New MAA XML Schema

An overview

MARCH 2019

Contents

Contents

CESSP Phase 1 -Essential information

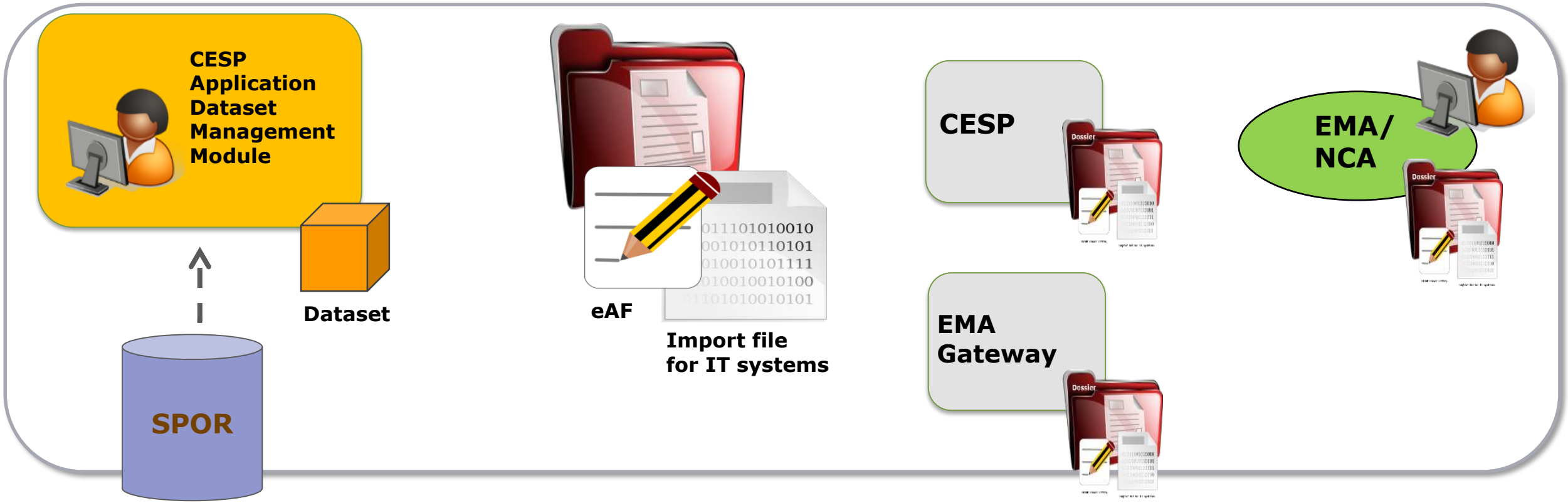
This implementation project step, CESSP Phase 1, will deliver an online web-user interface to make use of integrated human and veterinary Marketing Authorisation Application datasets for new marketing authorisation and extension applications.

The system will provide data from SPOR (OMS, RMS and Substances). The CESP Dataset Module will make it possible to re-use the application form content.

CESSP Phase 1 is planned to be followed by the addition of the Variation and Renewal form to the CESP Dataset Module.

The current eAFs will be withdrawn after transition periods and the use of the new CESP Dataset Module will become mandatory at a future date which will be communicated in advance.

Future process after implementation of CESSP Phase 1

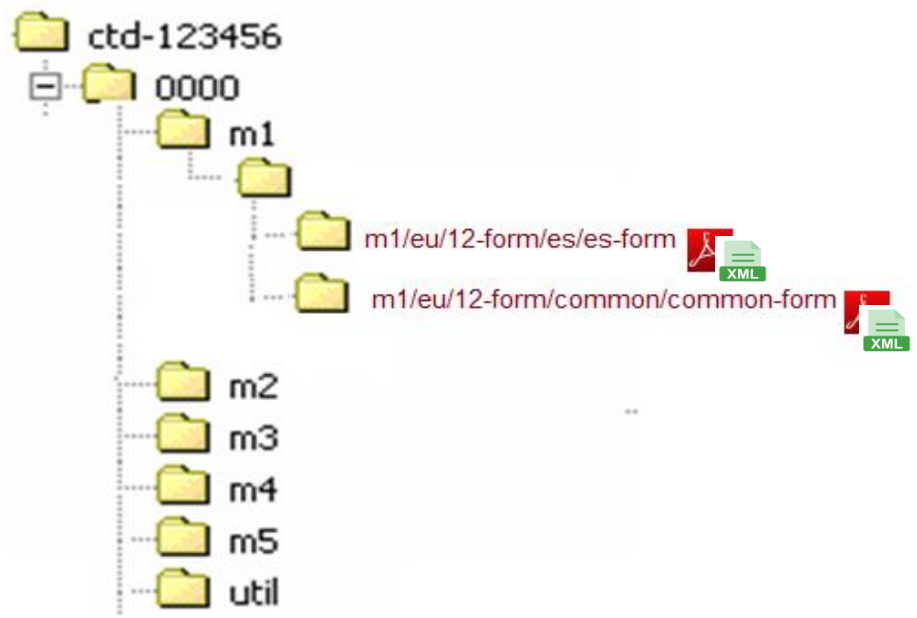


Start and Finalise Dataset (eAF etc.) → **Finalise dossier submission** → **Submit** → **Process submission**

CESP Submissions

eCTD/VNeeS Sequence

CESP



To load Data
National Regulatory Database

Huma Database
Vet Database

To load Documents
Docs Mangement System

eCTD Repository
NeeS Repository
VNeeS Repository

Training objectives

The webinar will focus on utilising the upgraded XML schema of the initial application form (eAF initial) to automise application data import into IT systems

The refactored XML schema of the eAF for initial applications (vet + human) will be described and explained. The new XML schema will come into place at the end of 2019.

Learning objectives

At the end of the course students will be able:

- To explain the reasons why the XML schema needed an upgrade
- To understand the new schema definition

