



METAS UncLib - Data Formats V2.9.2

Michael Wollensack

October 2025

Contents

1	XML Schema	3
1.1	ID Type	3
1.2	Distribution Types	4
1.2.1	Standard Normal	4
1.2.2	Normal	4
1.2.3	Standard Uniform	4
1.2.4	Uniform	5
1.2.5	Curvilinear Trapezoid	5
1.2.6	Trapezoidal	5
1.2.7	Triangular	6
1.2.8	Arc Sine	6
1.2.9	Exponential	6
1.2.10	Gamma	7
1.2.11	Chi Squared	7
1.2.12	Student T	7
1.2.13	Student T from Samples	8
1.2.14	Random Choices from Samples	8
1.3	Input Type	8
1.4	LinProp UncNumber Type	9
1.5	MCProp UncNumber Type	9
1.6	LinProp ComplexUncNumber Type	10
1.7	MCProp ComplexUncNumber Type	10
1.8	N-Array Types	10
1.8.1	LinProp Real N-Array Type	11
1.8.2	MCProp Real N-Array Type	11
1.8.3	LinProp Complex N-Array Type	11
1.8.4	MCProp Complex N-Array Type	12
1.9	Elements	12
1.9.1	LinProp Elements	12
1.9.2	MCProp Elements	12
1.9.3	Default Elements	13



METAS UncLib - Data Formats V2.9.2

2	Binary LinProp	14
2.1	UncNumber Version 1	14
2.1.1	Dependencies	14
2.2	UncNumber Version 2	14
2.2.1	Dependencies	14
2.2.2	Input Distribution	15
2.2.3	Distributions	15
2.3	Complex UncNumber Version 1	17
2.4	Flat Vector of UncNumbers Version 1	17
2.4.1	Input	17
2.4.2	Dependencies	18
2.5	Flat Vector of UncNumbers Version 2	18
3	Binary MCPProp	19
3.1	UncNumber Version 1	19
3.2	UncNumber Version 4	19
3.2.1	Dependencies	19
3.3	Complex UncNumber Version 1	19



1 XML Schema

The XML format for METAS UncLib is developed by METAS. It's described using an XML schema. See https://www.w3schools.com/xml/schema_intro.asp for more details about XML schemas.

The following listing shows an example of a LinProp uncertainty number which has two dependencies:

```
1 <?xml version="1.0" encoding="utf-16"?>
2 <UncNumber xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <Value>5</Value>
4   <Dependencies>
5     <DependsOn>
6       <Input>
7         <Id>5C-09-F4-58-A1-4A-40-D3-B5-14-13-94-AE-BC-CA-25</Id>
8         <Description>side a</Description>
9         <Distribution xsi:type="Normal">
10           <mu>3</mu>
11           <sigma>0.3</sigma>
12         </Distribution>
13       </Input>
14       <Jacobi>0.6</Jacobi>
15     </DependsOn>
16     <DependsOn>
17       <Input>
18         <Id>EA-BF-67-4B-28-32-40-84-A4-83-64-30-1D-7C-6A-F5</Id>
19         <Description>side b</Description>
20         <Distribution xsi:type="Normal">
21           <mu>4</mu>
22           <sigma>0.4</sigma>
23         </Distribution>
24       </Input>
25       <Jacobi>0.8</Jacobi>
26     </DependsOn>
27   </Dependencies>
28 </UncNumber>
```

The 'LinPropUncNumberType' is defined in section 1.4. It contains the value and a list of dependencies, where each dependency contains an input and a sensitivity to that input. The 'InputType' is defined in section 1.3. An input contains an identifier, a description and a distribution. The 'IdType' is defined in section 1.1 and the distribution types are defined in section 1.2.

