

eCH-0117 Meta-Attribute für INTERLIS-Modelle

Name	Meta-Attribute für INTERLIS-Modelle
Standard-Nummer	eCH-0117
Kategorie	Standard
Reifegrad	Experimentell
Version	1.00
Status	Genehmigt
Genehmigt am	2011-06-08
Ausgabedatum	2011-06-22
Ersetzt Standard	
Sprachen	Deutsch, Französisch
Autoren	eCH-Fachgruppe INTERLIS <i>Claude Eisenhut, Eisenhut Informatik AG</i> <i>Michael Germann, infoGrips GmbH</i> <i>Dr. Peter Staub, swisstopo</i>
Herausgeber / Vertrieb	Verein eCH, Mainaustrasse 30, Postfach, 8034 Zürich T 044 388 74 64, F 044 388 71 80 www.ech.ch / info@ech.ch

Zusammenfassung

Dieser Standard beschreibt die Definition von Meta-Attributen als Teil eines INTERLIS-Datenmodells. Mit den Regeln, die in der vorliegenden Spezifikation definiert werden, können Meta-Informationen in ein INTERLIS-Datenmodell integriert werden, ohne dass dafür die INTERLIS-Sprachspezifikation [1] angepasst werden muss.

Die Syntaxregeln für Meta-Attribute werden in das Regelwerk von INTERLIS eingepasst. Insbesondere wird definiert, an welchen Stellen in der bestehenden INTERLIS-Spezifikation neu Meta-Attribute definiert werden können.

Inhaltsverzeichnis

1	Status des Dokuments	4
2	Einleitung	5
	2.1 Überblick	5
	2.2 Vorteile	5
3	Konzept	5
4	Syntax	6
	4.1 Syntax-Definitionen	6
	4.2 Syntaxregeln für Meta-Attribute	6
5	Positionierung innerhalb der INTERLIS-Sprache	8
6	Beispiel	11
7	Haftungsausschluss/Hinweise auf Rechte Dritter	12
8	Urheberrechte	12
	Anhang – Referenzen & Bibliographie	13

1 Status des Dokuments

Das vorliegende Dokument wurde vom Expertenausschuss **genehmigt**. Es hat für das definierte Einsatzgebiet im festgelegten Gültigkeitsbereich normative Kraft.

2 Einleitung

2.1 Überblick

Meta-Attribute erlauben eine Ergänzung der INTERLIS-Modelle mit zusätzlichen Angaben, die in der aktuellen Spezifikation von INTERLIS [1] nicht vorgesehen sind.

Im Dokument zum INTERLIS 2-Metamodell [2] steht dazu:

«Einem Metaelement können kompositorisch Metaattribute zugeordnet sein (Klasse "MetaAttribut"). Metaattribute sind weder durch die Sprache INTERLIS noch durch das Metamodell genauer definiert. Sie dienen insbesondere dazu, dass Informationen, die über INTERLIS hinausgehen, als Bestandteil der Modelldaten beigefügt werden können. Metaattribute haben einen Namen (Attribut "Name"), der innerhalb der Metaattribute desselben Metaelementes eindeutig sein muss, und den beschreibenden Wert (Attribut "Value").»

Im vorliegenden Dokument wird eine Syntax beschrieben, welche erlaubt, solche Meta-Attribute direkt als Teil des Modells formulieren zu können. Dies im Gegensatz zum Metamodell, wo definiert wird, wie Meta-Attribute als Daten definiert bzw. in XML transferiert werden.

2.2 Vorteile

Die Definition von Meta-Attributen direkt im Datenmodell hat den Vorteil, dass Metainformationen unmittelbar bei denjenigen INTERLIS-Konstrukten erfasst werden können, die sie beeinflussen sollen. Die vorliegende Spezifikation ist so konzipiert, dass Meta-Attribute *ohne* Änderung der INTERLIS-Spezifikation [1] angewendet werden können.

Im Unterschied zu normalem Zeilenkommentar, der ausschliesslich informativen Charakter hat, können Meta-Attribute gemäss den Syntaxregeln aus Abschnitt 4.2 von entsprechend befähigten Softwarewerkzeugen interpretiert und ggf. für die weitere Verarbeitung weitergegeben werden.

