Mapping IDL Data Types to an XML Schema (XSD)

In the table below, the following metasymbols and informal terms are used for the IDL.

- The metasymbols "[" and "]" surround optional lexical entities.

- The informal term *number* (or in some cases *number1.number2*) is a sequence of numeric characters, for example 123.

IDL Data Type	Description	XMM	XSD

A*number*	Alphanumeric	string	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:string">`
			`<xsd:maxLength value="number"/>`
			`</xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

AV	Alphanumeric variable length	string	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:string"></xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

AV[*number*]	Alphanumeric variable length with maximum length	string	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:string">`
			`<xsd:maxLength value="number"/>`
			`</xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

B*number*	Binary	`binary`	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:base64Binary">`
			`<xsd:length value="base64Length"/>`
			`</xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`
			
			Note:
			$base64Length = 4 *$ rounded up($number/3$)

BV	Binary variable length	binary	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:base64Binary"></xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

BV[*number*]	Binary variable length with maximum length	binary	`<xsd:element name="name" form="unqualified"> <xsd:simpleType>`
			`<xsd:restriction base="xsd:base64Binary">`
			`<xsd:maxLength value="base64Length"/>`
			`</xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`
			
			Note:
			$base64Length = 4 *$ rounded up($number/3$)

D	Date	date:yyyy-MM-dd	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:date"></xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

F4	Floating point (small)	float	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:float"></xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

F8	Floating point (large)	float	`<xsd:element name="name" form="unqualified">`
			`<xsd:simpleType>`
			`<xsd:restriction base="xsd:double"></xsd:restriction>`
			`</xsd:simpleType>`
			`</xsd:element>`

IDL Data Type	Description	XMM	XSD
I1	Integer (small)	integer	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:byte">` ` <xsd:totalDigits value="3"/>` ` <xsd:fractionDigits value="0"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
I2	Integer (medium)	integer	```<xsd:element name=" name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:short">` ` <xsd:totalDigits value="5"/>` ` <xsd:fractionDigits value="0"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
I4	Integer (large)	integer	```<xsd:element name=" name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:int">` ` <xsd:totalDigits value="10"/>` ` <xsd:fractionDigits value="0"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
K*number*	Kanji	string	```<xsd:element name=" name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:string">` ` <xsd:maxLength value="number"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
KV	Kanji variable length	string	```<xsd:element name=" name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:string"></xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
KV[*number*]	Kanji variable length with maximum length	string	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:string">` ` <xsd:maxLength value="number"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
L	Logical	boolean	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:boolean"></xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>```
N*number1*[.*number2*]	Unpacked decimal	numeric	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:decimal">` ` <xsd:totalDigits value="number1 + number2"/>` ` <xsd:fractionDigits value="number2"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>``` **Note:** default of *number2* is 0.
NU*number1*[.*number2*]	Unpacked decimal unsigned	numeric	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:decimal">` ` <xsd:totalDigits value="number1 + number2"/>` ` <xsd:fractionDigits value="number2"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>``` **Note:** default of *number2* is 0.
P*number1*[.*number2*]	Packed decimal	numeric	```<xsd:element name="name" form="unqualified">` ` <xsd:simpleType>` ` <xsd:restriction base="xsd:decimal">` ` <xsd:totalDigits value="number1 + number2"/>` ` <xsd:fractionDigits value="number2"/>` ` </xsd:restriction>` ` </xsd:simpleType>` `</xsd:element>``` **Note:** default of *number2* is 0.

IDL Data Type	Description	XMM	XSD

PU*number1*[*.number2*]	Packed decimal unsigned	numeric	```<xsd:element name="name" form="unqualified">```
			```<xsd:simpleType>```
			```<xsd:restriction base="xsd:decimal">```
			```<xsd:totalDigits value="number1 + number2"/>```
			```<xsd:fractionDigits value="number2"/>```
			```</xsd:restriction>```
			```</xsd:simpleType>```
			```</xsd:element>```
			
			Note:
			default of *number2* is 0.

T	Time	dateTime:yyyy-MM-dd'T'H:mm:ss	```<xsd:element name="name" form="unqualified">```
			```<xsd:simpleType>```
			```<xsd:restriction base="xsd:dateTime"></xsd:restriction>```
			```</xsd:simpleType>```
			```</xsd:element>```

U*number*	Unicode	unicode	```<xsd:element name="name" form="unqualified">```
			```<xsd:simpleType>```
			```<xsd:restriction base="xsd:string">```
			```<xsd:maxLength value="number"/>```
			```</xsd:restriction>```
			```</xsd:simpleType>```
			```</xsd:element>```

UV	Unicode variable length	unicode	```<xsd:element name="name" form="unqualified">```
			```<xsd:simpleType>```
			```<xsd:restriction base="xsd:string"></xsd:restriction>```
			```</xsd:simpleType>```
			```</xsd:element>```

UV*number*	Unicode variable length with maximum length	unicode	```<xsd:element name="name" form="unqualified">```
			```<xsd:simpleType>```
			```<xsd:restriction base="xsd:string">```
			```<xsd:maxLength value="number"/>```
			```</xsd:restriction>```
			```</xsd:simpleType>```
			```</xsd:element>```