1.1 ID Type

The following listing shows the XML schema for 'IdType':

```
1 <!-- definition of IdType -->
2 <xs:simpleType name="IdType">
3   <!-- unique identifier -->
4   <xs:restriction base="xs:string">
5     <xs:pattern value="([0-9a-fA-F][0-9a-fA-F][\-\-])?"/>
6   </xs:restriction>
7 </xs:simpleType>
```



1.2 Distribution Types

The following listing shows the XML schema for 'Distribution':

```
1 <!-- definition of Distribution -->
2 <xs:complexType name="Distribution" abstract="true"/>
```

1.2.1 Standard Normal

The following listing shows the XML schema for 'StandardNormal':

```
1 <!-- definition of StandardNormal -->
2 <xs:complexType name="StandardNormal">
3   <!-- mu = 0, sigma = 1 -->
4   <!-- mean = mu -->
5   <!-- stdunc = sigma -->
6   <xs:complexContent>
7     <xs:extension base="Distribution">
8       <xs:sequence>
9       </xs:sequence>
10    </xs:extension>
11  </xs:complexContent>
12 </xs:complexType>
```

1.2.2 Normal

The following listing shows the XML schema for 'Normal':

```
1 <!-- definition of Normal -->
2 <xs:complexType name="Normal">
3   <!-- mean = mu -->
4   <!-- stdunc = sigma -->
5   <xs:complexContent>
6     <xs:extension base="Distribution">
7       <xs:sequence>
8         <xs:element name="mu" type="xs:double" />
9         <xs:element name="sigma" type="xs:double" />
10      </xs:sequence>
11    </xs:extension>
12  </xs:complexContent>
13 </xs:complexType>
```

1.2.3 Standard Uniform

The following listing shows the XML schema for 'StandardUniform':

```
1 <!-- definition of StandardUniform -->
2 <xs:complexType name="StandardUniform">
3   <!-- a = 0, b = 1 -->
4   <!-- mean = (a + b) / 2 -->
5   <!-- stdunc = (b - a) / sqrt(12) -->
6   <xs:complexContent>
7     <xs:extension base="Distribution">
8       <xs:sequence>
9       </xs:sequence>
10    </xs:extension>
```



METAS UncLib - Data Formats V2.9.2

```
11     </xs:complexContent>
12 </xs:complexType>
```

1.2.4 Uniform

The following listing shows the XML schema for 'Uniform':

```
1  <!-- definition of Uniform -->
2  <xs:complexType name="Uniform">
3    <!-- mean = (a + b) / 2 -->
4    <!-- stdunc = (b - a) / sqrt(12) -->
5    <xs:complexContent>
6      <xs:extension base="Distribution">
7        <xs:sequence>
8          <xs:element name="a" type="xs:double" />
9          <xs:element name="b" type="xs:double" />
10        </xs:sequence>
11      </xs:extension>
12    </xs:complexContent>
13 </xs:complexType>
```

1.2.5 Curvilinear Trapezoid

The following listing shows the XML schema for 'CurvilinearTrapezoid':

```
1  <!-- definition of CurvilinearTrapezoid -->
2  <xs:complexType name="CurvilinearTrapezoid">
3    <!-- mean = (a + b) / 2 -->
4    <!-- stdunc = sqrt((b - a)^2 / 12 + d^2 / 9) -->
5    <xs:complexContent>
6      <xs:extension base="Distribution">
7        <xs:sequence>
8          <xs:element name="a" type="xs:double" />
9          <xs:element name="b" type="xs:double" />
10         <xs:element name="d" type="xs:double" />
11        </xs:sequence>
12      </xs:extension>
13    </xs:complexContent>
14 </xs:complexType>
```

1.2.6 Trapezoidal

The following listing shows the XML schema for 'Trapezoidal':

```
1  <!-- definition of Trapezoidal -->
2  <xs:complexType name="Trapezoidal">
3    <!-- mean = (a + b) / 2 -->
4    <!-- stdunc = (b - a) * sqrt((1 + beta^2) / 24) -->
5    <xs:complexContent>
6      <xs:extension base="Distribution">
7        <xs:sequence>
8          <xs:element name="a" type="xs:double" />
9          <xs:element name="b" type="xs:double" />
10         <xs:element name="beta" type="xs:double" />
11        </xs:sequence>
```



METAS UncLib - Data Formats V2.9.2

```
12         </xs:extension>
13     </xs:complexContent>
14 </xs:complexType>
```

1.2.7 Triangular

The following listing shows the XML schema for 'Triangular':

```
1  <!-- definition of Triangular -->
2  <xs:complexType name="Triangular">
3      <!-- mean = (a + b) / 2 -->
4      <!-- stdunc = (b - a) / sqrt(24) -->
5      <xs:complexContent>
6          <xs:extension base="Distribution">
7              <xs:sequence>
8                  <xs:element name="a" type="xs:double" />
9                  <xs:element name="b" type="xs:double" />
10             </xs:sequence>
11         </xs:extension>
12     </xs:complexContent>
13 </xs:complexType>
```

