

Overview of the EU ETS Reporting Language (XETL)

General

The EU ETS Reporting Language is an electronic reporting language supporting EU ETS Monitoring, Reporting and Validation (MRV) activities such as submitting monitoring plans, reporting annual emissions or verification and improvement reports, both for stationary installations and aircraft operators.

The Reporting Language has updated by the European Commission in 2015 with the support of a Technical Working Group consisting of emissions trading specialists from many Member States.

Architecture

The EU ETS Reporting Language is an XML-based language designed using version 1.1 of the XML Schema Definition (XSD) standard.

The Reporting Language consists of a modular set of XML schemas that can be seen from two perspectives: the type of operator or the type of document they allow to create. The table below groups the 10 main schema files following these perspectives.

Operator / Document type	Stationary installation	Aircraft operator
Monitoring plan	installation_monitoring_plan.xsd	aircraft_operator_monitoring_plan_ae.xsd aircraft_operator_monitoring_plan_tkm.xsd
Annual emissions report	installation_annual_report.xsd	aircraft_operator_annual_report_ae.xsd aircraft_operator_annual_report_tkm.xsd
Improvement report	installation_improvement_report.xsd	aircraft_operator_improvement_report.xsd
Verification report	installation_verification_report.xsd	aircraft_operator_verification_report.xsd

Each of these schema file has one or several root element serving at instantiating an XLM document. They are listed in the table below.

Operator / Document type	Stationary installation	Aircraft operator
Monitoring plan	monitoringPlan	monitoringPlan
Annual emissions report	report reportRawData reportSummary	report reportRawData reportSummary
Improvement report	improvementReport	improvementReport
Verification report	verificationReport	verificationReport

As the definition of three root elements in the annual emissions report schema indicates, XETL allows the generation of annual emissions report documents in several ways:

- A document including only the final, calculated report figures, called *report summary*.
- A document including only the base information necessary to calculate the summary figures, called *report raw data*.
- A document including summary and raw data, called *report*. This is the document that is expected to be sent to authorities as it is a legal obligation for operators to submit a complete report including both the base and calculated figures.

It is worth noting that the other types of documents (monitoring plan, improvement report and verification report) also include values that are the results of calculations. Unlike for annual emissions reports, these values cannot stand on their own in a separate document. Therefore, an explicit root element does not exist for them and the root element for monitoring plan, improvement report or verification report simply includes a *derivedData* child element, grouping and highlighting derived values without separating them or allowing their instantiation as a distinct document.

In addition to the 10 main schema files, a set of other schema documents that provide common type definitions are also part of the XETL language. They are:

- types_common.xsd,
- types_common_annual_report.xsd,
- types_common_improvement_report.xsd,
- types_common_monitoring_plan.xsd
- types_common_verification_report.xsd
- types_aircraft_operator.xsd
- types_aircraft_operator_annual_report.xsd
- types_aircraft_operator_authorities.xsd
- types_aircraft_operator_monitoring_plan.xsd
- types_aircraft_operator_tiers.xsd
- types_aircraft_operator_tools.xsd
- types_installation.xsd
- types_installation_activities.xsd
- types_installation_crf.xsd
- types_installation_eprtr.xsd
- types_installation_tiers.xsd
- types_country_codes.xsd

Features

Data validation

The EU ETS Reporting Language provides a wide range of validation features. They are described below.

Validation category	Description
Data type	<p>Data types specify the type of each data in the document: integer, string of characters, etc.</p> <p>E.g. the concentration in GHG of a flue gas is a percentage (that is, a value between 0 and 100) with up to 4 fractional digits.</p> <p>Restriction of data types are enforced by the <i>type</i> attribute on each element or XETL attribute. These types are either</p> <ul style="list-style-type: none"> Native XML Schema types¹; e.g. <pre data-bbox="491 533 1209 566"><xs:attribute name="reference" type="xs:ID" use="required"/></pre> XETL specific simple types; e.g. <pre data-bbox="491 611 1193 645"><xs:element name="emissionCO2e" type="inst:Emission"/></pre> <p>Where <i>inst:Emission</i> is defined as</p> <pre data-bbox="491 696 930 831"><xs:simpleType name="Emission"> <xs:restriction base="xs:decimal"> <xs:fractionDigits value="4"/> <xs:minInclusive value="0"/> </xs:restriction> </xs:simpleType></pre> XETL complex types; e.g. <pre data-bbox="491 925 1114 992"><xs:element name="monitoringPlanReference" type="ns:MonitoringPlanReference" minOccurs="0"/></pre> <p>Where <i>ns:MonitoringPlanReference</i> is defined as</p> <pre data-bbox="491 1043 1249 1211"><xs:complexType name="MonitoringPlanReference"> <xs:sequence> <xs:element name="version" type="cmn:VersionNumber"/> <xs:element name="deviation" type="cmn:Description" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType></pre>
String formatting	<p>Specific formatting can apply to some strings of characters.</p> <p>E.g. a permit number is constituted of the 2-character code of the Member State that issued the permit and a numerical part.</p> <p>Note, formatting rules can be dynamic and depend on some other data in the document. E.g. the format of VAT numbers is Member State dependent. The dynamic aspect is not part of this category.</p> <p>Formatting is enforced by a <i>pattern</i> in XML Schema. E.g.</p> <pre data-bbox="491 1536 1121 1637"><xs:simpleType name="AircraftRegistrationNumber"> <xs:restriction base="xs:string"> <xs:pattern value="[A-Z0-9][A-Z0-9\ -]+"> </xs:restriction> </xs:simpleType></pre>

