



This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.

Attribution: Jean Vieille

Work: **ISA8895 Implementation**
Section: **Interoperability**
Chapter: **B2MML**
Exercise
Language: **Français**
Version: **V3 - 05/2011**



Jean Vieille

www.syntropicfactory.info j.vieille@syntropicfactory.info



Research community www.controlchainmanagement.org



Consulting group www.controlchaingroup.com



Exposé général du problème

- **Prendre en compte les extensions nécessaire au type B2MML “Equipment”**
 - Les extensions proviennent de 3 sources:
 - Extensions propres à l'entreprise concernée
 - Extensions des fournisseurs de système Company1 et Company2
 - Les extensions fournisseurs sont traitées dans des schémas séparés
 - Ces fichiers sont importés dans le fichier B2MML Extensions, seul schéma d'extension connu par les schémas standards B2MML qui ne doivent pas être modifiés
 - Il y a des conflits de noms entre ces extensions:
 - *Company1 – définit SpeedRotation comme string*
 - *Company2 – définit SpeedRotation comme integer*
 - Des espaces de nom additionnels sont nécessaires, un par fournisseur
 - Les squelettes de fichiers sont fournis,
 - Compléter les fichiers
 - Vérifier le résultat dans le schéma B2MML Equipment
- **Générer un document XML basé sur le schéma Equipment avec des valeurs pour les extensions définies**

Exposé du problème – Fichiers nécessaires

- **B2MML Core schemas**

- B2MML-V0300d1-Common.xsd
- B2MML-V0300d1-Equipment.xsd

Nota: ne doivent jamais être modifiés

- **B2MML Extension schema**

- B2MML-V0300d1-Extensions.xsd

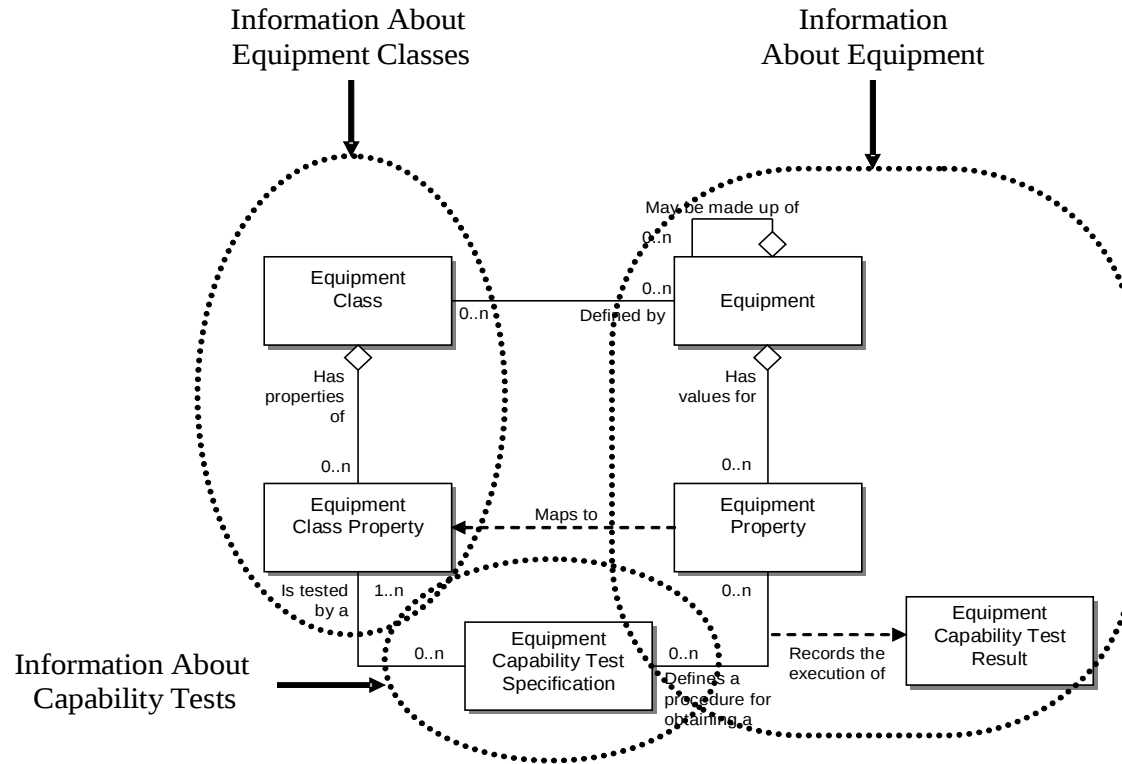
- **Fichiers d'extension B2MML des fournisseurs**