1.2.8 Arc Sine

The following listing shows the XML schema for 'ArcSine':

```
1  <!-- definition of ArcSine -->
2  <xs:complexType name="ArcSine">
3      <!-- mean = (a + b) / 2 -->
4      <!-- stdunc = (b - a) / sqrt(8) -->
5      <xs:complexContent>
6          <xs:extension base="Distribution">
7              <xs:sequence>
8                  <xs:element name="a" type="xs:double" />
9                  <xs:element name="b" type="xs:double" />
10             </xs:sequence>
11         </xs:extension>
12     </xs:complexContent>
13 </xs:complexType>
```

1.2.9 Exponential

The following listing shows the XML schema for 'Exponential':

```
1  <!-- definition of Exponential -->
2  <xs:complexType name="Exponential">
3      <!-- mean = mu -->
4      <!-- stdunc = mu -->
5      <xs:complexContent>
6          <xs:extension base="Distribution">
7              <xs:sequence>
8                  <xs:element name="mu" type="xs:double" />
9              </xs:sequence>
10         </xs:extension>
11     </xs:complexContent>
12 </xs:complexType>
```



1.2.10 Gamma

The following listing shows the XML schema for 'Gamma':

```
1      <!-- definition of Gamma -->
2      <xs:complexType name="Gamma">
3          <!-- mean = a * b -->
4          <!-- stdunc = sqrt(a) * b -->
5          <xs:complexContent>
6              <xs:extension base="Distribution">
7                  <xs:sequence>
8                      <xs:element name="a" type="xs:double" />
9                      <xs:element name="b" type="xs:double" />
10                 </xs:sequence>
11             </xs:extension>
12         </xs:complexContent>
13     </xs:complexType>
```

1.2.11 Chi Squared

The following listing shows the XML schema for 'ChiSquared':

```
1      <!-- definition of ChiSquared -->
2      <xs:complexType name="ChiSquared">
3          <!-- mean = k -->
4          <!-- stdunc = sqrt(2 * k) -->
5          <xs:complexContent>
6              <xs:extension base="Distribution">
7                  <xs:sequence>
8                      <xs:element name="k" type="xs:double" />
9                  </xs:sequence>
10             </xs:extension>
11         </xs:complexContent>
12     </xs:complexType>
```

1.2.12 Student T

The following listing shows the XML schema for 'StudentT':

```
1      <!-- definition of StudentT -->
2      <xs:complexType name="StudentT">
3          <!-- mean = mu for n > 2 with n = dof + 1 -->
4          <!-- stdunc = sigma * sqrt((n - 1) / (n - 3)) for n > 3 with n = dof +
5              1 -->
6          <xs:complexContent>
7              <xs:extension base="Distribution">
8                  <xs:sequence>
9                      <xs:element name="mu" type="xs:double" />
10                     <xs:element name="sigma" type="xs:double" />
11                     <xs:element name="dof" type="xs:double" />
12                 </xs:sequence>
13             </xs:extension>
14         </xs:complexContent>
15     </xs:complexType>
```



1.2.13 Student T from Samples

The following listing shows the XML schema for 'StudentTFromSamples':

```
1      <!-- definition of StudentTFromSamples -->
2      <xs:complexType name="StudentTFromSamples">
3          <!-- mean = mean(samples) for n > 2 with n = dof + 1 -->
4          <!-- stdunc = sigma * sqrt((n - 1) / (n - 3)) for n > 3 with n =
              length(samples) and sigma = std(samples) / sqrt(n) -->
5          <xs:complexContent>
6              <xs:extension base="Distribution">
7                  <xs:sequence>
8                      <xs:element name="Samples">
9                          <xs:complexType>
10                             <xs:sequence>
11                                 <xs:element minOccurs="0" maxOccurs="unbounded"
                                    name="Sample" type="xs:double" />
12                             </xs:sequence>
13                         </xs:complexType>
14                     </xs:element>
15                 </xs:sequence>
16             </xs:extension>
17         </xs:complexContent>
18     </xs:complexType>
```