¹ The namespace alias *xs* is consistently used across all files to refer to what belongs to XML Schema; that is, what is in the namespace <http://www.w3.org/2001/XMLSchema>.

Validation category	Description
<p>Multiplicity</p>	<p>Multiplicities specify the number of times an information or group of information can/must be repeated. This notably partly defines what is optional or mandatory in the document (see also “dependent mandatory/optional data” category below).</p> <p>E.g.</p> <ul style="list-style-type: none"> • The multiplicity of the waste catalogue number for a source stream is 0 to 1 as there is at most one. • The multiplicity for the description of activities is 1 to “unbounded” as at least one activity must be described but there is no upper limit. <p>Multiplicities are specified by the <i>minOccurs</i> and <i>maxOccurs</i> attributes on the elements. Note that, when <i>minOccurs</i> or <i>maxOccurs</i> is not specified it defaults to 1. E.g.</p> <pre data-bbox="491 712 1152 779"><xs:element name="deviation" type="cmn:Description" minOccurs="0" maxOccurs="unbounded"/></pre>
<p>Dependent mandatory/optional data</p>	<p>Additionally to constraints expressed by the multiplicities above whether an information or group of information is mandatory or optional (or forbidden) can depend on other data.</p> <p>E.g. choosing whether method A or B is applied for the calculation of emissions of PFC defines the set of (mandatory) factors that must be supplied to perform the calculation.</p> <p>Note, this can also apply to attached documents. E.g. evidence of competencies of laboratory must be provided if it is not ISO accredited.</p> <p>When such constraint exist on an element it is enforced either</p> <ul style="list-style-type: none"> • By defining a specific complex type that explicitly defines what information is expected; e.g. <pre data-bbox="491 1205 1375 1384"><xs:complexType name="SlopePFCEmission"> ... <xs:element name="aeDuration" type="ns:PFCCalculationFactor"/> <xs:element name="aeFrequency" type="ns:PFCCalculationFactor"/> <xs:element name="sEF" type="ns:PFCDefaultableCalculationFactor"/> ...</pre> <ul style="list-style-type: none"> • Or expressed with an XML Schema 1.1 assertion; e.g. for the procedure related to the method for monitoring CO₂ transfers. <pre data-bbox="491 1518 1343 1608"><xs:assert test="(monitoringMethod = 'method_b') = exists(uncertaintyAnalysis)" xpathDefaultNamespace="##targetNamespace"/></pre> <p>Which requires the uncertainty analysis if (and only if) the monitoring method B is applied.</p>

Validation category	Description
Referential integrity	<p>When information refers to some other information (defined elsewhere) the latter must effectively exist.</p> <p>E.g. when providing the data about calculated emissions a reference to the source stream must be given; the latter must be defined in the monitoring plan or, if applicable, in the report itself.</p> <p>Referential integrity is enforced using the <i>key</i> / <i>keyRef</i> mechanism of XML Schema. E.g. for the reference to the source streams from the emission reports or data gap information;</p> <pre data-bbox="491 562 1082 730"> <xs:keyref name="keyRefVariousSourceStream" refer="ns:keySourceStream"> <xs:selector xpath="ns:emission ns:dataGap"/> <xs:field xpath="ns:sourceStreamReference"/> ... </pre> <p>Where <i>keySourceStream</i> is the key given to the source streams (based on their reference attribute),</p> <pre data-bbox="491 831 999 927"> <xs:key name="keySourceStream"> <xs:selector xpath="ns:sourceStream"/> <xs:field xpath="@reference"/> ... </pre>
Enumerated values	<p>Enumerated values constraint the values admissible for a certain data. E.g. the type of an activity must be selected among the list of Annex 1 activities.</p> <p>Enumerated values are defined as XML Schema simple types with restriction on the admissible values. E.g.</p> <pre data-bbox="491 1173 935 1341"> <xs:simpleType name="GHG"> <xs:restriction base="xs:token"> <xs:enumeration value="CO2"/> <xs:enumeration value="N2O"/> <xs:enumeration value="PFC"/> ... </pre>