- B2MML-V0300d1-ExtensionsCompany1.xsd
- B2MML-V0300d1-ExtensionsCompany2.xsd

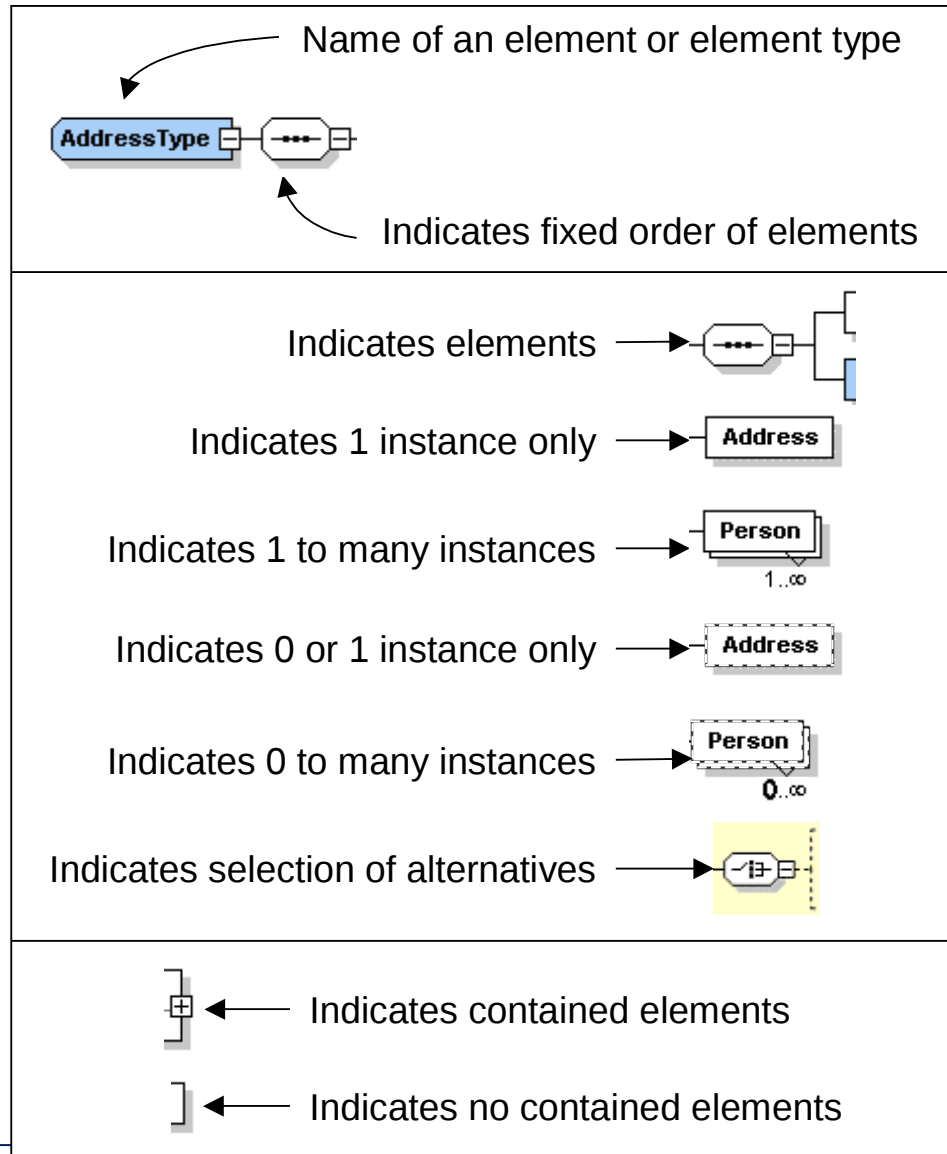
- **Echantillon de documents XML**

- B2MML-V0300d1-Example 1.xml

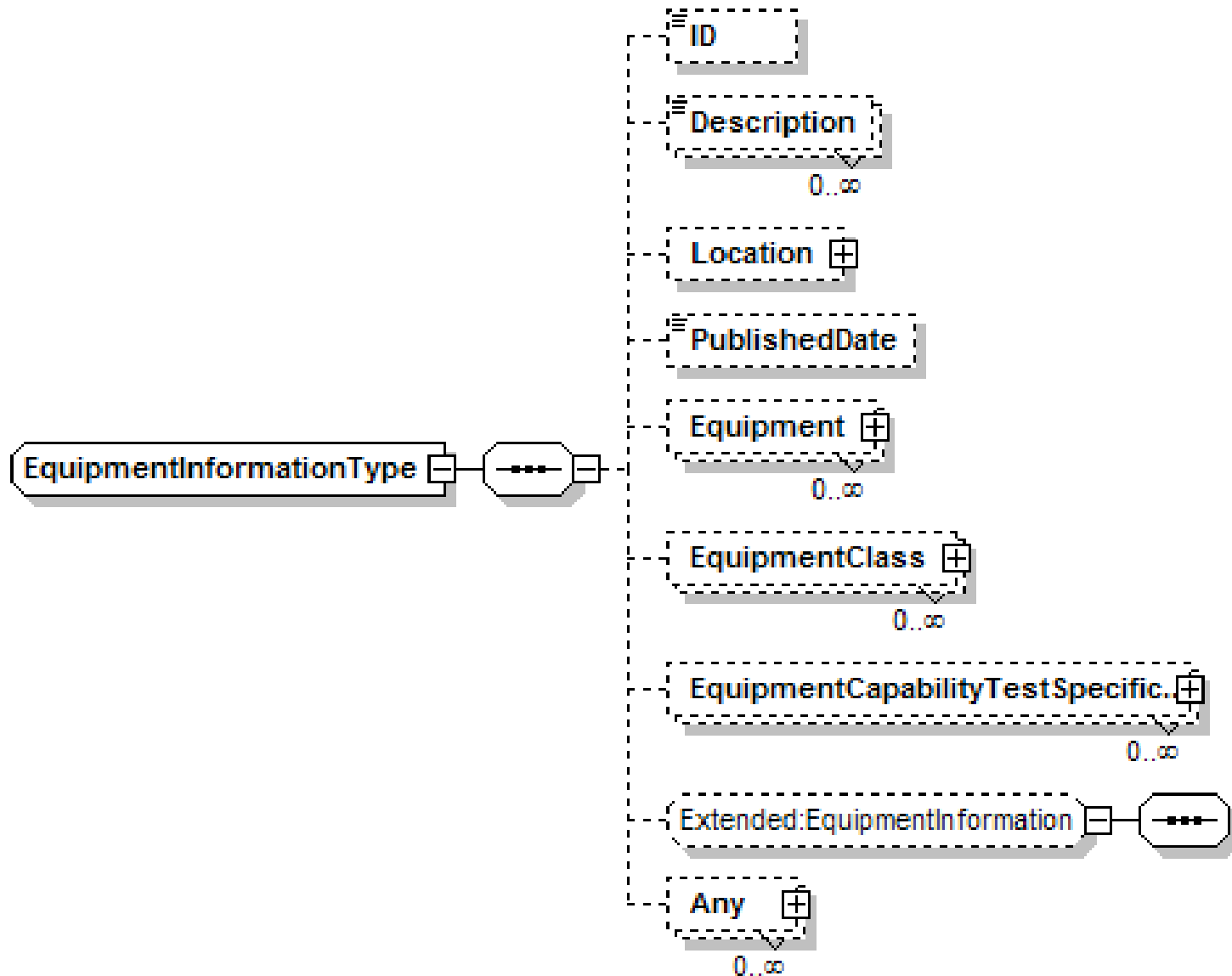
Modèle Equipement => equipment.xsd



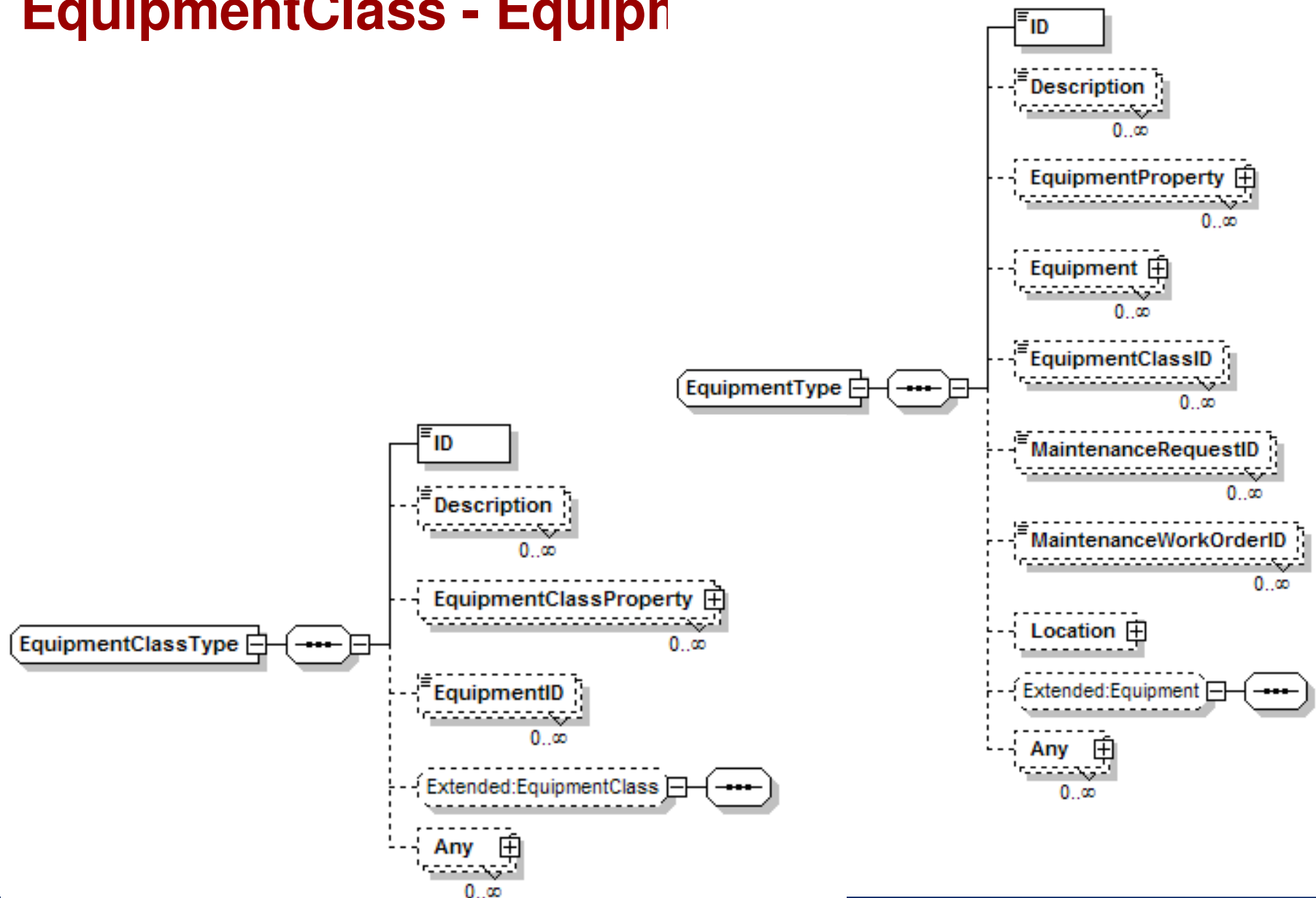
Représentation graphique XMLSpy



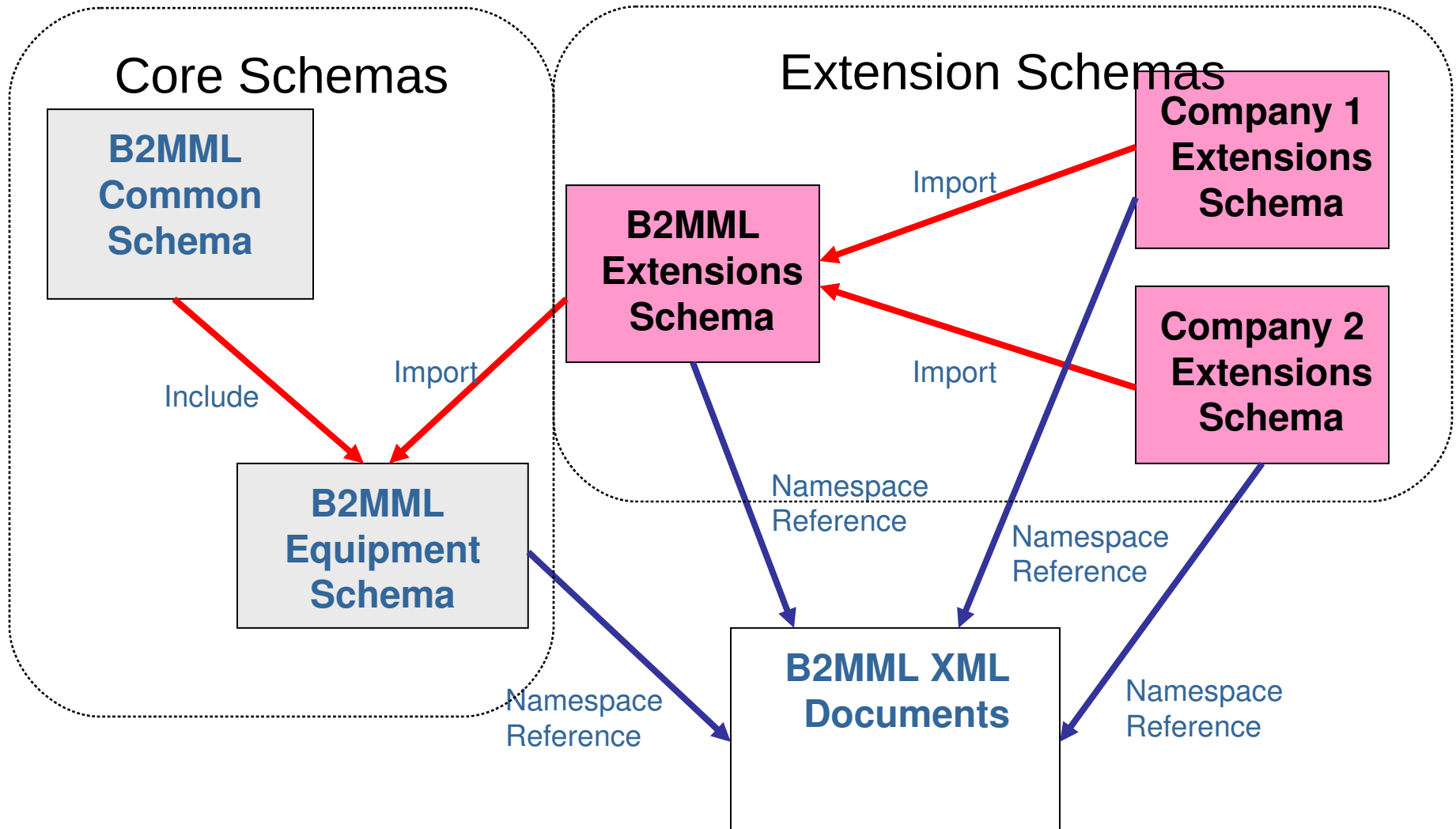
EquipmentInformation



EquipmentClass - Equipment



Exemple – Cartographie des Schema



Namespaces

- **W3C XSD**

- <http://www.w3.org/2001/XMLSchema>
- Les types de base XML Schema / Tous les schémas

- **B2MML**

- <http://www.wbf.org/xml/b2mml-v0300d1>
- Les types de base B2MML / Les schémas B2MML

- **Extension utilisateur**

- <http://www.wbf.org/xml/b2mml-v0300d1-extensions>
- Les extensions et types / le schéma « Extensions »

- **Autres extensions**

- <http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany1>
- <http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany2>
- D'autres types externes à intégrer / Des schémas externes divers

Programme de l'exercice

- 1. Définir les extensions entreprise
- 2. Définir les extensions Company1
- 3. Définir les extensions Company2
- 4. Intégrer les extensions Company1 et 2
- 5. Créer un document XML valide

1. Extensions Entreprise

Etendre le type complexe B2MML Equipment avec :

- Définir un type global “EquipmentAssetType”
 - Composé de **SerialNumber**, **Brand**, **Model**, de type **xsd:string**
- Intégrer les extensions Entreprise