1.2.14 Random Choices from Samples

The following listing shows the XML schema for 'RandomChoicesFromSamples':

```
1      <!-- definition of RandomChoicesFromSamples -->
2      <xs:complexType name="RandomChoicesFromSamples">
3          <xs:complexContent>
4              <xs:extension base="Distribution">
5                  <xs:sequence>
6                      <xs:element name="Seed" type="IdType" />
7                      <xs:element name="Samples">
8                          <xs:complexType>
9                              <xs:sequence>
10                                 <xs:element minOccurs="0" maxOccurs="unbounded"
                                    name="Sample" type="xs:double" />
11                             </xs:sequence>
12                         </xs:complexType>
13                     </xs:element>
14                 </xs:sequence>
15             </xs:extension>
16         </xs:complexContent>
17     </xs:complexType>
```

1.3 Input Type

The following listing shows the XML schema for 'InputType':

```
1      <!-- definition of InputType -->
2      <xs:complexType name="InputType">
3          <xs:sequence>
4              <xs:element name="Id" type="IdType" />
5              <xs:element name="Description" type="xs:string" />
```



METAS UncLib - Data Formats V2.9.2

```
6         <xs:choice>
7             <xs:element name="Distribution" type="Distribution" />
8             <!-- obsolete -->
9             <xs:element name="IDof" type="xs:double" />
10        </xs:choice>
11    </xs:sequence>
12</xs:complexType>
```

The 'IdType' is defined in section 1.1 and the distribution types are defined in section 1.2.

1.4 LinProp UncNumber Type

The following listing shows the XML schema for 'LinPropUncNumberType':

```
1    <!-- definition of LinPropUncNumberType -->
2    <xs:complexType name="LinPropUncNumberType">
3        <xs:sequence>
4            <xs:element name="Value" type="xs:double" />
5            <xs:element name="Dependencies">
6                <xs:complexType>
7                    <xs:sequence>
8                        <xs:element minOccurs="0" maxOccurs="unbounded"
9                            name="DependsOn">
10                            <xs:complexType>
11                                <xs:sequence>
12                                    <xs:element name="Input" type="InputType" />
13                                    <xs:element name="Jacobi" type="xs:double" />
14                                </xs:sequence>
15                            </xs:complexType>
16                        </xs:element>
17                    </xs:sequence>
18                </xs:complexType>
19            </xs:element>
20        </xs:sequence>
21    </xs:complexType>
```

The 'InputType' is defined in section 1.3. An input contains an identifier, a description and a distribution.

1.5 MProp UncNumber Type

The following listing shows the XML schema for 'MPropUncNumberType':

```
1    <!-- definition of MPropUncNumberType -->
2    <xs:complexType name="MPropUncNumberType">
3        <xs:sequence>
4            <xs:element name="FunctionValue" type="xs:double" />
5            <xs:element name="Values">
6                <xs:complexType>
7                    <xs:sequence>
8                        <xs:element minOccurs="0" maxOccurs="unbounded" name="Value"
9                            type="xs:double" />
10                    </xs:sequence>
11                </xs:complexType>
12            </xs:element>
13            <xs:element name="Dependencies">
14                <xs:complexType>
```



```
14         <xs:sequence>
15             <xs:element minOccurs="0" maxOccurs="unbounded"
16                 name="DependsOn">
17                 <xs:complexType>
18                     <xs:sequence>
19                         <xs:element name="Input" type="InputType" />
20                     </xs:sequence>
21                 </xs:complexType>
22             </xs:element>
23         </xs:sequence>
24     </xs:complexType>
25 </xs:element>
26 </xs:sequence>
</xs:complexType>
```

1.6 LinProp ComplexUncNumber Type

The following listing shows the XML schema for 'LinPropComplexUncNumberType':

```
1  <!-- definition of LinPropComplexUncNumberType -->
2  <xs:complexType name="LinPropComplexUncNumberType">
3      <xs:sequence>
4          <xs:element name="Real" type="LinPropUncNumberType" />
5          <xs:element name="Imag" type="LinPropUncNumberType" />
6      </xs:sequence>
7  </xs:complexType>
```