CESP Dataset Module Project Information

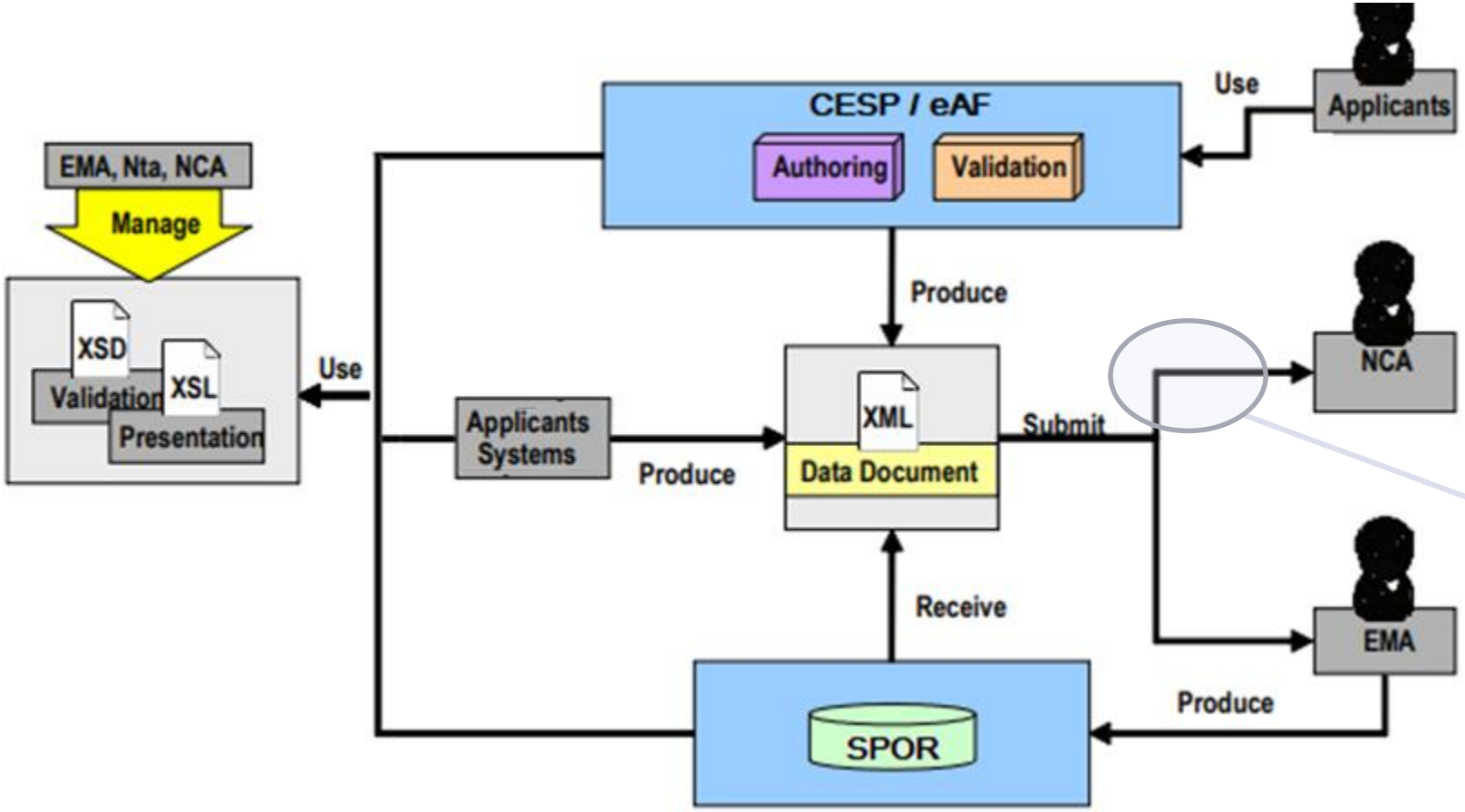
Current Data Exchange Standard on the eAF website

http://esubmission.ema.europa.eu/eaf/

Upgraded Data Exchange Standard:
http://esubmission.ema.europa.eu/cessp/cessp.htm

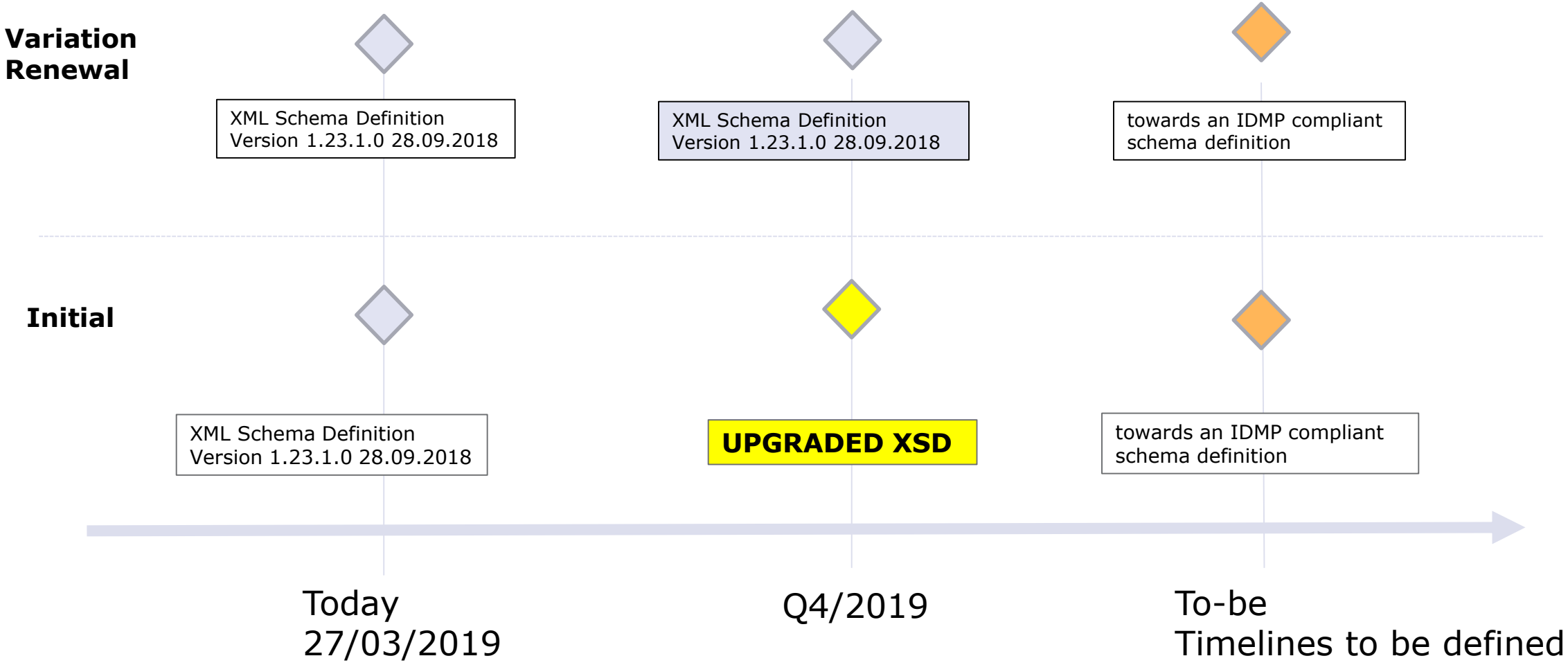
Contact us: cesspproject@hma.eu

High-level View



Import Tools Using the upgraded XML

Roadmap Schema Development



Contents

Current MAA schema problems / Solutions

What we saw in the DES XML specification	What we have done
The DES XML node order does not reflect the order of fields in the form. **Example(s):** The node *maa:scientific-advice* (Section 3) appears in the document before the node *maa:declaration* (First section in the PDF)	Reorganized the order of the nodes in the XML document so that it matches the order visually perceived in the PDF document
Some nodes in XML DES have names which corresponds to the UI control that allows its edition and does not reflect their business meaning. **Example(s):** Nodes *maa:yes*, *rdm:selected* or *maa:Manu-device-checkbox*	Changed the name of these nodes, so that it relates to the business meaning of the data it contains.
There are cases of high-level nodes that have the most part of its children nodes empty and cannot be filled from the PDF, resulting in an unnecessary overhead in size and complexity of the XML document. **Example(s):** The node *maa:contact-pharmaco-vigilance* has its children nodes *rdm:admin-office* and *rdm:manu-facility* with all their descendants empty with no possibility of being filled via the PDF form.	Removed unnecessary nodes (i.e.: all the nodes that cannot be filled using the PDF form)

Problems / Solutions (Cont.)