Assertions

In addition to the above validation principles, XETL includes a series of business rules expressed as XSD 1.1 assertions. They are listed below. For each assertion one can read:

- Its purpose, briefly described in natural language. For instance, “The conversion factor value must be supplied when tier 1 is not applied.”
- The assertion itself, expressed using the XML/XPath syntax. For instance, “(appliedTier != '1') or exists(value)”.
- The type in the schema to which the assertion applied. For instance, ConversionFactor.
- The document type where the assertion exists. For instance, annual report and monitoring plan.

Note that executing the assertions requires a XSD 1.1 validator. To support developers who cannot rely on that version of XSD yet, an XSD 1.0 version of the XETL schemas, stripped from the assertions, is also available.

Assertions for installations

Assertion	XETL type and document
Net Calorific Value is required when an energy based emission is used. <i>(ef/unit = 'tCO2_TJ') = exists(ncv)</i>	CombustionCalculatedEmission (annual report and monitoring plan)
The conversion factor value must be supplied when tier 1 is not applied. <i>(appliedTier != '1') or exists(value)</i>	ConversionFactor (annual report)
The net calorific value value must be supplied when tier 1 is not applied. <i>(appliedTier != '1') or exists(value)</i>	NetCalorificValue (annual report)
The oxidation factor value must be supplied when tier 1 is not applied. <i>(appliedTier != '1') or exists(value)</i>	OxidationFactor (annual report)
The PFC calculation factor value must be supplied when tier 1 is not applied. <i>(appliedTier != '1') or exists(value)</i>	PFCDefaultableCalculationFactor (annual report)
Emission factor unit must be specified when using a default value. <i>exists(defaultValue) = exists(unit)</i>	EmissionFactor (monitoring plan)
Net Calorific Value unit must be specified when using a default value. <i>exists(defaultValue) = exists(unit)</i>	NetCalorificValue (monitoring plan)
If Method B is applied, description of the procedure used for validating the result of method B with method A at least annually is required. <i>(monitoringMethod = 'method_b') = exists(resultValidationProcedure)</i>	PipelineSystems (monitoring plan)
If method B is applied, description of the procedure used for determining fugitive emissions is required. <i>(monitoringMethod = 'method_b') = exists(fugitiveProcedure)</i>	PipelineSystems (monitoring plan)
If method B is applied, description of the procedure used for determining vented emissions is required. <i>(monitoringMethod = 'method_b') = exists(ventedEmissionsProcedure)</i>	PipelineSystems (monitoring plan)
If method B is applied, uncertainty analysis is required. <i>(monitoringMethod = 'method_b') = exists(uncertaintyAnalysis)</i>	PipelineSystems (monitoring plan)
If method B is applied, description of equipment used for temperature and pressure measurement is required. <i>(monitoringMethod = 'method_b') or exists(temperatureOrPressureMeasurementInstrument)</i>	PipelineSystems (monitoring plan)

Assertion	XETL type and document
Procedure used for determining stock piles at end of year is required for batch determination method. <i>(activityDataDeterminationMethod = 'batch') = exists(endOfYearStockPilesProcedure)</i>	SourceStreamEmissionCalculation (monitoring plan)
The relation with the trade partner must be described if (and only if) the instrument is not under control of the operator. <i>(instrumentSupervisor = 'trade_partner') = exists(tradePartnerRelation)</i>	ActivityDataMeasurementInstrument (common installation types)
Lower end of range must be lower (or equal) to upper end. <i>lowerEnd &lt;= upperEnd</i>	MeasurementRange (common installation types)
The identifier or the name of the installation must be provided (or both). <i>identifier or name</i>	RemoteInstallation (common installation types)

