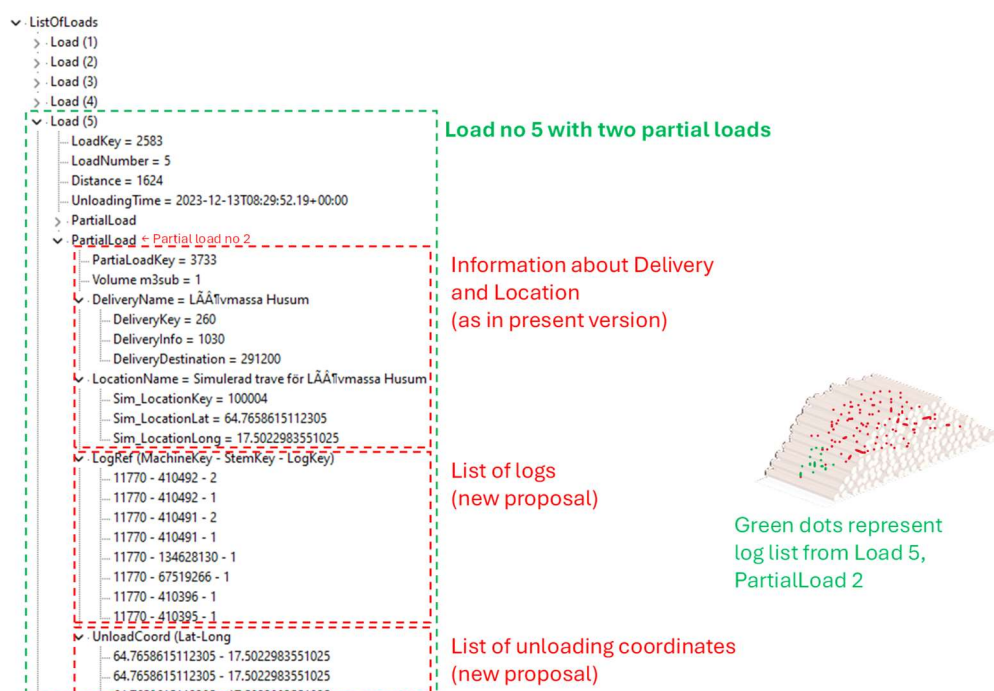
Name	Data type	Unit	Description
LogTransport			Structure for reporting the position where harvested logs have been moved by a forwarder. No PartialLoadKey to be registered if logs are unloaded outside a Location (normally in the forest).

Name	Data type	Unit	Description
LogTransportKey	unsignedLong	Seq. no	Machine specific identity per log transport. Must be totally unique for a certain MachineKey. Reset only when MachineKey is changed
LogReference			Structure for unique log references extracted from hpr-file.
Harvester-MachineKey	String		Globally unique harvester identity (GUID). Extracted from hpr message.
StemKey	unsignedLong		Harvester specific identity per stem. Extracted from hpr message.
LogKey	unsignedLong		Stem specific id for log. Running number reset for each stem. First log (butt end) always LogKey=1. Extracted from hpr message.
UnloadingLog-Coordinates	CoordinatesType		Coordinates describing the position of logs unloaded in a specific Location or outside any Location. Child elements and attributes defined according to the basic CoordinateType
LoadKey	unsignedLong		Machine specific identity per load. Must be totally unique for a certain MachineKey. Reset only when MachineKey is changed. In this case used as a reference to a Load.
PartialLoadKey	unsignedLong		Machine specific identity per partial load. Unique inside Load. In this case used as a reference to a PartialLoad.

Below is an example fpr generated based on data from an existing logging team:

The screenshot displays two hierarchical data structures. The top section, 'Load (8)', contains a table with columns: LoadKey, OperatorKey, LoadNumber, DistanceFromLastUnloading, UnloadingTime, and PartialLoad. The bottom section, 'LogTransport (152)', contains a table with columns: LogTransportKey, LogReference, UnloadingLogCoordinates, LoadKey, and PartialLoadKey. Red arrows indicate the mapping from 'Load' data to 'LogTransport' data. A red dashed box labeled 'List of log ids' encloses a sub-table within 'LogTransport' showing HarvesterMachineKey, StemKey, and LogKey. Another red dashed box labeled 'List of unloading coordinates' encloses a sub-table showing latitude, longitude, and altitude data for product positions.