What we saw in the DES XML specification	What we did
Attributes of some XML nodes are equal to "" and cannot be modified from the PDF. **Example(s):** Attribute *is_significant-benefit* of node *maa:orphan-designation*	Removed all attributes in the XML. Plain (scalar) data are just contained as node values (i.e.: "<node>data</node>")
There are some attributes, not related to the business but with the adobe technology , defined in the XFA schema and with namespace *http://www.xfa.org/schema/xfa-data/1.0/*. **XFA** is a proprietary family of XML specifications (see Wikipedia) used also by Adobe. **Example(s):** Attribute *xfa:APIVersion*	Removed all attributes in the XML, hence also proprietary ones
Node naming conventions are not uniform. Sibling nodes are not named following the same rules and notation. **Example(s):** • Node *maa:centralised-procedure* corresponds to section 1.1.1 of the PDF • Node *maa:section1-5-1* (sibling of the above) corresponds to section 1.5.1	Named all nodes consistently. Children of *form* node have names that reflect corresponding section number in the PDF document. All nodes of lower level have been given names that reflect their business meaning: • *"section-1"* for Section 1 node • *"centralisedProcedure"*, for Section 1.1 node

Problems / Solutions (Cont.)

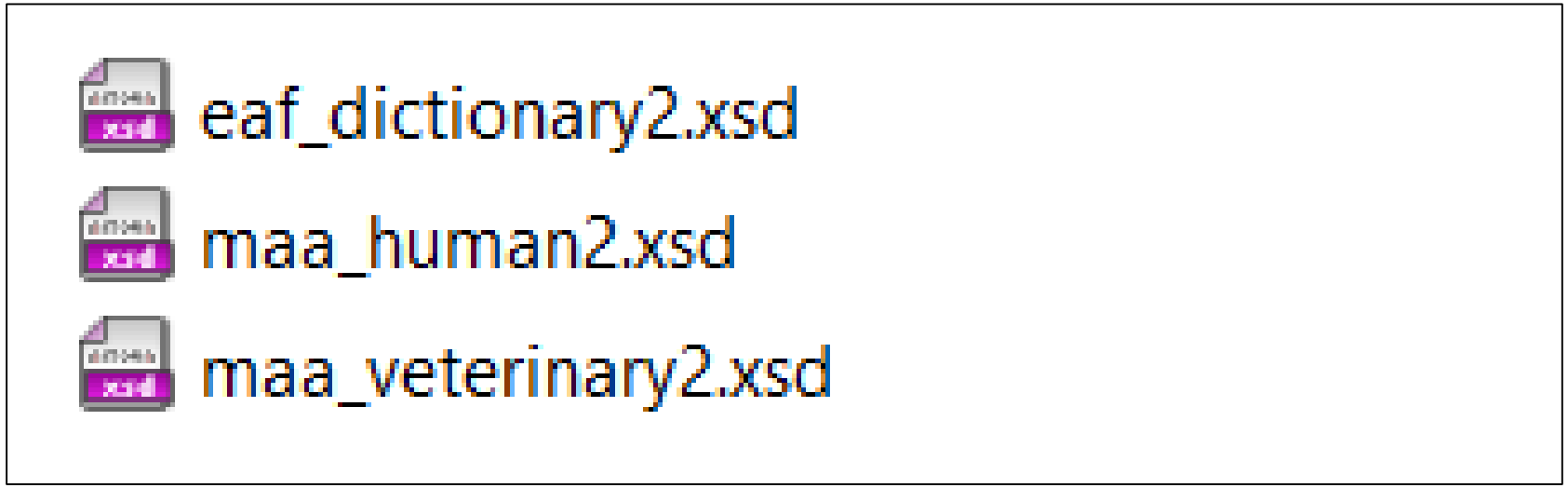
What we saw in the DES XML specification	What we did
Nodes aren't named consistently. Different use of lower/uppercase characters, uderscores ("_") and dashes ("-") in node names. **Example(s):** Nodes *maa:subject_to_prescription*, *maa:Device-identification*, *rdm:loc-modifiedDate*, *maa:not-subject-medical-prescription*	Adopted Camel case naming convention (Wikipedia) uniformely, while preserving DES names, where possible. Dash ("-") character is used only in section names and annex- names, in order to separate contained numbers. **Examples:** *subjectToPrescription, deviceIdentificarion, notSubjectMedicalPrescription, annex-2*
Repeatable nodes of different type share the same parent. **Example(s):** Nodes of type *maa:Manufacture-contact-details* (repeatable – Section 2.2.4.2) are siblings of nodes *maa:Device-identification* (repeatable too – Section 2.2.4.1)	Each collection of repeatable nodes is included in a node having same name, but "pluralized" and with "Collection" suffix **Example:** *manufacturerContactDetailsCollection* node contains a collection of *manufacturerContactDetail* nodes

Problems / Solutions (Cont.)

What we saw in the DES XML specification	What we did
Some nodes are used improperly. **Example(s):** If *"Vaccine antigen master file"* option is selected in the PDF, a *rdm:vamf* node is generated with all its subnodes empty except one (*rdm:is-vamf-issued* with a value of "1"). Data of all VAMF entries are located in following *rdm:vamf* (first entry in second node, second entry in third and so on), so that reading of first node has to be skipped in order to retrieve data.	Created a node called *isVamfIssued* wich indicates whether the option is selected or not. The node has a sibling called *vamfs* that contains the collection of *vamf* nodes, one corresponding to each real VAMF entry.

Contents

New schema structure overview

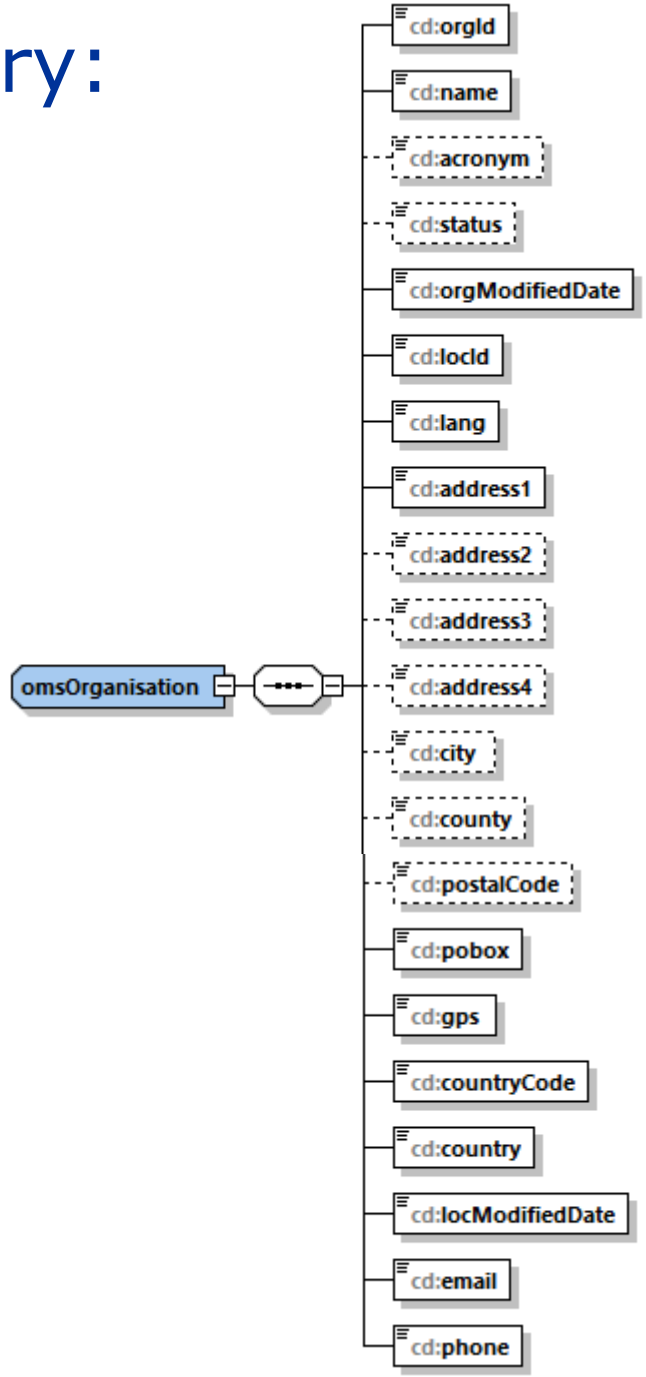


Remarks:

- Three XSD schemas:
  - **maa_human2.xsd** (eAF of **human** domain)
  - **maa_veterinary2.xsd** (eAF of **veterinary** domain)
  - **eaf_dictionary.xsd** (**common** dictionary e.g. Procedure type with type definitions for **both domains**)

- Version should be **2.0** (no backward compatibility with 1.2x.* versions)

Dictionary: OMS




```
<xs:complexType name="omsOrganisation">
    <xs:sequence>
        <xs:element name="orgId" type="xs:string"/>
        <xs:element name="name" type="xs:string"/>
        <xs:element name="acronym" type="xs:string" minOccurs="0"/>
        <xs:element name="status" type="xs:string" minOccurs="0"/>
        <xs:element name="orgModifiedDate" type="xs:date"/>
        <xs:element name="locId" type="xs:string"/>
        <xs:element name="lang" type="xs:string"/>
        <xs:element name="address1" type="xs:string"/>
        <xs:element name="address2" type="xs:string" minOccurs="0"/>
        <xs:element name="address3" type="xs:string" minOccurs="0"/>
        <xs:element name="address4" type="xs:string" minOccurs="0"/>
        <xs:element name="city" type="xs:string" minOccurs="0"/>
        <xs:element name="county" type="xs:string" minOccurs="0"/>
        <xs:element name="postalCode" type="xs:string" minOccurs="0"/>
        <xs:element name="pobox" type="xs:string"/>
        <xs:element name="gps" type="xs:string"/>
        <xs:element name="countryCode" type="xs:string"/>
        <xs:element name="country" type="xs:string"/>
        <xs:element name="locModifiedDate" type="xs:date"/>
        <xs:element name="email" type="xs:string"/>
        <xs:element name="phone" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
```

Dictionary: RMS

```
<xs:complexType name="pharmaceuticalFormsCollectionType">
    <xs:sequence>
        <xs:element name="pharmaceuticalForm" type="cd:rmsTerm" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
```


```
<xs:complexType name="rmsTerm">
    <xs:sequence>
        <xs:element name="listId" type="xs:string"/>
        <xs:element name="termId" type="xs:string"/>
        <xs:element name="name" type="xs:string"/>
        <xs:element name="shortName" type="xs:string" minOccurs="0"/>
        <xs:element name="symbol" type="xs:string" minOccurs="0"/>
        <xs:element name="version" type="xs:integer"/>
    </xs:sequence>
</xs:complexType>
```

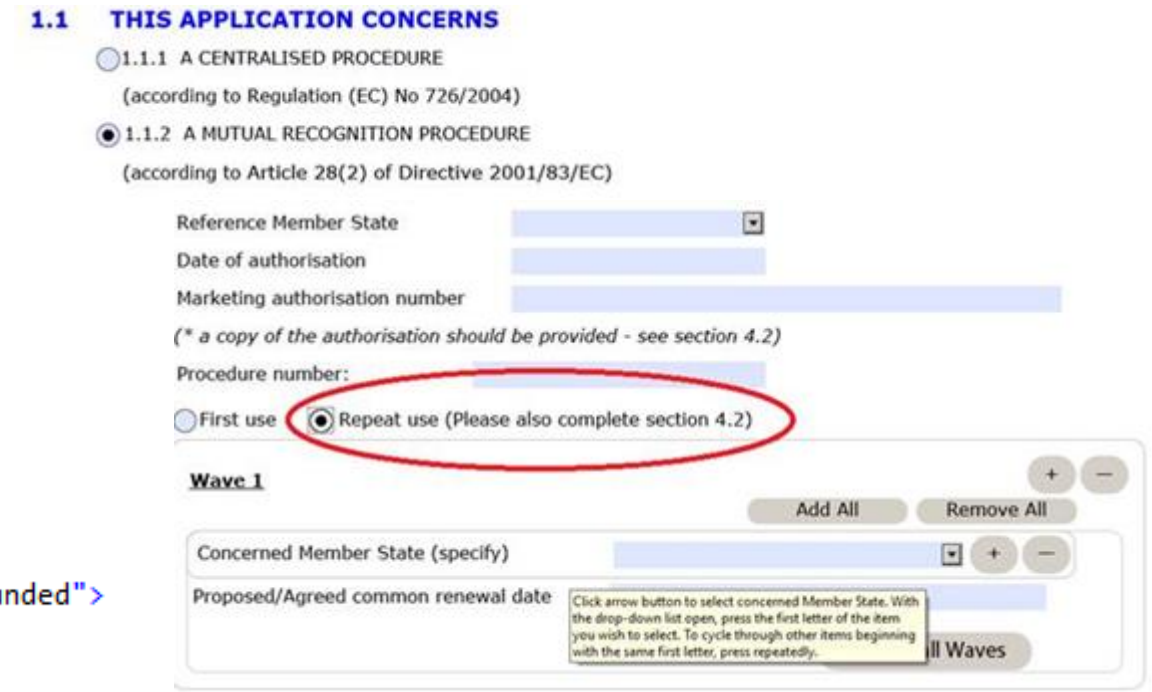
Dictionary: enumerations

```
<xs:simpleType name="legalBaseHumanType" final="restriction">
    <xs:restriction base="xs:string">
        <xs:enumeration value="article-8-3"/>
        <xs:enumeration value="article-10-1"/>
        <xs:enumeration value="article-10-3"/>
        <xs:enumeration value="article-10-1And10-3"/>
        <xs:enumeration value="article-10-4"/>
        <xs:enumeration value="article-10a"/>
        <xs:enumeration value="article-10b"/>
        <xs:enumeration value="article-10c"/>
        <xs:enumeration value="article-16a"/>
    </xs:restriction>
</xs:simpleType>
```

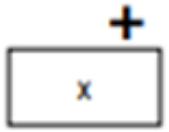
- advancedTherapyMedicinalProductHumanType
- annexHumanType
- annexVeterinaryType
- applicationConsiderationHumanType
- applicationConsiderationVeterinaryType
- authorisationGrantedByType
- changeType
- functionType
- legalBaseHumanType
- legalBaseVeterinaryType
- multipleApplicationSubmittedType
- orphanMedicinalProductDesignationStatusType
- procedureScopeHumanType
- procedureScopeVeterinaryType
- proposedAdministrationType
- qualitativeChangeDetailType
- standardTimeUnitsType

Dictionary: type definitions

```
<xs:complexType name="mutualRecognitionProcedureType">
    <xs:sequence>
        <xs:element name="referenceMemberState" type="cd:rmsTerm"/>
        <xs:element name="dateOfAuthorisation" type="xs:date"/>
        <xs:element name="marketAuthorisationNumber" type="cd:stringMax30"/>
        <xs:element name="procedureNumber" type="cd:stringMax50"/>
        <xs:element name="usage" type="cd:usageType"/>
        <xs:choice>
            <xs:element name="memberRenewal" type="cd:memberRenewalType"/>
            <xs:element name="wavesCollection">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="wave" minOccurs="1" maxOccurs="unbounded">
                            <xs:complexType>
                                <xs:sequence>
                                    <xs:element name="memberRenewal" type="cd:memberRenewalType"/>
                                </xs:sequence>
                            </xs:complexType>
                        </xs:element>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:choice>
        <!-- only veterinary -->
        <xs:element name="waiverOrAmendmentDetails" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:assert test="(cd:usage = 'repeat' and boolean(cd:wavesCollection)) or (cd:usage = 'first' and boolean(cd:memberRenewal))"/>
</xs:complexType>
```

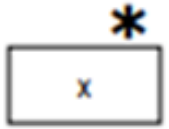


What can be validated through the schema?