1.7 MCProp ComplexUncNumber Type

The following listing shows the XML schema for 'MCPropComplexUncNumberType':

```
1  <!-- definition of MCPropComplexUncNumberType -->
2  <xs:complexType name="MCPropComplexUncNumberType">
3      <xs:sequence>
4          <xs:element name="Real" type="MCPropUncNumberType" />
5          <xs:element name="Imag" type="MCPropUncNumberType" />
6      </xs:sequence>
7  </xs:complexType>
```

1.8 N-Array Types

The following listing shows the XML schema for 'NArrayType':

```
1  <!-- definition of NArrayType -->
2  <xs:complexType name="NArrayType" abstract="true">
3      <xs:sequence>
4          <xs:element name="NDims" type="xsd:int" />
5          <xs:element name="Size">
6              <xs:complexType>
7                  <xs:sequence>
8                      <xs:element minOccurs="0" maxOccurs="unbounded" name="int"
9                          type="xs:int" />
10                 </xs:sequence>
11             </xs:complexType>
12         </xs:element>
```



METAS UncLib - Data Formats V2.9.2

```
12     </xs:sequence>
13 </xs:complexType>
```

1.8.1 LinProp Real N-Array Type

The following listing shows the XML schema for 'LinPropRealNArrayType':

```
1  <!-- definition of LinPropRealNArrayType -->
2  <xs:complexType name="LinPropRealNArrayType">
3    <xs:complexContent>
4      <xs:extension base="NArrayType">
5        <xs:sequence>
6          <xs:element name="Data">
7            <xs:complexType>
8              <xs:sequence>
9                <xs:element minOccurs="0" maxOccurs="unbounded"
10                  name="UncNumber" type="LinPropUncNumberType" />
11              </xs:sequence>
12            </xs:complexType>
13          </xs:element>
14        </xs:sequence>
15      </xs:extension>
16    </xs:complexContent>
17  </xs:complexType>
```

1.8.2 MCProp Real N-Array Type

The following listing shows the XML schema for 'MCPropRealNArrayType':

```
1  <!-- definition of MCPropRealNArrayType -->
2  <xs:complexType name="MCPropRealNArrayType">
3    <xs:complexContent>
4      <xs:extension base="NArrayType">
5        <xs:sequence>
6          <xs:element name="Data">
7            <xs:complexType>
8              <xs:sequence>
9                <xs:element minOccurs="0" maxOccurs="unbounded"
10                  name="UncNumber" type="MCPropUncNumberType" />
11              </xs:sequence>
12            </xs:complexType>
13          </xs:element>
14        </xs:sequence>
15      </xs:extension>
16    </xs:complexContent>
17  </xs:complexType>
```

1.8.3 LinProp Complex N-Array Type

The following listing shows the XML schema for 'LinPropComplexNArrayType':

```
1  <!-- definition of LinPropComplexNArrayType -->
2  <xs:complexType name="LinPropComplexNArrayType">
3    <xs:complexContent>
4      <xs:extension base="NArrayType">
```



METAS UncLib - Data Formats V2.9.2

```
5      <xs:sequence>
6      <xs:element name="Data">
7      <xs:complexType>
8      <xs:sequence>
9      <xs:element minOccurs="0" maxOccurs="unbounded"
      name="ComplexOfUncNumber"
      type="LinPropComplexUncNumberType" />
10     </xs:sequence>
11   </xs:complexType>
12 </xs:element>
13 </xs:sequence>
14 </xs:extension>
15 </xs:complexContent>
16 </xs:complexType>
```

1.8.4 MCProp Complex N-Array Type

The following listing shows the XML schema for 'MCPropComplexNArrayType':

```
1  <!-- definition of MCPropComplexNArrayType -->
2  <xs:complexType name="MCPropComplexNArrayType">
3    <xs:complexContent>
4      <xs:extension base="NArrayType">
5        <xs:sequence>
6          <xs:element name="Data">
7            <xs:complexType>
8              <xs:sequence>
9                <xs:element minOccurs="0" maxOccurs="unbounded"
                name="ComplexOfUncNumber"
                type="MCPropComplexUncNumberType" />
10             </xs:sequence>
11           </xs:complexType>
12         </xs:element>
13       </xs:sequence>
14     </xs:extension>
15   </xs:complexContent>
16 </xs:complexType>
```