3 Konzept

Um die bestehende Sprachdefinition nicht ändern zu müssen, werden die Meta-Attribute innerhalb von INTERLIS-Zeilenkommentaren formuliert. Ein INTERLIS-Zeilenkommentar ist gemäss [1] wie folgt definiert:

«Ein Zeilenkommentar wird mit zwei Ausrufezeichen eröffnet, die unmittelbar aufeinander folgen. Der Zeilenkommentar wird durch das Zeilenende abgeschlossen.»

Um Meta-Attribute von echtem Kommentar unterscheiden zu können, ist als drittes Zeichen zwingend das Zeichen «@», einzufügen. Das heisst, ein Zeilenkommentar, der Meta-Attribute enthält, beginnt mit «! !@».

Zu welchem Meta-Element (gemäss INTERLIS 2-Metamodell) ein solches Meta-Attribut gehört, ergibt sich aus der Position des Zeilenkommentars. Das Meta-Attribut bezieht sich auf

das unmittelbar nachfolgende Sprachkonstrukt wie beispielsweise eine Modell-, Themen- oder Klassendefinition.

4 Syntax

4.1 Syntax-Definitionen

Die Syntaxregeln sind entsprechend der Extended Backus-Naur-Form (EBNF) definiert, wie auch die übrigen INTERLIS-Sprachregeln (siehe [1], Abschnitt 2.1).

Zusätzlich werden folgende Syntaxregeln eingeführt:

<code>~ A</code>	Ein Zeichen aus allen möglichen Zeichen aber verschieden von den durch A definierten	
<code>'A' .. 'D'</code>	Ein Zeichen aus dem Bereich von Zeichen: «A», «B», «C» und «D»	
<code>'\t'</code>	Tabulatorzeichen	Unicode: <CHARACTER TABULATION>
<code>'\f'</code>	Seitenvorschubzeichen	Unicode: <FORM FEED (FF)>
<code>'\r'</code>	Wagenrücklaufzeichen	Unicode: <CARRIAGE RETURN (CR)>
<code>'\n'</code>	Zeilenvorschubzeichen	Unicode: <LINE FEED (LF)>

4.2 Syntaxregeln für Meta-Attribute

Ein *Meta-Attribut* besteht aus dem einleitenden Code «!!@» sowie einem Meta-Attribut-Namen und einem zugewiesenen Meta-Attribut-Wert:

```
MetaAttribute = '!!@' Name '=' Value { ';' [ Name '=' Value ] } .
```

Ein *Meta-Attribut-Wert* wird entweder aus einem Namen oder einem freien String gebildet:

```
Value = Metaattributenamen | String .
```

Ein *Metaattributenamen* ist eine beliebige Zeichenfolge (max. 255 Zeichen Länge) ohne bestimmte Sonderzeichen:

```
Metaattributenamen = (* ~( '\t' | '\f' | '\r' | '\n' | ' ' | '=' | ';' | ',' | '"' | '\' ) *) .
```

Ein *String* ist eine beliebige Zeichenfolge inklusive «Escape»-Sequenzen; eingeschlossen in doppelte Anführungszeichen:

```
String = '"' { Escape | ~( '"' | '\' ) } '"' .
```

Eine *Escape-Sequenz* beginnt mit einem Backslash, gefolgt entweder von einem doppelten Anführungszeichen oder einem Backslash oder einer vierstelligen Hex-Unicode-Zeichenummer:

```
Escape = '\ ' ( '"' | '\ '  
            | 'u' HexDigit HexDigit HexDigit HexDigit ) .
```

Ein Unicode-Zeichen wird aus *Hexadezimalzahlen* gebildet. Diese bestehen aus Ziffern und den Buchstaben a/A bis f/F:

```
HexDigit = '0' .. '9' | 'a' .. 'f' | 'A' .. 'F' .
```

5 Positionierung innerhalb der INTERLIS-Sprache

In den folgenden Ausschnitten aus Syntaxregeln aus dem INTERLIS-Referenzhandbuch [1] wird definiert, an welcher Stelle Meta-Attribute vorkommen dürfen. Meta-Attribute an anderen Stellen werden als reguläre Zeilenkommentare aufgefasst und deshalb ignoriert.