 - “EquipmentAsset”, type **EquipmentAssetType** unique optionnel
 - “Maintenance”, élément complexe local unique optionnel
 - **LastMaintenanceStop** – type **xsd:dateTime**, unique optionnel
 - **NextScheduledMaintenance** – type **xsd:dateTime**, unique optionnel
 - “TRS”, élément complexe local unique optionnel
 - **Jour** – type **xsd:decimal**, unique optionnel
 - **Mois** – type **xsd:decimal**, unique optionnel
 - **Année** – type **xsd:decimal**, unique optionnel
 - “Status”, élément simple – type **xsd:string**, unique optionnel
 - Restriction par une énumération de valeurs possible “**UnderMaintenance**”, “**Busy**”, “**Available**”

1. Extensions Entreprise – Définition Extensions (a)

- Définition du nouveau type complexe EquipmentAssetType

```
<xsd:complexType name="EquipmentAssetType">  
  <xsd:sequence>  
    <xsd:element name="SerialNumber" type="xsd:string" minOccurs="0"/>  
    <xsd:element name="Brand" type="xsd:string" minOccurs="0"/>  
    <xsd:element name="Model" type="xsd:string" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:complexType>
```

1. Extensions Entreprise – Définition Extensions (b)

- Localiser et compléter le groupe de substitution associé à l'élément standard B2MML Equipment
- Définir l'élément EquipmentAsset de type EquipmentAssetType complexe défini précédemment

```
<xsd:group name="Equipment">
  <xsd:sequence>
<!-- add extended elements here -->
    <xsd:element name="EquipmentAsset" type="EquipmentAssetType"
minOccurs="0"/>

    ...
  </xsd:sequence>
</xsd:group>
```