Illustration of how data example above can be presented:



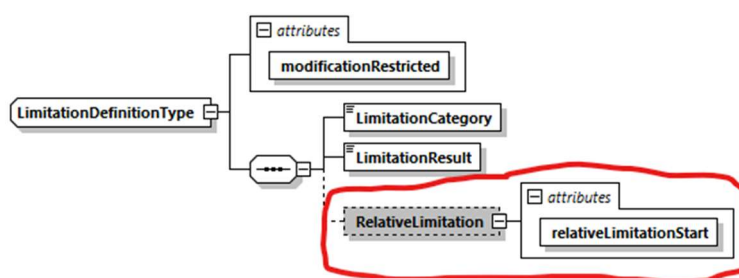
The meeting decided to add the new LogTransport structure to the fpr-file.

RELATIVELIMITATIONMATRIX

The basic idea, as described in a number of previous meetings, is to introduce a new type of limitation matrix where the limit is relative rather than absolute. The idea is also to include an attribute “relativeLimitationStart” that indicates when this type of limitation will affect the bucking optimization.

The proposal is to add the following new element and attribute to the LimitationDefinition:

`<RelativeLimitation RelativeLimitationStart="5">true</RelativeLimitation>`



Name	Data type	Unit	Description
RelativeLimitation	Boolean		LimitationCategory and PriceMatrix.Limitation interpreted as a relative values if value is TRUE. All other cases interpreted as an absolute limitation.
relativeLimitationStart	nonnegative-Integer	Pcs/m3	Minimum total number/volume of harvested log per product before the relative limitation matrix is applied. Relative limitation not in use when total volume is below this value. Logs harvested before this limit is reached is taken in account in the limitation calculation.

A first practical test has been carried out by Ponsse. However, some bugs were found that means that a new test will be run in June or August. The meeting decided to postpone the decision to the next meeting.

REGISTERING STEMS WITH “SUB-STEMS” (FORKS) IN HPR

A question has been raised in Finland how stems with two or more “sub-stems” (stems with one or several forks) should be measured and saved in the hpr so that these stems could be reported in a consistent way.



An argument for the improvement is that the count of harvested trees can be calculated incorrectly for objects including a lot of stems with forks depending on the harvester’s registration method and operator’s working style.

Currently a stem with a fork may be registered in different ways depending on controller and how the operator processes the stems:

- Logs of sub-stems included in the main stem.
- Sub-stems and their logs included as separate single stems.

Development of automatic calculation of thinning results and harvesting quality monitoring systems based on hpr-data require that the count of harvested stems is correct. This is especially important in first thinnings.

This also influences the calculation of average tree size (volume or DBH) and therefore the price of harvesting. This means that a correct stem count is also a cost-calculation and harvesting work compensation matter.

The objective of the proposal is that if a sub-stem is saved as a separate stem, it should have a reference to the main stem. However, it is a practical challenge that stems with sub-stems can be processed in so many ways.

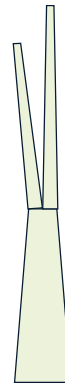
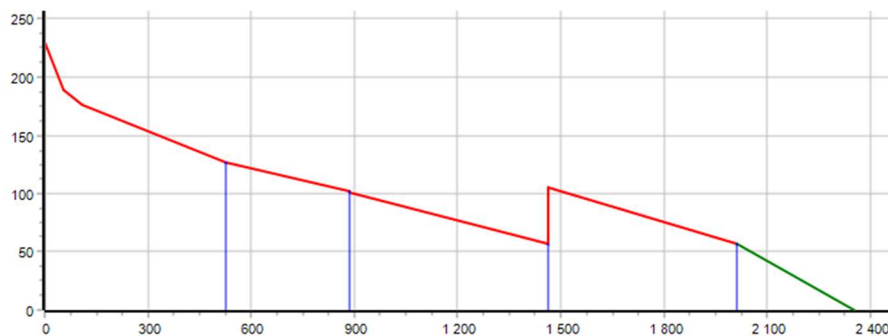
How these stems are registered also affects the post-processing of the diameter vector.

Example all forks registered as one single stem

Stem count, basal area and mean volume of removed trees are correct when used for

- follow-up of thinnings
- calculating the remaining stock
- updating forest resource data based on hpr-data
- using harvester data as ground truth for remote sensing

Processing time per stem gets higher which has an effect on productivity key figures and pricing of logging.



Example forked stem registered as separate stems

Stem count, basal area and mean volume of removed trees may be misleading when used for

- follow-up of thinnings
- calculating the remaining stock
- updating forest resource data based on operations
- using harvester data for ground truth of remote sensing

Processing time per stem corresponds better to actual logging conditions when used for productivity key figures and setting the price of logging.



Discussion

Tapio pointed out that this is a complex issue that does not need to be solved immediately.