Modell-Definition:

```
ModelDef = { MetaAttribute } [ 'CONTRACTED' ]  
          [ 'TYPE' | 'REFSYSTEM' | 'SYMBOLGY' ] 'MODEL' Model-Name ...
```

Topic-Definition:

```
TopicDef = { MetaAttribute } [ 'VIEW' ] 'TOPIC' Topic-Name ...
```

Klassen-Definition:

```
ClassDef = { MetaAttribute } 'CLASS' Class-Name ...
```

Struktur-Definition:

```
StructureDef = { MetaAttribute } 'STRUCTURE' Structure-Name ...
```

Attribut-Definition:

```
AttributeDef = { MetaAttribute } [ [ 'CONTINUOUS' ] 'SUBDIVISION' ]  
          Attribute-Name ...
```

Rollen-Definition in Assoziationen:

```
RoleDef = { MetaAttribute } Role-Name ...
```

Wertebereichs-Definition:

```
DomainDef = 'DOMAIN' { { MetaAttribute } Domain-Name ...
```

Aufzähltyp-Element:

```
EnumElement = { MetaAttribute } EnumElement-Name ...
```


Linienform-Definition:

```
LineFormTypeDef = 'LINE' 'FORM' { { MetaAttribute }  
  LineFormType-Name ...
```

Einheiten-Definition:

```
UnitDef = 'UNIT' { { MetaAttribute } Unit-Name ...
```

Definition eines Metaobjekt-Behälters:

```
MetaDataBasketDef = { MetaAttribute } ( 'SIGN' | 'REFSYSTEM' )  
  'BASKET' Basket-Name Properties <FINAL>  
  [ 'EXTENDS' MetaDataBasketRef ] '~' TopicRef  
  { 'OBJECTS' 'OF' Class-Name ':'  
    { MetaAttribute } MetaObject-Name  
    { ',', { MetaAttribute } MetaObject-Name } } ';'.  

```

Parameter-Definition:

```
ParameterDef = { MetaAttribute } Parameter-Name ...
```

Laufzeitparameter-Definition:

```
RunTimeParameterDef = 'PARAMETER' { { MetaAttribute }  
  RunTimeParameter-Name ...
```

Definition einer Konsistenzbedingung:

```
ConstraintDef = { MetaAttribute } ( MandatoryConstraint | ...
```

Funktions-Definition:

```
FunctionDef = { MetaAttribute } 'FUNCTION' Function-Name ...
```

Definition einer Sicht (View):

```
ViewDef = { MetaAttribute } 'VIEW' View-Name ...
```

Grafik-Definition:

```
GraphicDef = { MetaAttribute } 'GRAPHIC' Graphic-Name ...
```

Meta-Attribute zu einer Topic-Definition werden in den Schemadaten, das heisst, den Metaobjekten gemäss Metamodell [2] dem «SubModel» zugeordnet (und nicht der «DataUnit»).

Meta-Attribute zu einer Konsistenzbedingungs-Definition («ConstraintDef») sind als Schemadaten nicht möglich, weil Constraints gemäss Metamodell [2] keine OID haben beziehungsweise kein «MetaElement» sind.

6 Beispiel

Folgendes Beispiel soll den Gebrauch von Meta-Attributen illustrieren. Dabei wird angenommen, in einem INTERLIS-Datenmodell soll der verwendete Zeichensatz (ISO-8859-1) angegeben werden. Gemäss Syntaxregel in Kapitel 4 besteht das entsprechende Meta-Attribut aus folgenden Elementen:

Meta-Attribut-Element	INTERLIS-Codefragment
Zeilencode	!!@
Meta-Attribut-Name	ili.charset
Meta-Attribut-Wert	ISO-8859-1

Ein vollständiges, gültiges Beispiel sieht also INTERLIS-Code wie folgt aus:

```
INTERLIS 2.3;
!!@ ili.charset = ISO-8859-1
MODEL Example AT "mailto:noreply@localhost" VERSION "2010-02-30" =
END Example.
```