1. Extensions Entreprise – Définition Extensions (b)

- Définir l'élément complexe local Maintenance

```
<xsd:element name="Maintenance" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="LastMaintenanceStop" type="xsd:dateTime«
        minOccurs="0"/>
      <xsd:element name="NextPlannedMaintenance" type="xsd:dateTime«
        minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

1. Extensions Entreprise – Définition Extensions (c)

- Définir l'élément complexe local TRS

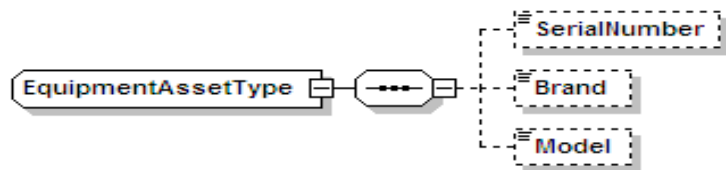
```
<xsd:element name="TRS" minOccurs="0">  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element name="Jour" type="xsd:decimal" minOccurs="0"/>  
      <xsd:element name="Semaine" type="xsd:decimal" minOccurs="0"/>  
      <xsd:element name="Année" type="xsd:decimal" minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>  
</xsd:element>
```

1. Extensions Entreprise – Définition Extensions (d)

- Définir l'élément simple avec énumération Status

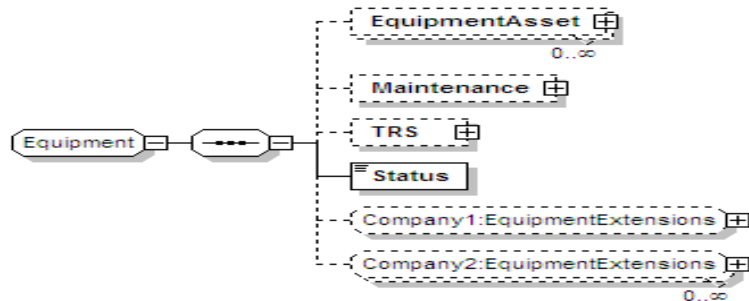
```
<xsd:element name="Status">  
  <xsd:simpleType>  
    <xsd:restriction base="xsd:string">  
      <xsd:enumeration value="UnderMaintenance"/>  
      <xsd:enumeration value="Busy"/>  
      <xsd:enumeration value="Available"/>  
    </xsd:restriction>  
  </xsd:simpleType>  
</xsd:element>
```


1. Extensions Entreprise – Résultat attendu

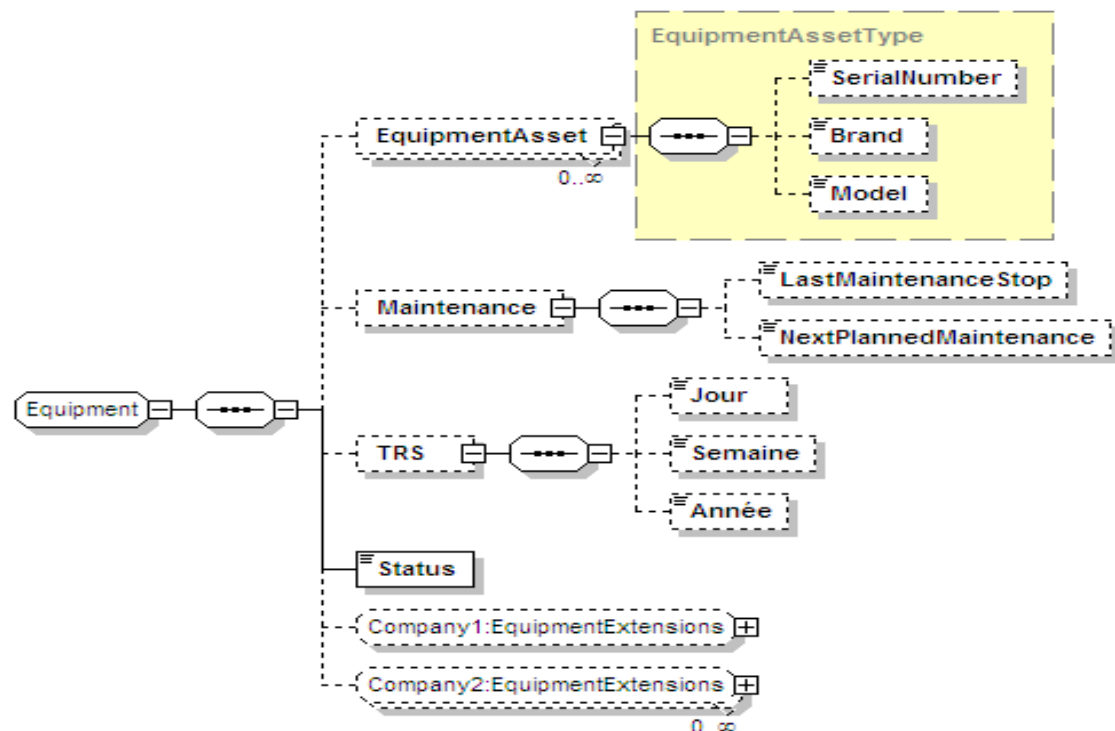


Generated with XMLSpy Schema Editor www.altova.com

+



Generated with XMLSpy Schema Editor www.altova.com



Generated with XMLSpy Schema Editor www.altova.com

Programme de l'exercice

- 1. Définir les extensions entreprise
- 2. Définir les extensions Company1
- 3. Définir les extensions Company2
- 4. Intégrer les extensions Company1 et 2
- 5. Créer un document XML valide

2. Extensions Company1

- Fichier « **B2MML-V0300d1-ExtensionsCompany1.xsd** »
- Objectif: Définir les extensions spécifique au type Equipment du fournisseur Company1
 - Target Namespace « <http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany1> »
- Définit un groupe « **EquipmentExtensions** » composé de
 - **SpeedRotation** – Type `xsd:string` unique optionnel
 - **CouplingHigh** – Type `xsd:string` unique optionnel
 - **CouplingDiameter** – Type `xsd:string` unique optionnel

2. Extensions Company1 – En-tête schéma

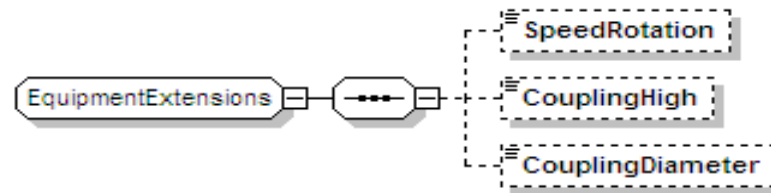
```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2005 sp2 U (http://www.altova.com) by Jean Vieille (Jean
Vieille) -->
<xsd:schema
  xmlns=http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany1
  xmlns:xsd=http://www.w3.org/2001/XMLSchema
  targetNamespace=http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany1
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
<!-- Schema elements
===== -->
</xsd:schema>
```

2. Extensions Company1 – Définitions Extensions

- Définition du groupe global « EquipmentExtensions »

```
<xsd:group name="EquipmentExtensions">  
  <xsd:sequence>  
    <xsd:element name="SpeedRotation" type="xsd:string" minOccurs="0"/>  
    <xsd:element name="CouplingHigh" type="xsd:string" minOccurs="0"/>  
    <xsd:element name="CouplingDiameter" type="xsd:string" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:group>
```

2. Extensions Company1 - Résultat attendu



Generated with XMLSpy Schema Editor www.altova.com

Programme de l'exercice

- 1. Définir les extensions entreprise
- 2. Définir les extensions Company1
- 3. Définir les extensions Company2
- 4. Intégrer les extensions Company1 et 2
- 5. Créer un document XML valide

3. Extensions Company2

- Fichier « **B2MML-V0300d1-ExtensionsCompany2.xsd** »
- Objectif: Définir les extensions spécifique au type Equipment du fournisseur Company2
 - Target Namespace « <http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany2> »
- Définir le type complexe **LubricatorType** composé de
 - ID – Type **xsd:integer** unique obligatoire
 - GreaseQuality – Type **xsd:string** unique optionnel
 - GreaseQuantity – Type **xsd:integer** unique optionnel
 - GreasingPeriod – Type **xsd:duration** unique optionnel
- Définir un groupe « **EquipmentExtensions** » composé de
 - SpeedRotation – Type **xsd:integer** unique optionnel
 - NoiseLevel – Type **xsd:integer** unique optionnel
 - Lubricators – Type **LubricatorType** multiple optionnel

3. Extensions Company2 – En-tête schéma

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- edited with XMLSpy v2005 sp2 U (http://www.altova.com) by Jean Vieille (Jean
Vieille) -->

<xsd:schema
  xmlns=http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany2
  xmlns:xsd=http://www.w3.org/2001/XMLSchema
  targetNamespace=http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany2
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

<!-- Schema elements
===== -->

</xsd:schema>
```