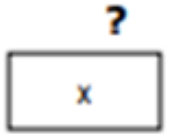
element x is mandatory and repeatable (1..n)

```
minOccurs="1" maxOccurs="unbounded"
```



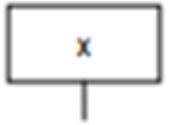
element x is optional and repeatable (0..n)

```
minOccurs="0" maxOccurs="unbounded"
```



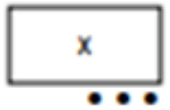
element x is optional

```
minOccurs="0"
```



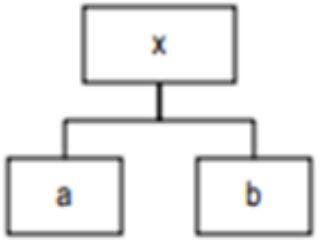
element x has the following content model

```
<xs:sequence>
    <xs:element....
    <xs:element.....
```



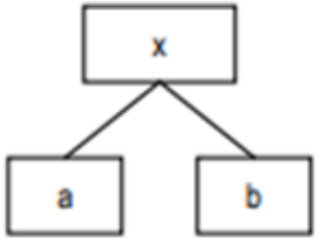
elements are defined elsewhere

```
<xs:import namespace
```



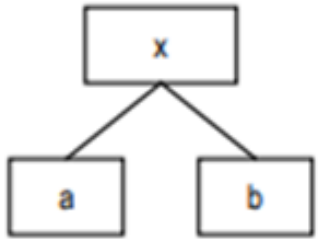
element x contains elements a and b

```
<xs:sequence>
    <xs:element...  minOccurs="1"
    <xs:element ... minOccurs="1"
```



element x contains either element a or b

```
<xs:choice>
      <xs:element ... minOccurs="0"/>
      <xs:element ... minOccurs="0"/>
</xs:choice>
```



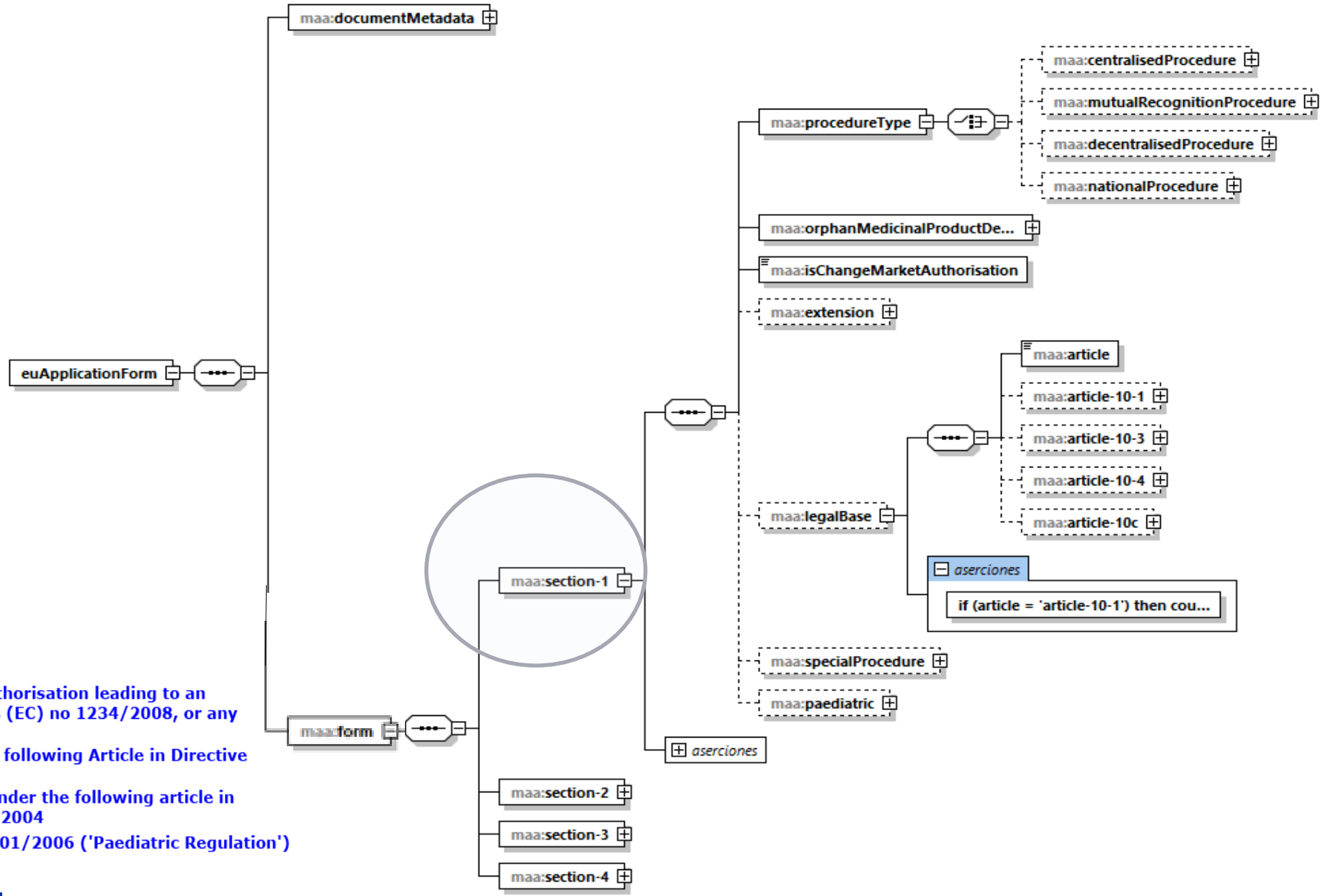
Conditions among elements depending on their values

```
<xs:complexType>
    <xs:sequence>
        <xs:element name="field1" .... minOccurs="0"/>
        <xs:element name="field2" .... minOccurs="0"/>
    </xs:sequence>
    <xs:assert test="field2 or not(field2 = 'value')"/>
```