Assertions for aircraft operators

Assertion	XETL type and document
The number of flights must be reported for each of the 3 four-month period when (and only when) a simplified approach is followed. <i>(simplifiedProcedure = true() and count(flightNumberPerPeriod) = 3) or (simplifiedProcedure = false() and count(flightNumberPerPeriod) = 0)</i>	ReportRawData (annual emissions report)
The on-board equipment uncertainty must be provided if (and only if) on-board equipment is used to measure fuel uplifts and the quantity remaining in the tank. <i>(fuelUpliftDataSource = 'onboard_equipment') = exists(onboardEquipmentUncertainty)</i>	ActivityDataCalculation (annual emission monitoring plan)
Only low emitters can opt for a simplified procedure. <i>lowEmitter = true()</i>	SimplifiedProcedureMonitoringPlan (annual emission monitoring plan)
Risk assessment is applicable for small emitters who do not intend to use the “small emitters tool”. <i>(consumptionEstimationTool/ao:standardTool = 'small_emitters_tool') != exists(riskAssessment)</i>	SimplifiedProcedureMonitoringPlan (annual emission monitoring plan)
Consumption estimation tool either refers to a tool from the approved list or provide the name of an alternate tool. <i>exists(standardTool) != exists(otherToolName)</i>	ConsumptionEstimationTool (common aircraft operator types)

Assertion	XETL type and document
The country and Member State must be different for a non-domestic flight. <i>not(exists(nonDomesticFlightEndpoint)) or (nonDomesticFlightEndpoint/country != memberState)</i>	FlightEndpoints (common aircraft operator types)
A name must be provided in case of alternate fuel. <i>type != 'other_alternative_fuel' or name</i>	Fuel (common aircraft operator types)
If a unique ICAO designator is not available, the identification for ATC purposes (tail numbers) of all the aircraft operated as used in box 7 of the flight plan must be provided. <i>exists(descriptor) != exists(aircraftRegistrationNumber)</i>	Operator (common aircraft operator types)

Common assertions

Assertion	XETL type and document
For WGS84 latitude must be between -90° and +90° and longitude between -180° and +180°. <i>datum != 'wgs84' or (coordinate[1] ge -90 and coordinate[1] le 90 and coordinate[2] ge -180 and coordinate[2] le 180)</i>	Coordinates (common types)
Evidence must only be provided by non EN ISO/IEC 17025 accredited laboratories. <i>iso17025Accredited = not(exists(evidence))</i>	Laboratory (common types)

Schema namespace and versioning

XML namespaces are used to reflect the organisation of the schemas and to distinguish types having the same name in different schema files. XETL namespace names are given a structure that directly reflects the architecture of the XML schema files, based on the type of operator and the type of document:

eu:europa:ec:clima:xetl:version[:operator_type[:document_type:[ao_sub_type]]]

with

version = major version number

operator_type = inst or ao

document_type = mp, report, verification or improvement

ao_sub_type = ae or tkm, only when operator type = ao and document type = mp or report.

The table below lists the main namespaces.

Operator / Document type	Stationary installation	Aircraft operator
Monitoring plan	eu:europa:ec:clima:xetl:0:inst:mp	eu:europa:ec:clima:xetl:0:ao:mp:ae eu:europa:ec:clima:xetl:0:ao:mp:tkm

Annual emissions report	eu:europa:ec:clima:xetl:0:inst:report	eu:europa:ec:clima:xetl:0:ao:report:ae eu:europa:ec:clima:xetl:0:ao:report:tkm
Improvement report	eu:europa:ec:clima:xetl:0:inst:improvement	eu:europa:ec:clima:xetl:0:ao:improvement
Verification report	eu:europa:ec:clima:xetl:0:inst:verification	eu:europa:ec:clima:xetl:0:ao:verification

Schema versioning is linked with the schema namespace and the major version number it includes.

Non-backward compatible change

When a major, non-backward compatible change will be made to a schema file, the version ID in the namespace will be incremented.

A non-backward compatible change is one that is not structurally compatible with the previous version: adding elements or types, changing elements or types in non-compatible ways: e.g. adding child elements, going from optional to mandatory, changing the order of elements.

Changing the namespace in such cases is a sure way to invalidate all existing documents that make use of a previous version.

Backward compatible change

When a minor, backward compatible change is made to a schema file the schema namespace must not change as a document valid under the new version remains valid with the new version. Typical examples of backward-compatible changes are: adding optional elements or attributes, adding enumeration values to a type, making a pattern less strict, going from mandatory to optional.

A “minor” version numbering is provided by the *schemaVersion* attribute to the BaseDocument element serving as the root from which all main elements are derived.

When a new, backward compatible version will be created, a value will be added to the *schemaVersion* enumeration. Software manipulating instance documents are then able to distinguish between minor versions by reading the *schemaVersion* attribute in the document root element.

Further information

For further information about the EU ETS Reporting Language, please contact the European Commission's Directorate-General for Climate Action