3. Extensions Company2 – Définition Extensions

- Définition type complexe LubricatorType

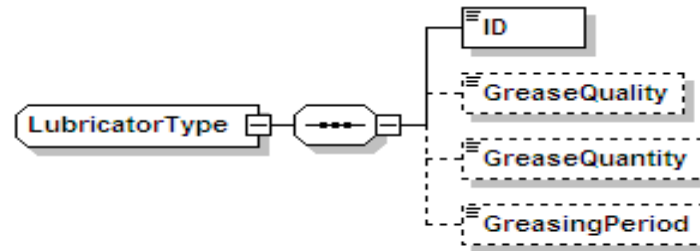
```
<xsd:complexType name="LubricatorType">  
  <xsd:sequence>  
    <xsd:element name="ID" type="xsd:integer"/>  
    <xsd:element name="GreaseQuality" type="xsd:string" minOccurs="0"/>  
    <xsd:element name="GreaseQuantity" type="xsd:integer" minOccurs="0"/>  
    <xsd:element name="GreasingPeriod" type="xsd:duration" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:complexType>
```

3. Extensions Company2 – Définition Extensions

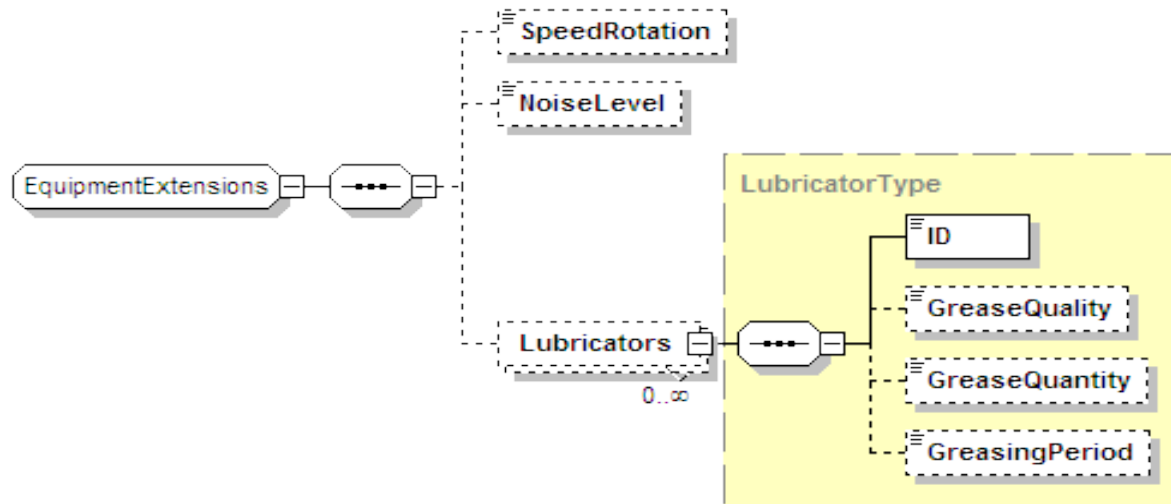
- Définition du groupe global « EquipmentExtensions »

```
<xsd:group name="EquipmentExtensions">  
  <xsd:sequence>  
    <xsd:element name="SpeedRotation" type="xsd:integer" minOccurs="0"/>  
    <xsd:element name="NoiseLevel" type="xsd:integer" minOccurs="0"/>  
    <xsd:element name="Lubricators" type="LubricatorType" minOccurs="0"  
      maxOccurs="unbounded"/>  
  </xsd:sequence>  
</xsd:group>
```

3. Extensions Company2 – Résultat attendu



Generated with XMLSpy Schema Editor www.altova.com



Generated with XMLSpy Schema Editor www.altova.com

Programme de l'exercice

- 1. Définir les extensions entreprise
- 2. Définir les extensions Company1
- 3. Définir les extensions Company2
- 4. Intégrer les extensions Company1 et 2
- 5. Créer un document XML valide

4. Intégration extensions Company1 et 2

Ajouter 2 éléments d'extension de fournisseurs pour le type Equipment

- Ces éléments sont contenus dans des fichiers séparés
- Ces fichiers sont importés dans le fichier B2MML Extensions
- Il y a conflit de noms, des espaces de nom additionnels sont nécessaires
 - Company1 – définit **SpeedRotation** comme **string**
 - Company2 – définit **SpeedRotation** comme **integer**
- Observer l'héritage des types externes dans le schéma B2MML Extensions
- Observer la consolidation des extensions dans le schéma B2MML Equipment (non modifié par lui-même)

4. Intégration extensions Company1 et 2

- Ajout de 2 espaces de noms, un pour chaque fournisseur de schémas d'extensions

```
<?xml version="1.0" encoding="UTF-8"?>  
<xsd:schema targetNamespace ="http://www.wbf.org/xml/b2mml-v0300d1-extensions"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  xmlns="http://www.wbf.org/xml/b2mml-v0300d1-extensions"  
  xmlns:Company1="http://www.wbf.org/xml/b2mml-v0300d1-  
    extensionsCompany1"  
  xmlns:Company2="http://www.wbf.org/xml/b2mml-v0300d1-  
    extensionsCompany2"  
  elementFormDefault="qualified"  
  attributeFormDefault="unqualified">  
...  
...
```

4. Intégration extensions Company1 et 2

- Importation des schémas correspondant aux espaces de nom définis ci-dessus

...

```
<xsd:import namespace="http://www.wbf.org/xml/b2mml-v0300d1-  
    extensionsCompany1"  
    schemaLocation="B2MML-V0300d1-ExtensionsCompany1.xsd"/>  
<xsd:import namespace="http://www.wbf.org/xml/b2mml-v0300d1-  
    extensionsCompany2"  
    schemaLocation="B2MML-V0300d1-ExtensionsCompany2.xsd"/>
```

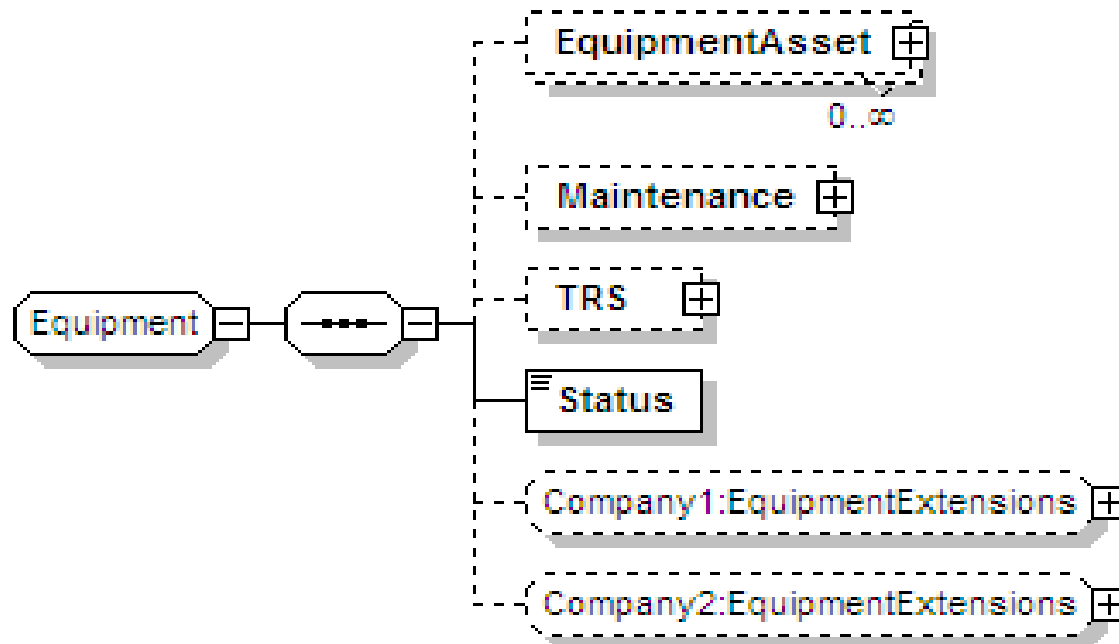
...