Das Meta-Attribut würde einem Objekt der Klasse «Model» aus dem Metamodell mit dem Namen «Example» zugeordnet:

```
<IlisMeta07.ModelData BID="MODEL.Example">
  <IlisMeta07.ModelData.Model TID="Example">
    <Name>Example</Name>
    <iliVersion>2.3</iliVersion>
    <Contracted>>false</Contracted>
    <Kind>NormalM</Kind>
    <At>mailto:noreply@localhost</At>
    <Version>2010-02-30</Version>
  </IlisMeta07.ModelData.Model>
  <IlisMeta07.ModelData.MetaAttribute
    TID="Example.METAOBJECT.ili.charset">
    <Name>ili.charset</Name>
    <Value>ISO-8859-1</Value>
    <MetaElement REF="Example" />
  </IlisMeta07.ModelData.MetaAttribute>
</IlisMeta07.ModelData>
```

7 Haftungsausschluss/Hinweise auf Rechte Dritter

eCH-Standards, welche der Verein **eCH** dem Benutzer zur unentgeltlichen Nutzung zur Verfügung stellt, oder welche **eCH** referenziert, haben nur den Status von Empfehlungen. Der Verein **eCH** haftet in keinem Fall für Entscheidungen oder Massnahmen, welche der Benutzer auf Grund dieser Dokumente trifft und / oder ergreift. Der Benutzer ist verpflichtet, die Dokumente vor deren Nutzung selbst zu überprüfen und sich gegebenenfalls beraten zu lassen. **eCH**-Standards können und sollen die technische, organisatorische oder juristische Beratung im konkreten Einzelfall nicht ersetzen.

In **eCH**-Standards referenzierte Dokumente, Verfahren, Methoden, Produkte und Standards sind unter Umständen markenrechtlich, urheberrechtlich oder patentrechtlich geschützt. Es liegt in der ausschliesslichen Verantwortlichkeit des Benutzers, sich die allenfalls erforderlichen Rechte bei den jeweils berechtigten Personen und/oder Organisationen zu beschaffen.

Obwohl der Verein **eCH** all seine Sorgfalt darauf verwendet, die **eCH**-Standards sorgfältig auszuarbeiten, kann keine Zusicherung oder Garantie auf Aktualität, Vollständigkeit, Richtigkeit bzw. Fehlerfreiheit der zur Verfügung gestellten Informationen und Dokumente gegeben werden. Der Inhalt von **eCH**-Standards kann jederzeit und ohne Ankündigung geändert werden.

Jede Haftung für Schäden, welche dem Benutzer aus dem Gebrauch der **eCH**-Standards entstehen ist, soweit gesetzlich zulässig, wegbedungen.

8 Urheberrechte

Wer **eCH**-Standards erarbeitet, behält das geistige Eigentum an diesen. Allerdings verpflichtet sich der Erarbeitende sein betreffendes geistiges Eigentum oder seine Rechte an geistigem Eigentum anderer, sofern möglich, den jeweiligen Fachgruppen und dem Verein **eCH** kostenlos zur uneingeschränkten Nutzung und Weiterentwicklung im Rahmen des Vereinszweckes zur Verfügung zu stellen.

Die von den Fachgruppen erarbeiteten Standards können unter Nennung der jeweiligen Urheber von **eCH** unentgeltlich und uneingeschränkt genutzt, weiterverbreitet und weiterentwickelt werden.

eCH-Standards sind vollständig dokumentiert und frei von lizenz- und/oder patentrechtlichen Einschränkungen. Die dazugehörige Dokumentation kann unentgeltlich bezogen werden.

Diese Bestimmungen gelten ausschliesslich für die von **eCH** erarbeiteten Standards, nicht jedoch für Standards oder Produkte Dritter, auf welche in den **eCH**-Standards Bezug genommen wird. Die Standards enthalten die entsprechenden Hinweise auf die Rechte Dritter.

Anhang – Referenzen & Bibliographie

- [1] eCH-Fachgruppe Geoinformation (2006): *eCH-0031: Geoinformation: INTERLIS 2 – Referenzhandbuch*. eCH-Standard
- [2] KOGIS (2008): *INTERLIS 2-Metamodell*.
Online http://www.interlis.ch/interlis2/docs23/ili2-metamodel_2008-08-29_d.pdf