The discussion that followed made it clear that there are a number of different cases that might result in processing “incomplete” stems:

- A. Stem forks
- B. Breakage (e.g. tree length operations or snow damage)
- C. Separate felling and processing by harvester (e.g. in steep terrain).
- D. Manual felling and cutting of first log before processing of the remaining stem.

Two possible standard adjustments/clarifications were discussed:

- Allowing increasing diameters when comparing separate logs. This is already done by at least one manufacturer.
- Open up the possibility of registering a 0-value (zero) for DBH in cases where the butt-part of the stem is not included. This case would also mean that the ButtEndExtrapolation function is not to be used. This solution has some basic similarities with the suggestion for an element UncompleteStemTop.

The meeting decided to discuss this issue further at upcoming meetings.

EMISSION REPORTING

Fuel costs for logging have increased in recent years. As energy efficiency reporting evolves, routines are needed to monitor fuel consumption and emission developments transparently, uniformly, representatively and at a sufficiently individual level (support for emission-efficient choices).

There are initiatives from EU (Corporate Sustainability Reporting Directive) whereby relevant reporting of CO₂ emissions will become mandatory.

How to report fuel and CO₂ emissions has been discussed at a large number of meetings during the last years. It has e.g. been discussed to create a new message or to include this information in the production files.

Metsäteho has, after discussions with both Skogforsk, Finnish contractors union, Finnish forest companies and machine manufacturers, compiled the following list of different possible solutions:

1. **To add fuel consumption information in production files (hpr and fpr) to Extension construct.**
Quick and simple, but not a permanent solution for testing and development purposes.
Machine manufacturers’ own decisions to add the needed information.
Aim is to get the fuel consumption on Object level only .
Saving the information should be based on operator’s selection in machine settings.
Fpr of limited use in Finland
2. **To add fuel consumption and fuel category (type, name etc.) as new elements in StanForD production files (hpr, fpr)**

Change in version 4.1

On Object and SubObject levels?

To consider if amount of CO2 emissions should be added too.

Regarding the possibility to calculate them either in the machine or in the cloud service of the machine manufacturer.

Saving the information should be based on operator's selection in machine settings.

Fpr of limited use in Finland.

3. **New file type EnvironmentalEmissionMonitoring**

As designed by Skogforsk and Metsäteho (draft version).

A new file type would require changes in WoodForce and in forest company ERP's, not a prioritized solution in Finnish forest companies. However, could it be a long-term solution?

4. **Third party mom conversion tool (parser) for filtering an emission report**

There are already mom converters made which could be modified for the purpose (Skogforsk, Luke, Metsäteho).

Tool could be placed in the machine system environment and be connected to production reporting behind a setting. Increased complexity in file handling and sending for the operators?

Optionally it could locate in the cloud service. How would the access to the forest companies be arranged in that case?

On whose responsibility the tool would be? How the tool would be connected to the other software and updates?

5. **Fuel consumption and other information via machine manufacturer's cloud service to the emission calculation system**

Fuel consumption to machine manufacturer's cloud either from mom, hpr/fpr or manufacturer's internal API interface. Method is not a StanForD issue.

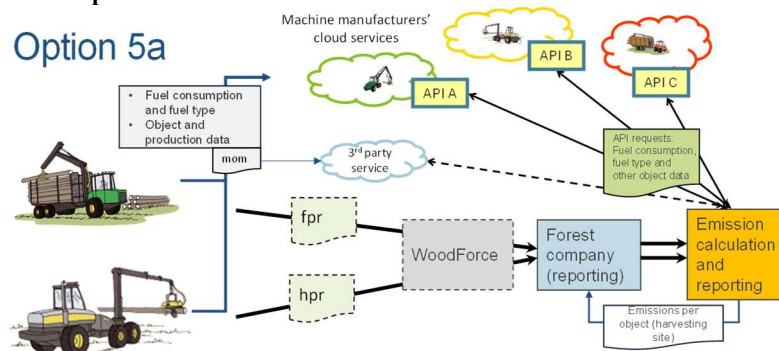
Fuel consumption and required object and production data from cloud to the emission calculation system based on authorized user rights. Based on machine manufacturer's cloud service API or emission calculation system API. User right and data permission management questions?