4. Intégration extensions Company1 et 2

- Ajouter les références de groupes Company1 et Comany2

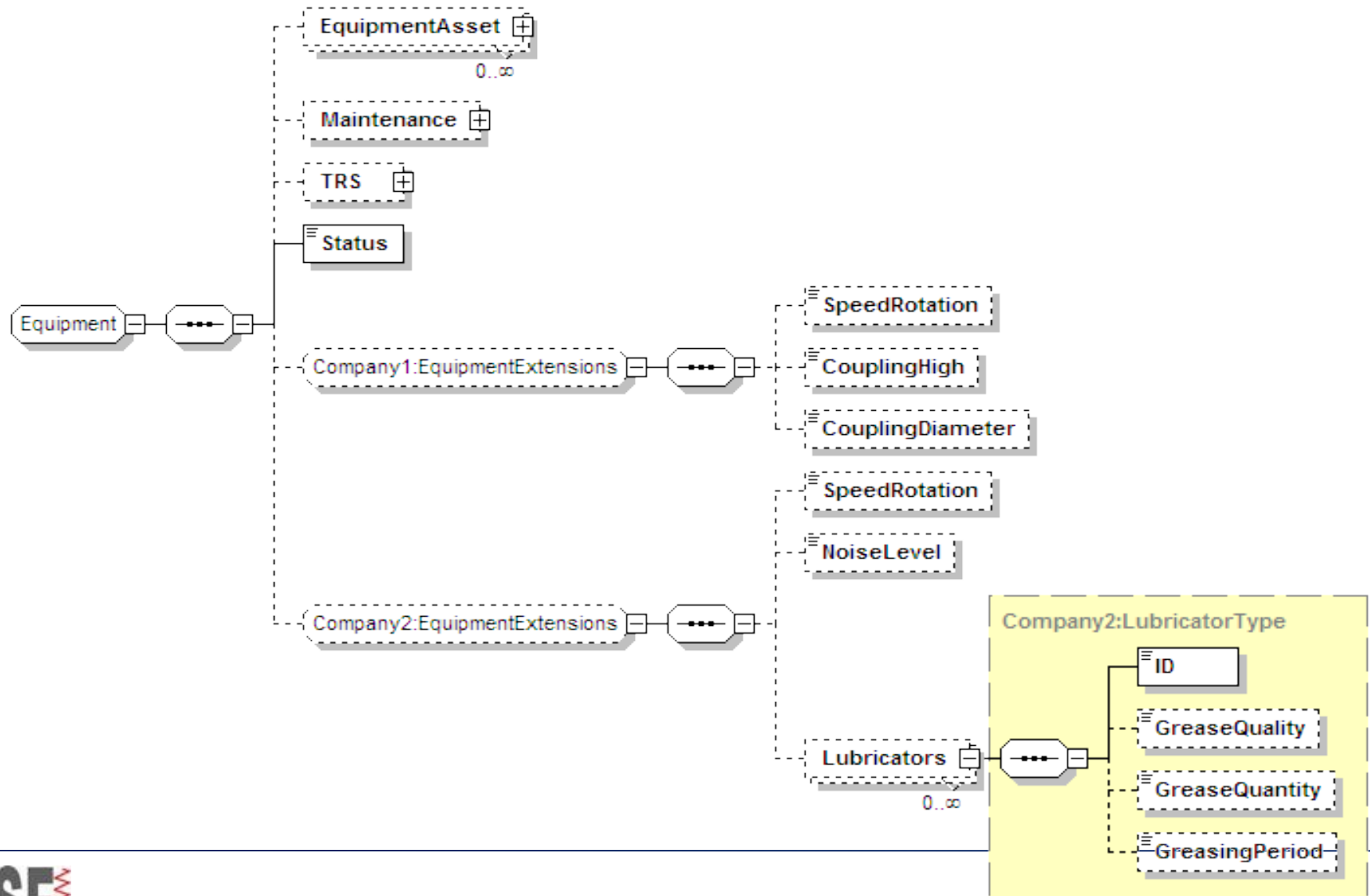
```
<xsd:group name="Equipment">
  <xsd:sequence>
<!-- add extended elements here -->
    <xsd:element name="EquipmentAsset" type="EquipmentAssetType"
      minOccurs="0"maxOccurs="unbounded"/>
    <xsd:element name="Maintenance" minOccurs="0">
    <xsd:element name="TRS" minOccurs="0">
    <xsd:element name="Status">
    <xsd:group ref="Company1:EquipmentExtensions" minOccurs="0"/>
    <xsd:group ref="Company2:EquipmentExtensions" minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Extensions

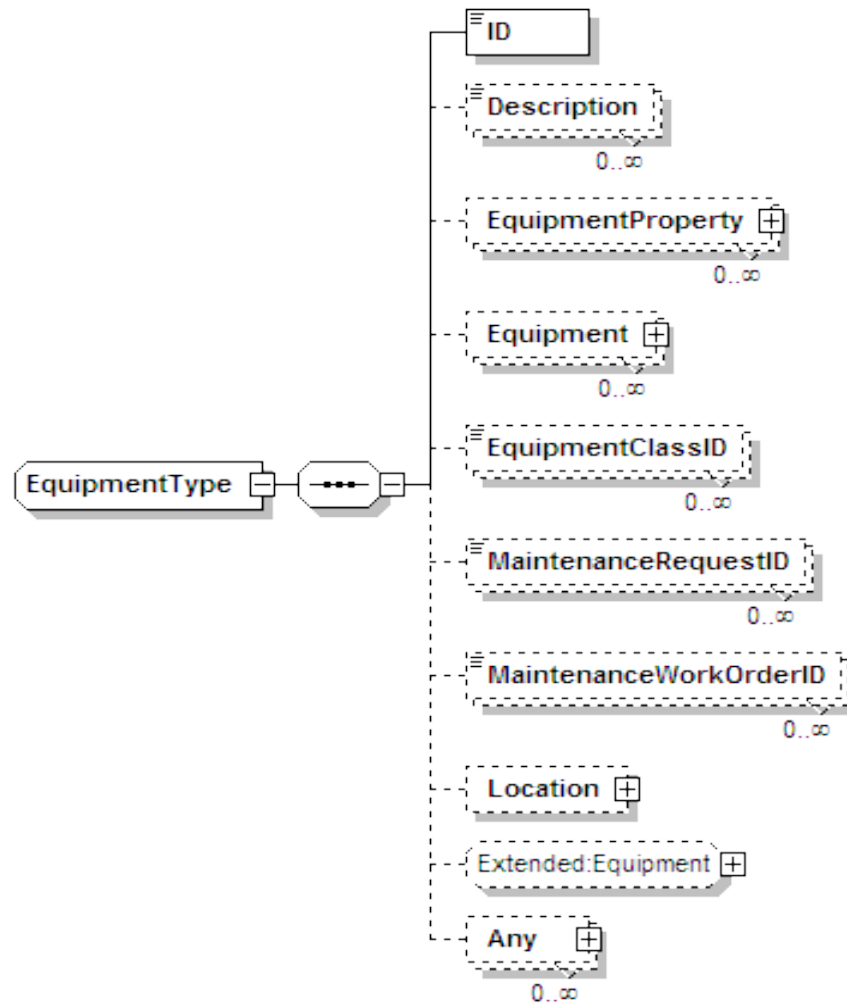


Generated with XMLSpy Schema Editor www.altova.com

4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Extensions

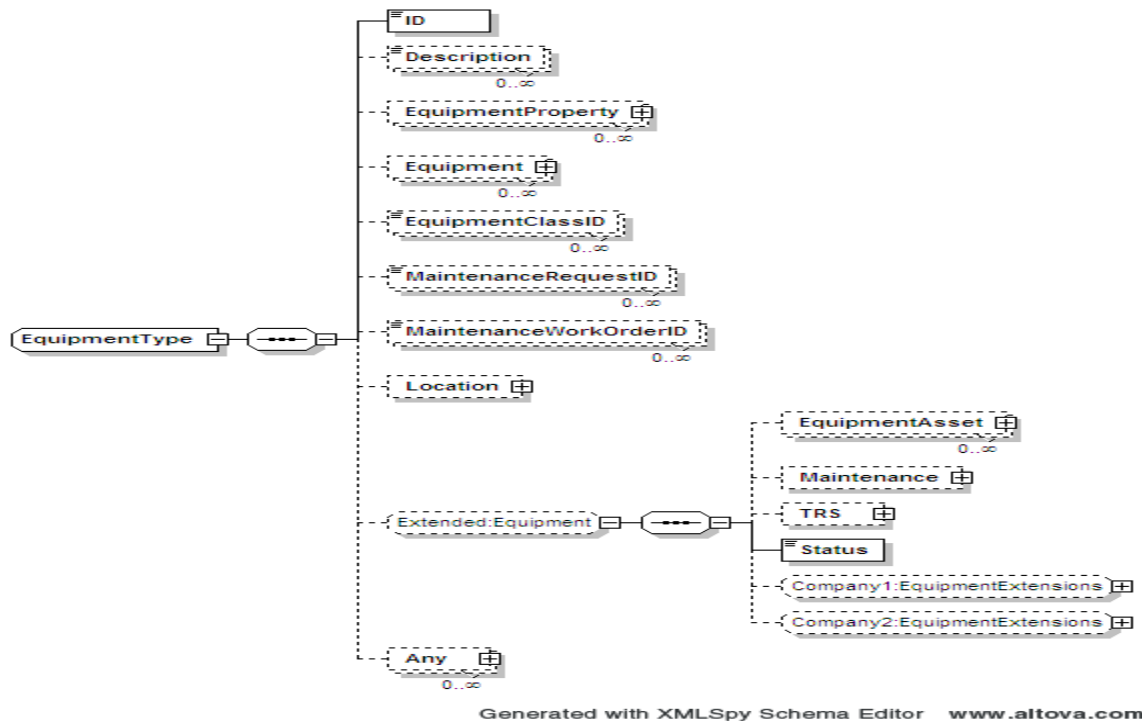


4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Equipement

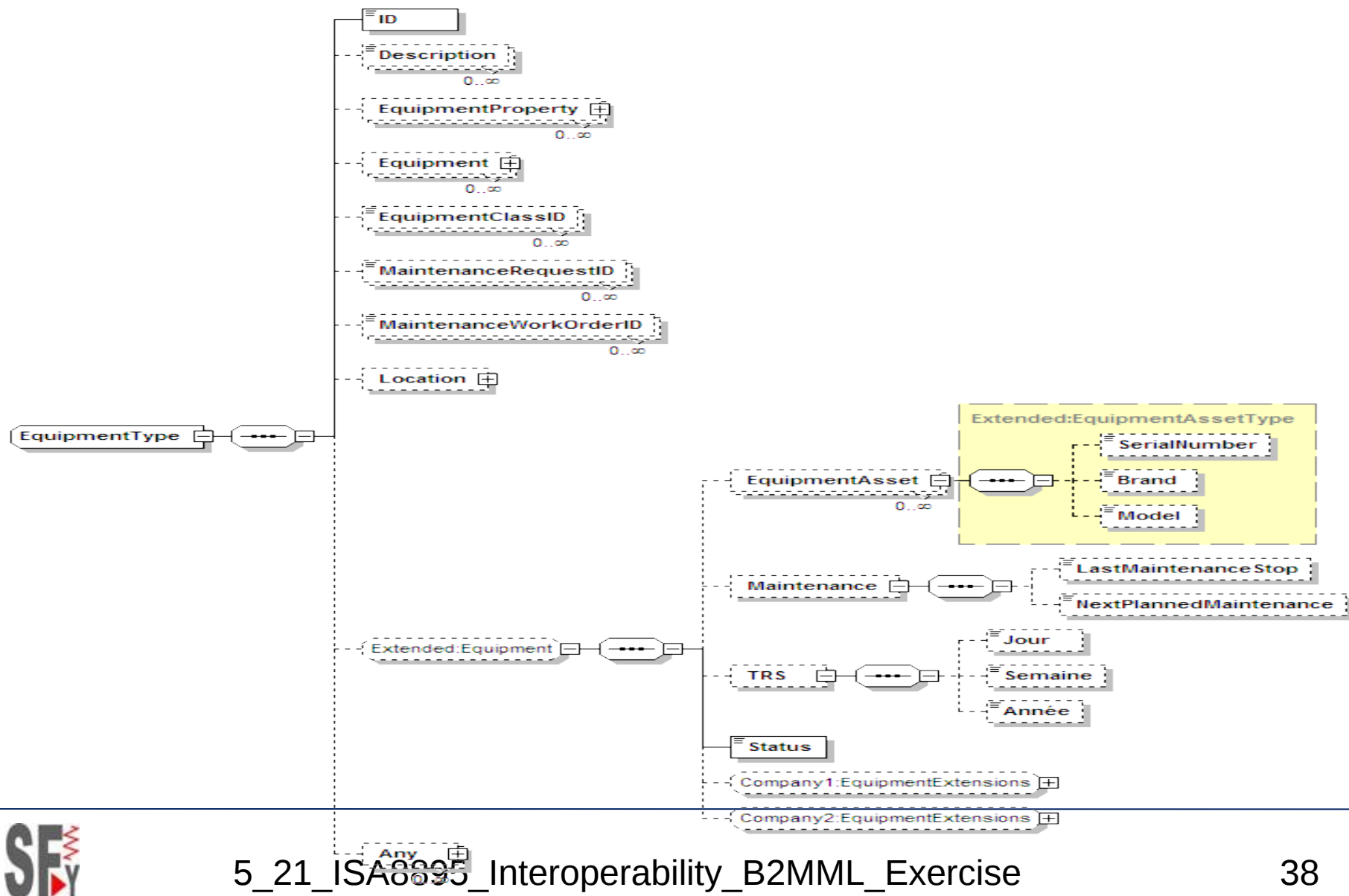


Generated with XMLSpy Schema Editor www.altova.com

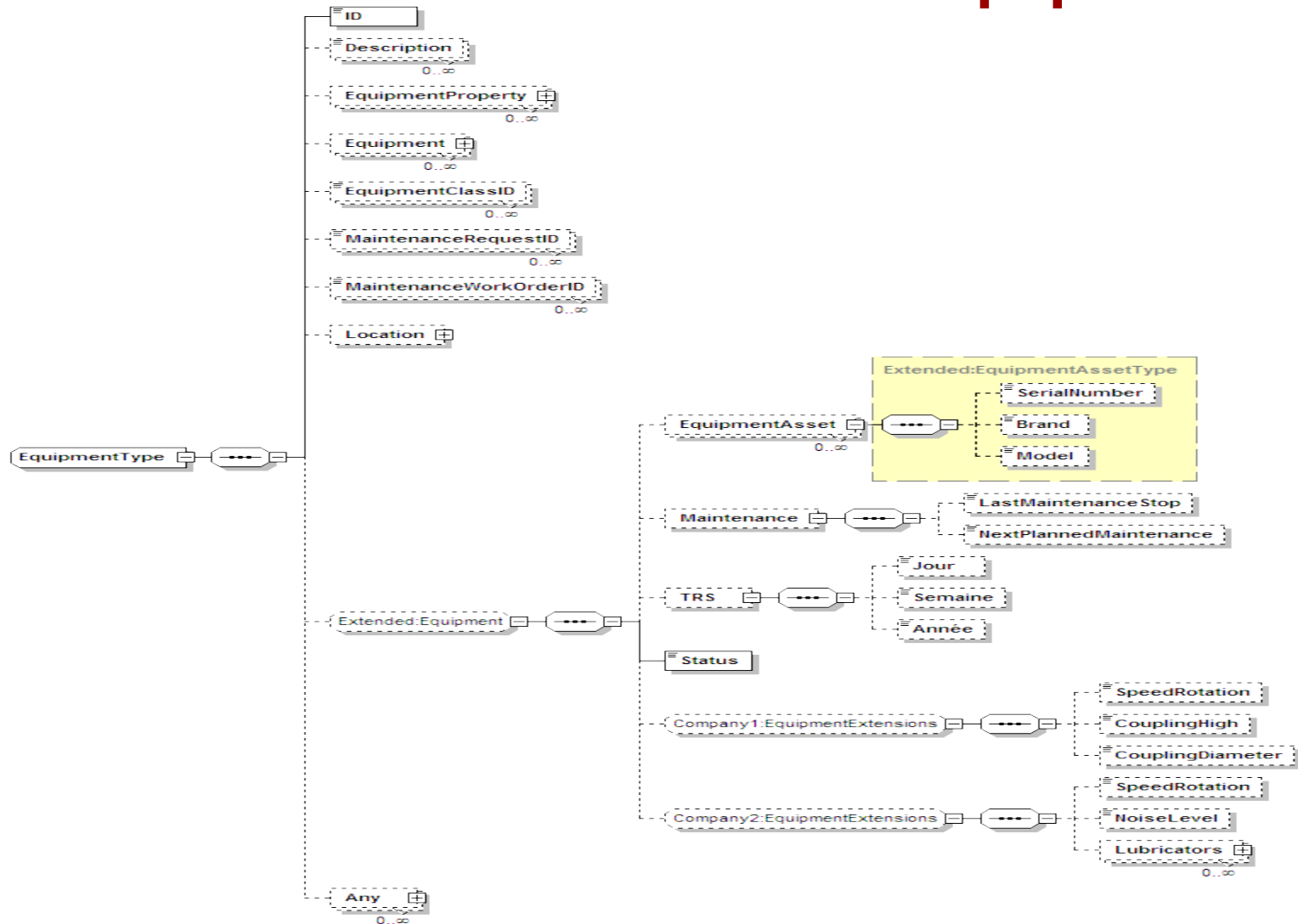
4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Equipement



4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Equipement



4. Intégration extensions Company1 et 2 – Résultat attendu dans le schéma Equipement