Max length of a string

```
type="cd:stringMax50"
```

MAA human Section 1



1. TYPE OF APPLICATION

1.1 This application concerns

1.2 Orphan medicinal product information

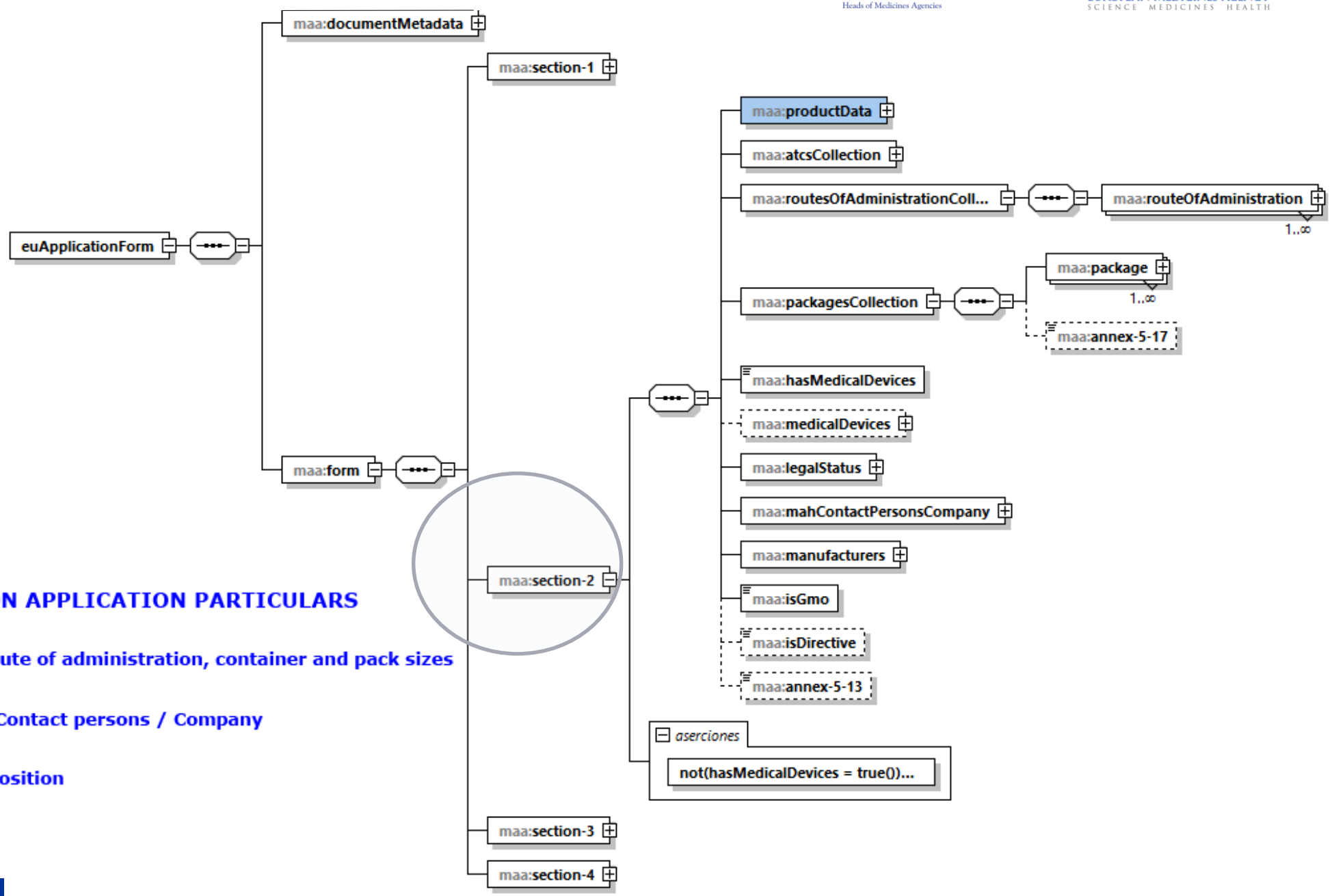
1.3 Application for a change to existing marketing authorisation leading to an extension as referred to in Annex I of Regulations (EC) no 1234/2008, or any national legislation, where applicable

1.4 This application submitted in accordance with the following Article in Directive 2001/83/EC

1.5 Consideration of this application also requested under the following article in Directive 2001/83/EC or Regulation (EC) N° 726/2004

1.6 Requirements according to Regulation (EC) No 1901/2006 ('Paediatric Regulation')

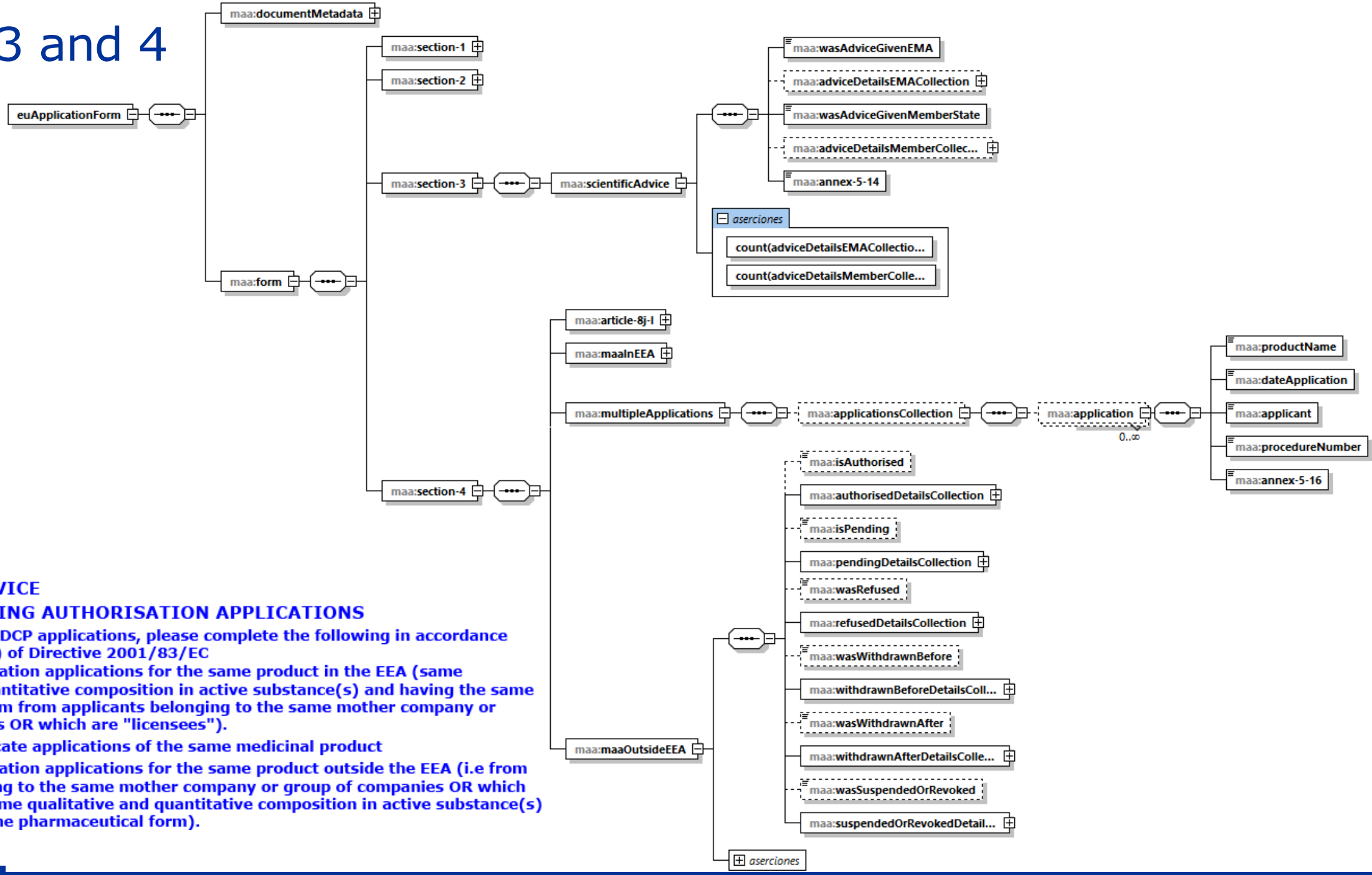
Section 2



2. MARKETING AUTHORISATION APPLICATION PARTICULARS

- **2.1 Name(s) and ATC code**
- **2.2 Strength, pharmaceutical form, route of administration, container and pack sizes**
- **2.3 Legal status**
- **2.4 Marketing authorisation holder / Contact persons / Company**
- **2.5 Manufacturers**
- **2.6 Qualitative and quantitative composition**