Alternatively, production data could be merged from other sources to the emission calculation system, but this should be avoided e.g. full hpr and fpr from WoodForce

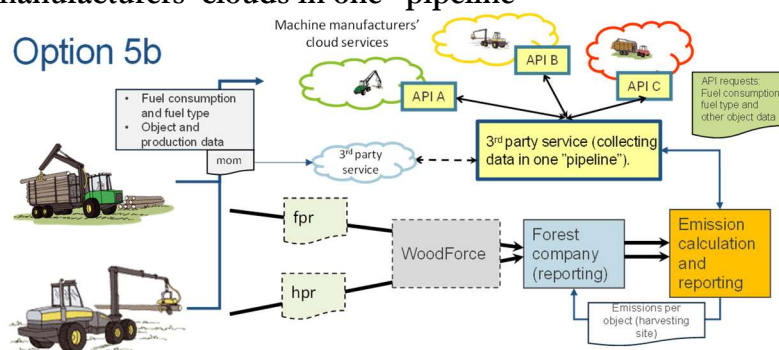
Depending on the solution design, all machine specific API interfaces should probably be supported in the emission calculation system of the forest company

Increasing need to promote StanForD Data-API ?

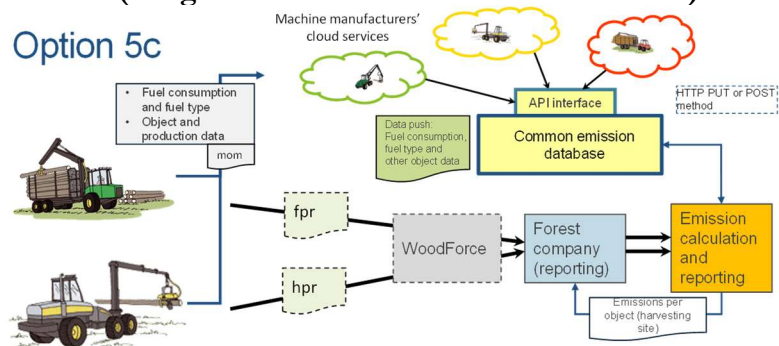
- a) **Applying each machine manufacturers own API interfaces in data requests**



- b) **3rd party service for collecting data from machine manufacturers' clouds in one "pipeline"**



- c) **Data push from manufacturers' clouds to a common database (integrated to emission calculation service)**



The following tables describe the ranking of the alternatives described above. The ranking has been done by Metsäteho and is based on the discussions with the Finnish forest companies in ACE project.

Ranking of the alternatives in the short term (POC and piloting):

Rank.	Altern.	+	-	Comments
1	4	Quick and simple solution Low costs	Some changes to file management and sending logic are probably needed in the machines.	To be used for concept design, testing and piloting only. Responsibilities of the conversion tool support in machine systems probably cause that this can not be a long term solution. Data management in WoodForce?
2	1	Quick and simple solution Low costs	Extension not possible to implement quickly by some of the manufacturers? Risk of incompatible data from different machines. Not widely applicable for forwarders.	To be used for concept design, testing and piloting only. Limited testing potential of fpr files in Finland. Data management in WoodForce?
3	5a, 5b	Flexible, extendable and modern solution No or very little costs to manufacturers	Access to data for different customers of the entrepreneurs must be managed in cloud services (varying access rights etc.). Format of the data may differ between the machines (converting to be done in the emission calculation system).	Options 5a and 5b will probably not require any technical changes from machine manufacturers to current cloud interfaces. Experiences from API based solutions in general can be achieved by testing and piloting these options.

Ranking of alternatives in the long term:

Rank.	Altern.	+	-	Comments
1	5c	Flexible, extendable and modern solution Advanced possibility of managing data and further development for the end user party	Longer development time and higher costs than in other options? Access to data for different customers of the entrepreneurs must be managed in cloud services (varying access rights etc.).	Amount of technical changes to machine manufacturers' cloud interfaces should be small. What kind of experiences there are of this kind of solution based on automatic sending of data as push mode? API-based solution can be extended to other kind of reporting purposes and applications.
2	5b	Flexible, extendable and modern solution No or very little costs to manufacturers Better maintenance of business-critical data	Access to data for different customers of the entrepreneurs must be managed in cloud services (varying rights etc.).	Options 5a and 5b will not require any technical changes from machine manufacturers to cloud interfaces. API-based solution can be extended to other kind of reporting purposes and applications.
3	5a	Flexible, extendable and modern solution No or very little costs to manufacturers	Access to data for different customers of the entrepreneurs must be managed in cloud services (varying rights etc.). Complexity in the management of several API interfaces in option 5a?	Options 5a and 5b will not require any technical changes from machine manufacturers to cloud interfaces. API-based solution can be extended to other kind of reporting purposes and applications.