Programme de l'exercice

- 1. Définir les extensions entreprise
- 2. Définir les extensions Company1
- 3. Définir les extensions Company2
- 4. Intégrer les extensions Company1 et 2
- 5. Créer un document XML valide

5. Créer un document XML valide (a)

- **Ouvrir le schéma B2MML Equipment**
- **Utilisez XMLSpy pour générer la structure d'un document XML**
 - Menu DTD/Schema>Generate Sample XML File
 - Validez toutes les options, 1 occurrence des éléments répétables
 - Sélectionnez l'élément racine Equipment
 - Le document est généré avec des valeurs par défaut
- **Supprimer l'élément « Element » inclus**
 - Récursivité du modèle non utilisée, pour éviter d'éditer 2 fois les extensions

5. Créer un document XML valide (b)

- **F7 : le document est bien formé**
- **F8 : Le document n'est pas valide.**
 - Les extensions Company1 et Company2 ne sont pas correctement reconnues
 - Importées dans des namespaces séparés dans le schéma extensions
 - Le document généré ne reprend que les namespaces de premier niveau
- **Modifier l'en-tête XML pour inclure les namespaces company1 et Company2**

5. Créer un document XML valide – Entête du document XML

- Les déclarations d'espaces de nom doivent inclure les schémas d'extension