1.9 Elements

1.9.1 LinProp Elements

The following listing shows the XML schema for LinProp elements:

```
1  <!-- definition of LinProp elements -->
2  <xs:element name="LinPropUncNumber" type="LinPropUncNumberType" />
3  <xs:element name="LinPropComplexOfUncNumber"
      type="LinPropComplexUncNumberType" />
4  <xs:element name="LinPropRealNArray" type="LinPropRealNArrayType" />
5  <xs:element name="LinPropComplexNArray" type="LinPropComplexNArrayType"
      />
```

1.9.2 MCProp Elements

The following listing shows the XML schema for MCProp elements:



METAS UncLib - Data Formats V2.9.2

```
1  <!-- definition of MProp elements -->
2  <xs:element name="MPropUncNumber" type="MPropUncNumberType" />
3  <xs:element name="MPropComplexOfUncNumber"
4    type="MPropComplexUncNumberType" />
5  <xs:element name="MPropRealNArray" type="MPropRealNArrayType" />
6  <xs:element name="MPropComplexNArray" type="MPropComplexNArrayType" />
```

1.9.3 Default Elements

The following listing shows the XML schema for default elements:

```
1  <!-- definition of default elements -->
2  <xs:element name="UncNumber" type="LinPropUncNumberType" />
3  <xs:element name="ComplexOfUncNumber" type="LinPropComplexUncNumberType"
4    />
5  <xs:element name="RealNArray" type="LinPropRealNArrayType" />
6  <xs:element name="ComplexNArray" type="LinPropComplexNArrayType" />
```



2 Binary LinProp

The version 1 data formats assume a standard normal distribution for the input of each dependencies. The version 2 data formats allow to specify the type of distribution and its parameters for the input of each dependencies.

2.1 UncNumber Version 1

The following enumeration describes the binary structure version 1 of 'UncNumber':

1. Version (int32), value: 1
2. Value (double)
3. Version2 (int32), value: 4
4. Number of Dependencies (int32)
5. Dependencies (DependsOn[]), size: number of dependencies

2.1.1 Dependencies

The following enumeration describes the binary structure of 'DependsOn':

1. Number of Id Bytes (int32)
2. Input Id (byte[]), size: number of id bytes
3. Input Description (string)
4. Input IDof (double)
5. Jacobi (double)

2.2 UncNumber Version 2

The following enumeration describes the binary structure version 2 of 'UncNumber':

1. Version (7-bit encoded int), value: 2
2. Value (double)
3. Number of Dependencies (7-bit encoded int)
4. Dependencies (DependsOn2[]), size: number of dependencies

2.2.1 Dependencies

The following enumeration describes the binary structure of 'DependsOn2':

1. Input (InputDistribution)
2. Jacobi (double)



METAS UncLib - Data Formats V2.9.2

2.2.2 Input Distribution

The following enumeration describes the binary structure of 'InputDistribution':

1. Version (7-bit encoded int), value: 2
2. Number of Id Bytes (7-bit encoded int)
3. Id (byte[]), size: number of id bytes
4. Description (string)
5. Distribution (Distribution)

2.2.3 Distributions

The following enumeration describes the binary structure of 'StandardNormalDistribution':

1. Type (7-bit encoded int), value: 0

The following enumeration describes the binary structure of 'NormalDistribution':

1. Type (7-bit encoded int), value: 1
2. mu (double)
3. sigma (double)

The following enumeration describes the binary structure of 'StandardUniformDistribution':

1. Type (7-bit encoded int), value: 2

The following enumeration describes the binary structure of 'UniformDistribution':

1. Type (7-bit encoded int), value: 3
2. a (double)
3. b (double)

The following enumeration describes the binary structure of 'CurvilinearTrapezoidDistribution':

1. Type (7-bit encoded int), value: 4
2. a (double)
3. b (double)
4. d (double)

The following enumeration describes the binary structure of 'TrapezoidalDistribution':

1. Type (7-bit encoded int), value: 5
2. a (double)



METAS UncLib - Data Formats V2.9.2

3. b (double)
4. beta (double)

The following enumeration describes the binary structure of 'TriangularDistribution':

1. Type (7-bit encoded int), value: 6
2. a (double)
3. b (double)

The following enumeration describes the binary structure of 'ArcSineDistribution':

1. Type (7-bit encoded int), value: 7
2. a (double)
3. b (double)

The following enumeration describes the binary structure of 'GammaDistribution':