Rank.	Altern.	+	-	Comments
4	2	A traditional standard solution Easy to implement Low costs	Data management in WoodForce? Not widely applicable for forwarders.	Standardized solution for further implementations. Requires probably a common agreement for fuel consumption data exchange from contactors side.
5	3	Extendable solution	Extra changes and costs in machine and forest company systems	New file type which can be extended to various types of emission reporting. Supporting a new file type causes changes and costs in logging information systems.
6	4	Quick and simple solution Low costs	Responsibilities of the tool support in machine systems Some changes to file management and sending are probably needed in machines Special agreements with the entrepreneurs regarding data exchange (e.g. updates of contracts)	Probably not a long term solution, but good for piloting. Requires probably a common agreement from contactors side if used as long term solution. Responsibilities of the tool support might be problematic from point of view of the machine manufacturer.

Tapio Räsänen pointed out that the ranking above means that there is no concrete proposal for any StanForD updates at the moment.

The meeting discussed several different aspects of fuel reporting such as reasons for variation in accuracy, control/calibration of fuel consumption measuring and methods used by other similar business activities (e.g. construction / earth moving machines).

It was pointed out that the EU Directive will give some credits (-25%) if directly measured data is reported in comparison to using statistical data.

It was decided to continue discuss and monitor this issue at upcoming meetings.

§9 StanForD2010 projects / implementations

WOOD VALUE TEST

Suggested time frame is October and November 2025.

§10 Other issues

BARK FUNCTIONS IN PREVIOUS HPR VERSIONS

It was suggested with the previous minutes to recommend that the enumeration value “None” is registered in attribute “barkFunctionCategory” in cases when converting bark parameters from version 4 to version 2 and 3.

However, several participants noted that this might indicate that no bark function has been used at all and that m3sob and m3sub thus would be equal. The previous meeting decided to let the secretary analyse this issue further.

The evolution of bark functions within StanForD is described below.

StanForD Classic (apt-file) had the following bark functions:

- *Zacco*, linear function based on diameter ob (var113_t1), included 1988.
- *German* table, bark deduction based on diameter ob (var113_t2 and var113_t3), added to standard 1995
- *Skogforsk 2004, Scots pine*, added to standard 2004.
- *Skogforsk 2004, Norway spruce*, added to standard 2004.

The *GermanDistanceBased* function was added with StanForD2010, version 3.0.

The *LogisticTypeBarkFunction* (Lithuanian) bark function was added with StanForD2010, version 3.4.

Two new functions were also added with 4.0 in the form of a “*2nd degree polynomial*” and a “*Logarithmic*” function. Both of these are included in the Consolidated function. The ConsolidatedFunction also includes the old linear *Swedish Zacco* function and the *Skogforsk 2004, Norway spruce function*.

It can thus be noted that this issue has existed for a fairly long period of, e.g. if generating a 2.0 hpr based on a spi according to 3.4 (both *GermanDistanceBased* and *LogisticTypeBarkFunction* missing from 2.0).

The StanForD2010 versions up to 3.6 all requires attribut BarkFunctionCategory and element BarkFunction. One of the enumerations in BarkFunctionCategory is “*None*” which is used on markets where volumes and diameters under bark is not in use at all (e.g. Finland).

A solution could be to either use enumeration value “*None*” or to use a specific function such as the linear *Zacco* function with parameter values set to 0. The second alternative would indicate that a bark function has been used but is not supported.

The meeting decided to not include any recommendations for these special cases. However, it was decided to let the secretary analyse the possibility of adding a new “*Unknown*” enumeration to the BarkFunctionCategory which means that future problems of this kind could be handled in a good way.

PLUS (+) CHARACTER IN USERIDS

Issue raised by Jonas Thorsson, Dasa.

Description of ProductUserIDType is included below:

*Identity per product defined by user of data, normally logging organisation Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, lt, gt, :, " (double quote), /, \, |, *.*

However, when looking at the data type RestrictedStringType it can be noted that a plus (+) character is included:

```
<xsd:simpleType name="RestrictedStringType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="([S-[%?&lt;&gt;&quot;&apos;\\|*+]])+([S-[%?&lt;&gt;&quot;&apos;\\|*+]])*" />
  </xsd:restriction>
```

</xsd:simpleType>

It was therefore decided to update the description of UserIDs according to the following text:

*Identity per product defined by user of data, normally logging organisation Do not use a single space or null value. Do not begin or end with a space. Do not include any of the following characters: Double space, Linefeed, carriage return, %, ?, lt, gt, :, " (double quote), /, \, |, *, +.*

NEW FINNISH STEM CODES

Two new stem codes has been added to the Finnish default spi-file.