Sections 3 and 4



3. SCIENTIFIC ADVICE

4. OTHER MARKETING AUTHORISATION APPLICATIONS

4.1 For National/MRP/DCP applications, please complete the following in accordance with Article 8(j)-(l) of Directive 2001/83/EC

4.2 Marketing authorisation applications for the same product in the EEA (same qualitative and quantitative composition in active substance(s) and having the same pharmaceutical form from applicants belonging to the same mother company or group of companies OR which are "licensees").

4.3 For multiple/duplicate applications of the same medicinal product

4.4 Marketing authorisation applications for the same product outside the EEA (i.e from applicants belonging to the same mother company or group of companies OR which are "licensees". Same qualitative and quantitative composition in active substance(s) and having the same pharmaceutical form).

New schema structure overview (Cont.)

```
<xs:element name="form">
    <xs:complexType>
        <xs:sequence>
            <!-- section 1 start -->
            <xs:element name="section-1">...</xs:element>
            <!-- section 1 end -->
            <!-- section 2 start -->
            <xs:element name="section-2">...</xs:element>
            <!-- section 2 end -->
            <!-- section 3 start -->
            <xs:element name="section-3">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="scientificAdvice">
                            <xs:complexType>
```

Remarks:

- First level children of *"form"* have names that reflect their position in current PDF structure

New schema structure overview (Cont.)

```
<xs:element name="euApplicationForm">
    <xs:complexType>
        <xs:sequence>
            <!-- meta data start -->
            <xs:element name="documentMetadata" type="cd:documentMetadataType"/>
            <!-- meta data end -->
            <!-- form data start -->
            <xs:element name="form">...</xs:element>
            <!-- form data end -->
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

Remarks:

- Document has root node *"euApplicationForm"* with two children:
  - *documentMetadata:* domain, version, creation date
  - *form:* the eAF data
- Only two namespaces:
  - maa:http://www.eaf.com/maa/ (instances data namespace)
  - cd:http://www.eaf.com/dictionary/ (dictionary namespace)
- No external namespaces (~~xfa=http://www.xfa.org/schema/xfa-data/1.0/~~)

New schema structure overview (Cont.)

```
<!-- section 1-2 orphan medicinal product designation start -->
<xs:element name="orphanMedicinalProductDesignation">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="hasBeenApplied" type="xs:boolean"/>
            <!-- section 1-2-1 procedures start -->
            <xs:element name="proceduresCollection" minOccurs="0">
                <xs:complexType>
```

Remarks:

- Enforced consistent naming conventions (use of *Camel Case* notation)
- Boolean nodes (yes/no) have names that suggest their nature (begin with “has” “is” “was”,…)
- Used XSD schema to define/enforce:
  - Data types
  - Be mandatory/optional
  - Cardinality

New schema structure overview (Cont.)

```
<!-- section 1-2-2 market exclusivity start -->
<xs:element name="hasOrphan" type="xs:boolean" minOccurs="0"/>
<xs:element name="procedureNumbersCollection" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="procedureNumber" type="cd:stringMax50" maxOccurs="u
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="hasGrantedMarketAuthorisation" type="xs:boolean" minOccurs="0"/
<xs:element name="referenceProductsCollection">
```

Remarks:

- Collections of repeteable nodes always contained in an **exclusive** parent
- Parent name of nodes *xxx* is named *xxxsCollection* (plural form + "Collection" suffix)
- **exclusive** = no other nodes contained in it

New schema structure overview (Cont.)

```
<!-- section 1-1 procedure type start -->
<xs:element name="procedureType">
    <xs:complexType>
        <xs:choice>
            <xs:element name="centralisedProcedure" type="cd:centralisedProcedureType" minOccurs="0"/>
            <xs:element name="mutualRecognitionProcedure" type="cd:mutualRecognitionProcedureType" minOccurs="0"/>
            <xs:element name="decentralisedProcedure" type="cd:decentralisedProcedureType" minOccurs="0"/>
            <xs:element name="nationalProcedure" type="cd:nationalProcedureType" minOccurs="0"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- section 1-1 procedure type end -->
```

Remarks:

- Children of 2nd level (or higher) have names that reflect their business meaning (e.g.: *centralizedProcedure*)
- Use of *xs:choice* element to limit document size and avoid inconsistencies (if it is a *centralizedProcedure*, no *nationalProcedure* contents exist)
- Use of types and deferred node definitions (in the dictionary)

New schema structure overview (Cont.)

```
<xs:element name="medicinalProductsWhereBioequivalence">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="medicinalProduct" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="bioStyRefEudractNumber" type="cd:stringMax30" minOccurs="0"/>
                        <xs:element name="inventedName" type="cd:stringMax250" minOccurs="0"/>
                        <xs:element name="pharmaceuticalForms" type="cd:rmsTerm"/>
                        <xs:element name="marketingAuthorisations" type="cd:marketingAuthorisationsTypeWithDate"/>
                        <xs:element name="authorisationGrantedBy" type="cd:authorisationGrantedByType" minOccurs="0"/>
                        <xs:element name="authorisationMemberState" type="cd:rmsTerm" minOccurs="0"/>
                        <xs:element name="memberStateOfSource" type="cd:rmsTerm" minOccurs="0"/>
                    </xs:sequence>
                    <xs:assert test="authorisationMemberState or not(authorisationGrantedBy = 'memberState')"/>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

Remarks:

- Use of *xs:assert* elements to enforce complex business rules compliance (**Note:** XSD schema **version 1.1** used)
- Assertions defined after element fields enumeration, so the element definition is always self-contained