```
<?xml version="1.0" encoding="UTF-8"?>  
<!--Sample XML file generated by XMLSpy v2005 sp2 U (http://www.altova.com)-->  
<Equipment xmlns=http://www.wbf.org/xml/b2mml-v0300d1  
  xmlns:Extended=http://www.wbf.org/xml/b2mml-v0300d1-extensions  
  xmlns:Company1="http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany1"  
  xmlns:Company2="http://www.wbf.org/xml/b2mml-v0300d1-extensionsCompany2"  
  xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance  
  xsi:schemaLocation="http://www.wbf.org/xml/b2mml-v0300d1  
    B2MML-V0300d1-Equipment.xsd">
```

5. Créer un document XML valide – Editer les extensions Company1 et Comapny2

- **Modifier les balises des extensions company1 et 2**
 - Exemple : Company1:SpeedRotation au lieu de SpeedRotation
 - Le document doit être à nouveau valide
- **Conclusion: le document XML valide met en évidence l'origine des extensions**
 - Extended = Entreprise
 - Company1 = Company1
 - Company2 = Company2

Solution – Contenu du document XML

```
<ID>String</ID>
<Description>String</Description>
□ <EquipmentProperty>
<EquipmentClassID>String</EquipmentClassID>
<MaintenanceRequestID>String</MaintenanceRequestID>
<MaintenanceWorkOrderID>String</MaintenanceWorkOrderID>
□ <Location>
□ <Extended:EquipmentAsset>
□ <Extended:Maintenance>
□ <Extended:TRS>
<Extended:Status>UnderMaintenance</Extended:Status>
<Company1:SpeedRotation>String</Company1:SpeedRotation>
<Company1:CouplingHigh>String</Company1:CouplingHigh>
<Company1:CouplingDiameter>String</Company1:CouplingDiameter>
<Company2:SpeedRotation>0</Company2:SpeedRotation>
<Company2:NoiseLevel>0</Company2:NoiseLevel>
□ <Company2:Lubricators>
<Any/>
```

Solution – Contenu du document XML – Détail extension Entreprise

```
<Extended:EquipmentAsset>
  <Extended:SerialNumber>String</Extended:SerialNumber>
  <Extended:Brand>String</Extended:Brand>
  <Extended:Model>String</Extended:Model>
</Extended:EquipmentAsset>
<Extended:Maintenance>
  <Extended:LastMaintenanceStop>
    2001-12-17T09:30:47.0Z</Extended:LastMaintenanceStop>
  <Extended:NextPlannedMaintenance>
    2001-12-17T09:30:47.0Z</Extended:NextPlannedMaintenance>
</Extended:Maintenance>
<Extended:TRS>
  <Extended:Jour>3.1415926535897932384626433832795</Extended:Jour>
  <Extended:Semaine>3.1415926535897932384626433832795</Extended:Semaine>
  <Extended:Année>3.1415926535897932384626433832795</Extended:Année>
</Extended:TRS>
<Extended:Status>UnderMaintenance</Extended:Status>
```

Solution – Contenu du document XML – Détail extension Company1 et 2

```
<Company1:SpeedRotation>String</Company1:SpeedRotation>
<Company1:CouplingHigh>String</Company1:CouplingHigh>
<Company1:CouplingDiameter>String</Company1:CouplingDiameter>
<Company2:SpeedRotation>0</Company2:SpeedRotation>
<Company2:NoiseLevel>0</Company2:NoiseLevel>
<Company2:Lubricators>
  <Company2:ID>0</Company2:ID>
  <Company2:GreaseQuality>String</Company2:GreaseQuality>
  <Company2:GreaseQuantity>0</Company2:GreaseQuantity>
  <Company2:GreasingPeriod>P1Y2M3DT10H30M0S</Company2:GreasingPeriod>
</Company2:Lubricators>
```

FIN

MERCI !