StemCode (v3.x – 4.x)		@stemCodeInfo (v4.x)	@stemCodeGroup (v4.x)
Tekopötkkelö	High stump	Rungosta on jätetty osa tekopötkkelöksi	Nature
Säästöpuuryhmä	Retention tree group	Säästöpuuryhmän ympäriltä tai yksittäisen säästöpuun läheltä kaadettu puu, jolla merkitään säästöpuiden sijainti	Nature
<u>Suojatiheikkö</u>	<u>Protective thicket</u>	<u>Suojatiheikön ympäriltä tai sisältä kaadettu puu, jolla merkitään suojatiheikön sijainti</u>	<u>Nature</u>
Luonto- tai suojelukohde	Nature reservation	Uuden käsittelemättä jätettävän luonto- tai suojelukohteen sijainti tallennetaan merkitsemällä yksi tai useampi sen läheisyydestä kaadettu runko	Nature
<u>Muinaismuisto-kohde</u>	<u>Culture reservation</u>	<u>Uuden käsittelemättä jätettävän muinaismuistokohteen sijainti tallennetaan merkitsemällä yksi tai useampi sen läheisyydestä tai kohteen sisältä kaadettu runko</u>	<u>Culture</u>
Myrskytuho	Windthrow	Hakkuukoneella käsiteltävät myrskytuho- tai tuulenskaatorungot merkitään	Special harvesting
Hyönteistuho	Insect damage	Runko, jonka puutavaran laatua hyönteistuho (esim. kirjanpainaaja) on heikentänyt	Special harvesting
Lahorunko	Decayed tree	Juurikäävän tai muun lahonaiheuttajan infektoima runko	Special harvesting
Latvakatko	Top breakage	Runko, jonka latva on katkenut ennen hakkuuta	Special harvesting
Muu tuho	Other damage	Runko, jossa muu tuhonaiheuttaja kuin hyönteiset on aiheuttanut runko- tai puuaineksen laatuvaurioita	Special harvesting
Poikkeava käsittely	Exceptional processing	Esim. monihaaraisen tai hyvin järeän rungon käsittely kahtena runkona	Special harvesting
Korjuuvaurio	Logging damage	Korjuuvaurio	Special harvesting
Ajoura	Trees cut from strip road outside object	Leimikon rajauksen ulkopuolella tehty ajouran aukaisu tai muun vastaavan syyn takia kaadettu puu	Other
Varastopaikka	Trees cut from landing outside object	Leimikon rajauksen ulkopuolella tehty varastopaikan aukaisu	Other
<u>Hakkuu suojajalueelta</u>	<u>Trees cut from protection area</u>	<u>Hakkuu vesistön suojavyöhykkeeltä, säästöpuuryhmästä, suojeltavalta tai muuten käsittelemättä jätettävältä alueelta</u>	<u>Other</u>
Muu	Other	Muu syy	Other

§11 Next meetings

StanForD meeting 2024-06-19, Teams (10:00-14:00 CET)

StanForD meeting 2024-09-12, Teams (10:00-14:00 CET)

StanForD meeting 2024-11-13, Uppsala/Teams (10:00-15:00 CET)

By default, all financially contributing members will receive an Outlook invitation to all meetings in advance.

§12 End of meeting

The chairman Johan J Möller thanked everyone for attending the meeting.



Arlinger John

2024-03-28

Previously accepted 4.1 proposals

Meeting 2022-11-16:

- Description of element FillStyle is to be updated in the DesignInstruction. The following enumerations are not to be used (depreciated) as they are already covered by by element Color and FillOpacity: LightGrayFill, GrayFill, DarkGrayFill and HalfTransparentFill.

New description :

Fill style per StyleDefinition ~~format class and layer~~. Allowed values are: SolidFill, TransparentFill, HorizontalFill, VerticalFill, UpwardDiagonalFill, DownwardDiagonalFill, CrossFill and DiagonalCrossFill. The following values not to be used as they are in conflict with elements FillOpacity and Color: LightGrayFill, GrayFill, DarkGrayFill and HalfTransparentFill

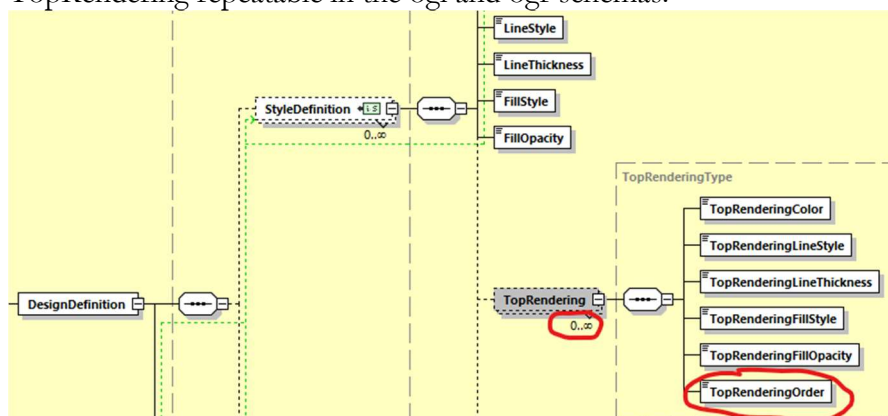
Meeting 2023-04-26:

- The term “base diameter for extrapolation” and abbreviation D_{base} is to be used for ButtEndExtrapolation elements.