MAA Human XML example

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2019 (http://www.altova.com)-->
<maa:euApplicationForm xmlns:maa="http://www.eaf.com/maa/" xmlns:cd="http://www.eaf.com/dictionary/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.eaf.com/maa/ maa_human2.xsd">
    <maa:documentMetadata>...</maa:documentMetadata>
    <maa:form>
        <maa:section-1>
            <maa:procedureType>
                <maa:centralisedProcedure>...</maa:centralisedProcedure>
            </maa:procedureType>
            <maa:orphanMedicinalProductDesignation>...</maa:orphanMedicinalProductDesignation>
            <maa:extension>...</maa:extension>
            <maa:legalBase>
                <maa:article>article-10c</maa:article>
                <maa:article-10-1>
                    <maa:medicinalProductsLessThanTenYearsCollection>...</maa:medicinalProductsLessThanTenYearsCollection>
                    <maa:medicinalProductsWhereApplicationMadeCollection>...</maa:medicinalProductsWhereApplicationMadeCollection>
                    <maa:medicinalProductsWhereBioequivalenceCollection>...</maa:medicinalProductsWhereBioequivalenceCollection>
                </maa:article-10-1>
                <maa:article-10-3>...</maa:article-10-3>
                <maa:article-10-4>
                    <maa:medicinalProductsLessThanTenYearsCollection>...</maa:medicinalProductsLessThanTenYearsCollection>
                    <maa:medicinalProductsWhereApplicationMadeCollection>...</maa:medicinalProductsWhereApplicationMadeCollection>
                    <maa:medicinalProductsComparabilityTestsCollection>...</maa:medicinalProductsComparabilityTestsCollection>
                </maa:article-10-4>
                <maa:article-10c>...</maa:article-10c>
            </maa:legalBase>
            <maa:specialProcedure>...</maa:specialProcedure>
            <maa:paediatric>...</maa:paediatric>
        </maa:section-1>
        <maa:section-2>...</maa:section-2>
        <maa:section-3>
            <maa:scientificAdvice>...</maa:scientificAdvice>
        </maa:section-3>
        <maa:section-4>...</maa:section-4>
    </maa:form>
</maa:euApplicationForm>
```

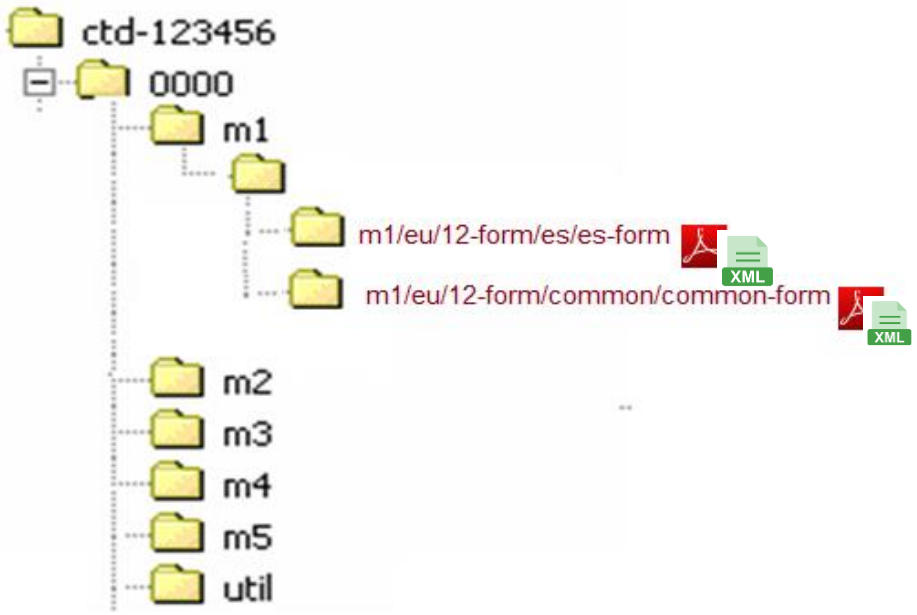
To be published on eSubmission Website

Contents

Information Uploading with the new Schemas

eCTD/VNeeS Sequence

CESP



To load
National Regulatory Database

Human Database
Vet Database

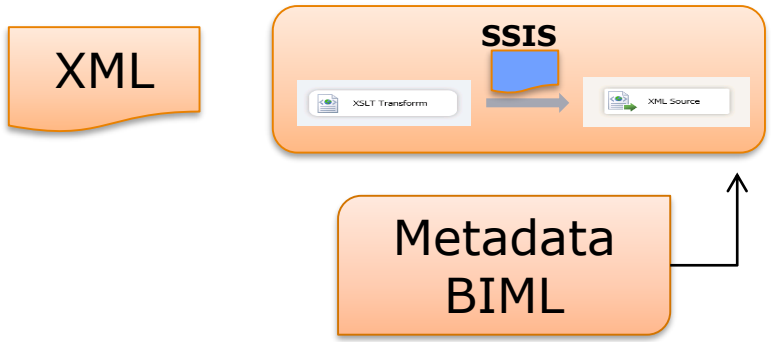
Update the data uploading according to the new XMLs

To load
Document Mangement System

eCTD Repository
NeeS Repository
VNeeS Repository

Uploading strategies

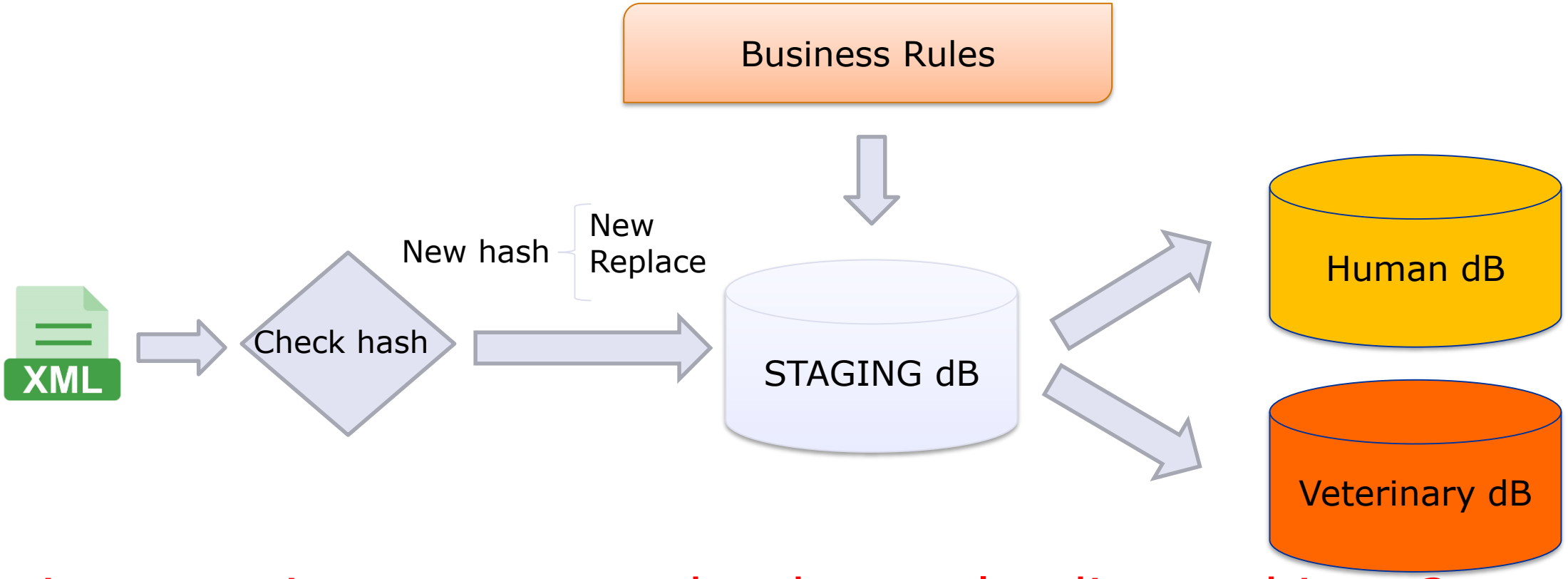
- ETLs (Extract, Transform and Load) Tools



- Ad hoc development (Java, .NET)

Decoupled uploading

Transferring data from the xml to a staging area and finally to the business databases.



Is it convenient to repeat the data uploading webinar ?
please contact cesspproject@hma.eu

Questions?

Georg Neuwirther (georg.neuwirther@ages.at)
Gary Wilson (gary.wilson@ext.hpra.ie)
JM Simarro (jsimarro@aemps.es)