1. Type (7-bit encoded int), value: 8
2. a (double)
3. b (double)

The following enumeration describes the binary structure of 'ChiSquaredDistribution':

1. Type (7-bit encoded int), value: 9
2. k (int32)

The following enumeration describes the binary structure of 'StudentTDistribution':

1. Type (7-bit encoded int), value: 10
2. mu (double)
3. sigma (double)
4. dof (double)

The following enumeration describes the binary structure of 'StudentTFromSamplesDistribution':

1. Type (7-bit encoded int), value: 11
2. Version (7-bit encoded int), value: 2
3. Number of Samples (7-bit encoded int)
4. Samples (double[]), size: number of samples

The following enumeration describes the binary structure of 'RandomChoicesFromSamples':



METAS UncLib - Data Formats V2.9.2

1. Type (7-bit encoded int), value: 99
2. Version (7-bit encoded int), value: 2
3. Number of Seed Bytes (7-bit encoded int)
4. Seed (byte[]), size: number of seed bytes
5. Number of Samples (7-bit encoded int)
6. Samples (double[]), size: number of samples

2.3 Complex UncNumber Version 1

The following enumeration describes the binary structure of 'ComplexUncNumber':

1. Version (int32), value: 1
2. Real (UncNumber)
3. Imag (UncNumber)

'UncNumber' could be either of version 1 or 2.

2.4 Flat Vector of UncNumbers Version 1

The following enumeration describes the binary structure version 1 of 'FlatVectorUncNumbers':

1. Version (int32), value: 1
2. Length (7-bit encoded int)
3. Values (double[]), size: length
4. Number of Inputs (7-bit encoded int)
5. Inputs (UncInput[]), size: number of inputs
6. Dependencies (UncDependencies[]), size: length

2.4.1 Input

The following enumeration describes the binary structure of 'UncInput':

1. Temp (byte), bit 0: same id size, bit 1: empty description, bit 2: zero idof, bit 3-7: 0
2. Id Size (7-bit encoded int), field only present if not same id size
3. Input Id (byte []), size: id size
4. Input Description (string), field only present if not empty description
5. Input IDof (double), field only present if not zero idof



METAS UncLib - Data Formats V2.9.2

2.4.2 Dependencies

The following enumeration describes the binary structure of 'UncDependencies'

1. Number of Dependencies (7-bit encoded int)
2. Dependencies (UncDependency[]), size: number of dependencies, pointer to inputs set to 0.

The following enumeration describes the binary structure of 'UncDependency'

1. Relative Pointer to Inputs (7-bit encoded int)
2. Jacobi (double)

2.5 Flat Vector of UncNumbers Version 2

The following enumeration describes the binary structure version 2 of 'FlatVectorUncNumbers':

1. Version (7-bit encoded int), value: 2
2. Length (7-bit encoded int)
3. Values (double[]), size: length
4. Number of Inputs (7-bit encoded int)
5. Inputs (InputDistribution[]), size: number of inputs
6. Dependencies (UncDependencies[]), size: length

For the definition of 'InputDistribution' see section 2.2.2. For the definition of 'UncDependencies' see section 2.4.2.



3 Binary MCProp

3.1 UncNumber Version 1

The following enumeration describes the binary structure version 1 of 'UncNumber':

1. Version (int32), value: 1
2. Function Value (double)
3. Number of Values (int32)
4. Values (double[]), size: number of values

3.2 UncNumber Version 4

Version 2 and 3 had an bug and they are not used. The following enumeration describes the binary structure version 4 of 'UncNumber':

1. Version (int32), value: 4
2. Function Value (double)
3. Number of Values (int32)
4. Values (double[]), size: number of values
5. Number of Dependencies (int32)
6. Dependencies (MCPropDependencies[]), size: number of dependencies

3.2.1 Dependencies

The following enumeration describes the binary structure of 'MCPropDependencies':

1. Input (InputDistribution)

For the definition of 'InputDistribution' see section 2.2.2.

3.3 Complex UncNumber Version 1

The following enumeration describes the binary structure of 'ComplexUncNumber':

1. Version (int32), value: 1
2. Real (UncNumber)
3. Imag (UncNumber)

'UncNumber' could be either of version 1 or 4.