Element name	Present description	New description
ButtEndProfile-Extrapolation	Structure for butt end profile extrapolation method. Extrapolation method must be used if this structure is included in spi. Extrapolation method must be used also in cases where you get measurement values below breast height (normally 130 cm).	Structure for butt end profile extrapolation method. Extrapolation method must be used if this structure is included in spi. Extrapolation method is used even if machine generates measurement values below ReferenceHeight (normally 130 cm). Volume calculation of butt end is done according the extrapolated diameters.
ReferenceHeight	Height from stump to the diameter used for predicting the butt end diameters. Normally breast height (130 cm in Finland, 120 cm in Sweden) is used for extrapolation of butt end diameter values.	Height from fell cut point (H_r) to the position of the base diameter used for predicting the butt end diameters. Abbreviation D_{base} used for measured diameter at this position. Normally at 130 cm but it is possible to develop extrapolation functions based on other height positions.
Extrapolation-Coefficient	Extrapolation coefficient tables: coefficient values (multiplying values) / distance class / diameter class / tree species: 1...var170_t4 / 1 ... var170_t3 / 1...var111_t1. Example: Measured diameter at reference height (normally dbh) of stem = 23,3 cm. Distance to	Extrapolation coefficient tables with coefficient values per distance class and base diameter class (D_{base}). Example: Measured diameter at reference height (D_{base}) of stem = 23.3 cm.

Element name	Present description	New description
	extrapolation point = 90 cm. Coefficient value in table at (23, 90) = 1.042. Estimated diameter value at 90 cm = 23,3 cm * 1.042 = 24,3 cm (243 mm).	Distance to extrapolation point = 90 cm. Coefficient value in table at (23, 90) = 1.042. Estimated diameter value at 90 cm = 23.3 cm * 1.042 = 24.3 cm (243 mm).
diameterClass	Diameter class values for the reference height in extrapolation coefficient table / species group.	Base diameter class values for the reference height in extrapolation coefficient table.
ButtEndProfile-Extrapolation-FunctionCategory	extrapolation of butt end diameter values. Volume calculation of butt end is done according the extrapolated diameters. Allowed values: Poikela 2007 => Function DL = $[1 + (a_0 * (1.3 - L) + a_1 * (1.3 - L)^2) / 100] * D_{1.3}$ $a_0 = a_{00} + a_{01} * DBH + a_{02} * DBH^2 + a_{03} * DBH^3 + a_{04} * DBH^4$ $a_1 = a_{10} + a_{11} * DBH + a_{12} * DBH^2 + a_{13} * DBH^3$ $a_2 = a_{20} + a_{21} * DBH + a_{22} * DBH^2$	Type of function for extrapolation of butt end diameter values. Allowed value: Poikela 2007 => Function DL = $[1 + (a_0 * (H_r - L) + a_1 * (H_r - L)^2) / 100] * D_{base}$ $a_0 = a_{00} + a_{01} * D_{base} + a_{02} * D_{base}^2 + a_{03} * D_{base}^3 + a_{04} * D_{base}^4$ $a_1 = a_{10} + a_{11} * D_{base} + a_{12} * D_{base}^2 + a_{13} * D_{base}^3$ $a_2 = a_{20} + a_{21} * D_{base} + a_{22} * D_{base}^2$ where DL = diameter at the distance L from the felling cut point, cm L = distance from the felling cut point, m Dbase = measured diameter at the reference height Hr = reference height from the felling cut point, cm Note that function Poikela 2007 was developed for Hr being 1.3 m (element ReferenceHeight with value 130 cm). Hr must be 1.3 m in case of using parameters according to Finnish regulations.

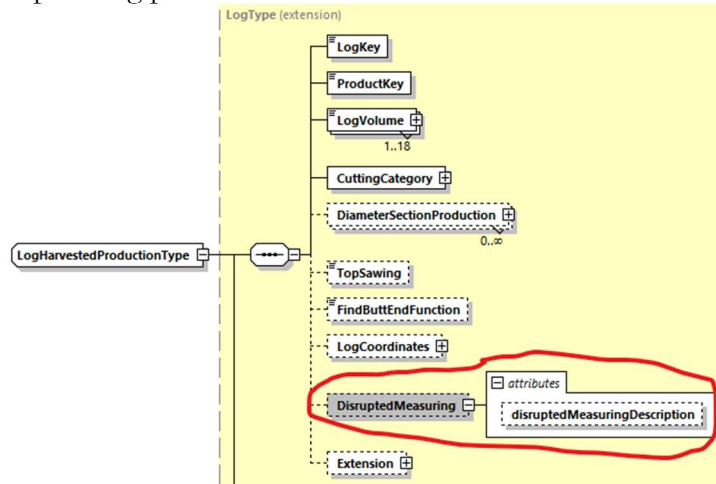
- Add a new optional element TopRenderingOrder and to make TopRendering repeatable in the ogi and ogr schemas.



Name	Type	Unit	Description
TopRendering-Order	Positive integer	no	Order of presentation for rendering additional styles (e.g. colors) on top of the basic style definition. For example, 1 is the topmost layer

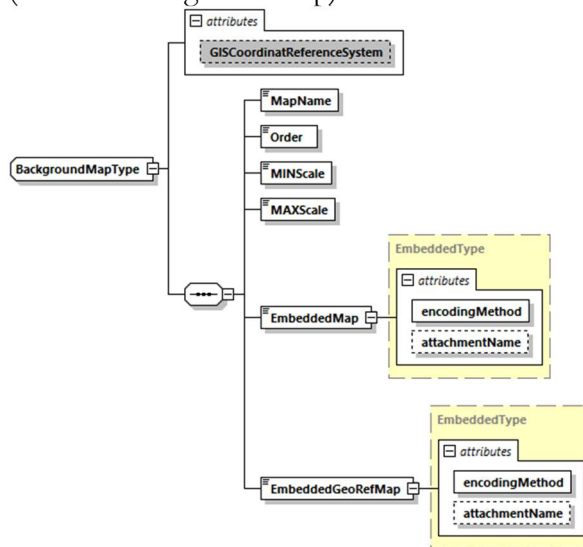
- New element DisruptedMeasuring with attribute disruptedMeasuringDescription to be added to LogType. This is in order to indicate that a stem or log has been handled in a way that it is not suitable for measuring control or calibration. For example if a “reversed

top sawing procedure” has been used.



Name	Type	Unit	Description
Disrupted-Measuring	boolean	true/false	Disruptions in measuring of log that decreases the measuring accuracy in a significant way. Typically, value is true if length measuring wheel is not in constant contact with the stem. An example is if the head is completely opened during feeding of log due to significant defect such as stem fork.
disrupted-Measuring-Description	String	Free text	Text describing reason for disrupted measuring. Free text set in the harvester controller.

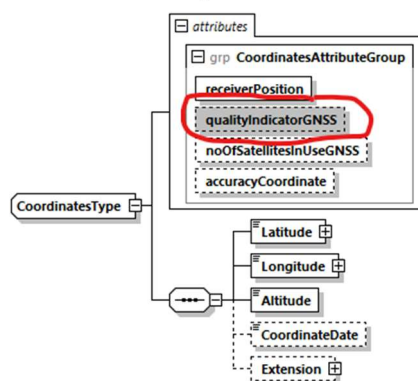
- Add new GISCoordRefSystem attribute (GISCoordinateReferenceSystemType) also to the background maps (element BackgroundMap) in version 4.1.



Name	Type	Unit	Description
GISCoord-RefSystem	string	code	Coordinate reference system used for multipolygon/line/point, codes according to the EPSG database (http://www.epsg.org/). Not used for individual coordinates registered by machine, e.g. per harvested stem or forwarder location. EPSG-value must have pre-fix "EPSG:" and after that at least four digit integer. Example of valid EPSG-value is "EPSG:3006"

Meeting 2023-11-22:

- The following two new enumerations are to be added to attribute qualityIndicatorGNSS:
 - "PPP RTK Fix" (Precise point positioning fix and Real Time Kinematic)
 - "PPP RTK Float" (Precise point positioning fix and Float Real Time Kinematic)



- The data type DegreeType is to be changed from nonNegativeInteger to nonNegativeDecimal in order to improve the precision.
- Element FelledStemCoordinates is to be repeatable to make it possible to register both BaseMachine and Attachment center when stems are laid down on the ground.



Meeting 2024-02-07:

- Element EmbeddedGeoRefMap to be made optional as some GIS formats do not require a separate world file with geo-references.

