

XML ja SAS

Mitä tänä päivänä voidaan
SASista tehdä XML-muotoon

Sisältö

- XML
- SAS ja XML:
 - ODS:n käyttäminen xml-tulosteisiin
 - Libname enginen käyttäminen XML tulosteisiin
- XML mapperi
- Libname enginen käyttäminen XML tietojen lukemiseen

XML

- XML on tekstipohjainen tapa esittää strukturoitua informaatiota.
- Pohjautuu SGML:ään (kuten HTMLkin).
- XML on tapa välittää tietoa sovellusten, alustojen, ihmisten, tietokoneiden jne välillä.
- XML on tekniikka / teknologiaperhe.

XBRL

- XBRL on taloudelliseen raportointiin kehitetty 'murre', joka vastaa IFRS standardia.
- iXBRL on 'parannettu' versio XBRL:stä.
- XBRL käytetään EBA raportointiin, verotiedot yrityksiltä tulevaisuudessa XBRL / iXBRL muodossa viranomaisille.

XML:n perus syntaksi

- Jokainen XML dokumentti alkaa vähintään versiolla, mielellään myös encoding-arvolla
<?xml version="1.0" encoding="ISO-8859-1"?>
- Jokainen aloitettu tagi pitää lopettaa.
- XML dokumentissa on vain yksi juuri.
- Jokaisella juurella voi olla kuinka monta lasta tahansa.
- Tagilla ei tarvitse olla attribuutteja, sillä voi olla yksi attribuutti tai monta attribuuttia.

Esimerkki XML:stä

```
<part number="1976">
```

```
<name>Windscreen Wiper</name>
```

```
<description>The Windscreen wiper automatically  
removes rain from your windscreen, if it should  
happen to splash there. It has a rubber
```

which can be ordered separately if you need to
replace it.

```
</description>
```

```
<ref part="1977">blade</ref>
```

```
</part>
```

Attribuutti vai elementti?

<kirje>

<aika><pv>10</pv><kk>10</kk><v>2017</v></aika>

<vastaanottaja>Tuija</vastaanottaja>

<laatija>Jani</laatija>

<otsikko>Muistutus</otsikko>

<viesti>Ethän minua unhoita!</viesti>

</kirje>

XML nimeämissäännöt

- Tagien pitää alkaa kirjaimilla a-z, A-Z tai alaviivalla. Tämän jälkeen voi tulla kirjaimia, numeroita, väliviivoja, pisteitä, kaksoispisteitä ja alaviivoja.
- XML nimet eivät saa sisältää välejä.
- Nimissä ei saa olla XML alkuliitettä.
- Käytä mieluummin kuvaavia nimiä.
- Elementit Upper Camel Case, attrib. lower

XML jäsentäjä

- XML jäsentäjä (parser) on väline, joka osaa muodostaa rakenteen XML-tiedostosta.
- Jäsentäjä tarkistaa, että XML on ehyt (well formed):
 - Alku ja loppumerkinnyt paikoillaan.
 - Elementit aidosti sisäkkäisiä.

Encoding

- XML-dokumentin alussa kerrotaan, missä merkistössä dokumentti on talletettu.
- ISO-8859-1 eli ISO Latin 1 sisältää ääkköset, Unicode (UTF-8) on sen laajennus.
- UTF-8:ssa ja ISO-8859-1:ssä ääkköset mappautuvat eri lailla.

Standardoimaton standardi?

- Yleiskäyttöisten XML-sovellusten lisääminen johtaa tilanteeseen, jossa halutaan varata jotkut nimet määriteltyyn käyttöön.
- XML-nimiavaruudet on tekniikka, jossa voidaan määritellä attribuuttien ja elementtien yksikäsitteinen käyttö.

Namespace

- Nimiavaruus tarkoittaa dokumentissa käytössä olevaa sanastoa ja sen määrittämiä.
- Synonyymit eivät sekoitu dokumentin sisällä.

Schema

- XML-kielinen.
- Tukee nimiavaruuksia.
- Upotettu dokumentaatio.
- Käyttäjän omat tietotyypit ja määritysten perintä.
- Tiedostopäätteenä .xsd

Schema

- Scheman avulla välitetään tiedon rakenne sovellusriippumattomasti.

Schema rakenne

XML määrittelykset

Juuri ja nimiavaruus, jota käytetään

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="BuyerCommunicationDetails">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="BuyerPhoneNumberIdentifier"
type="genericStringType035" minOccurs="0" />
        <xs:element name="BuyerEmailAddressIdentifier"
type="genericStringType070" minOccurs="0" />
      </xs:sequence>
    </xs:complexType>
  </xs:element> ...
```

Elementin määrittelykset

Rakenteinen tyyppi

Valinnaisia

Peräkkäiset elementit
sequencen sisään, ALL tagi
jos järjestyksellä ei väliä

Upotettu määritelmä

SAS ja XML

Miten SASilla luetaan ja tuotetaan
XML-dataa

ODSn käyttäminen

- Suositeltavaa silloin, kun raportointiproseduurin tulokset halutaan XML muotoon.
- Tagsettien käytön helppous.
- Tuttua muun SAS käytön perusteella.
- Haluttu tulos voi olla report- tai tabulate-proseduurilla tuotettua!

ODS XML tulostuskohde

- ODS:n käyttö yksinkertaisimmillaan tarkoittaa ODS XML-kohteen avaamista ja sulkemista.

```
ods xml file='c:\temp\demoxml.xml' ;
```

```
proc print data=sashelp.class; run;
```

```
ods xml close;
```

```
<?xml version="1.0" encoding="UTF-8"?>
- <odsxml>
  - <head>
    <meta operator="virtvir"/>
  </head>
  - <body>
    - <proc name="Print">
      <label name="IDX"/>
      <title toc-level="1" unformatted_width="0" unformatted_type="string" class="systemtitle"
        type="string">The SAS System</title>
      - <branch name="Print" toc-level="1" class="contentprocname" label="The Print Procedure">
        - <leaf name="Print" toc-level="2" class="contentitem" label="Data Set SASHELP.CLASS">
          - <output name="Print" label="Data Set SASHELP.CLASS" clabel="Data Set
            SASHELP.CLASS">
            - <output-object class="table" type="table">
              - <style>
                <border frame="box" rules="groups" padding="7" spacing="1"/>
              </style>
              - <colspecs columns="6">
                - <colgroup>
                  <colspec name="Obs" type="double" align="right" width="2"
                    column="1"/>
                </colgroup>
                - <colgroup>
                  <colspec name="Name" type="string" width="7" column="2"/>
                </colgroup>
              </colspecs>
            </output-object>
          </leaf>
        </branch>
      </proc>
    </body>
  </odsxml>
</xml>
```

ODS:n käyttäminen XML:n tekoon

Toinen tapa käyttää ods:ää XML:n tekoon on ods markup tai tagsets-vaihtoehto.

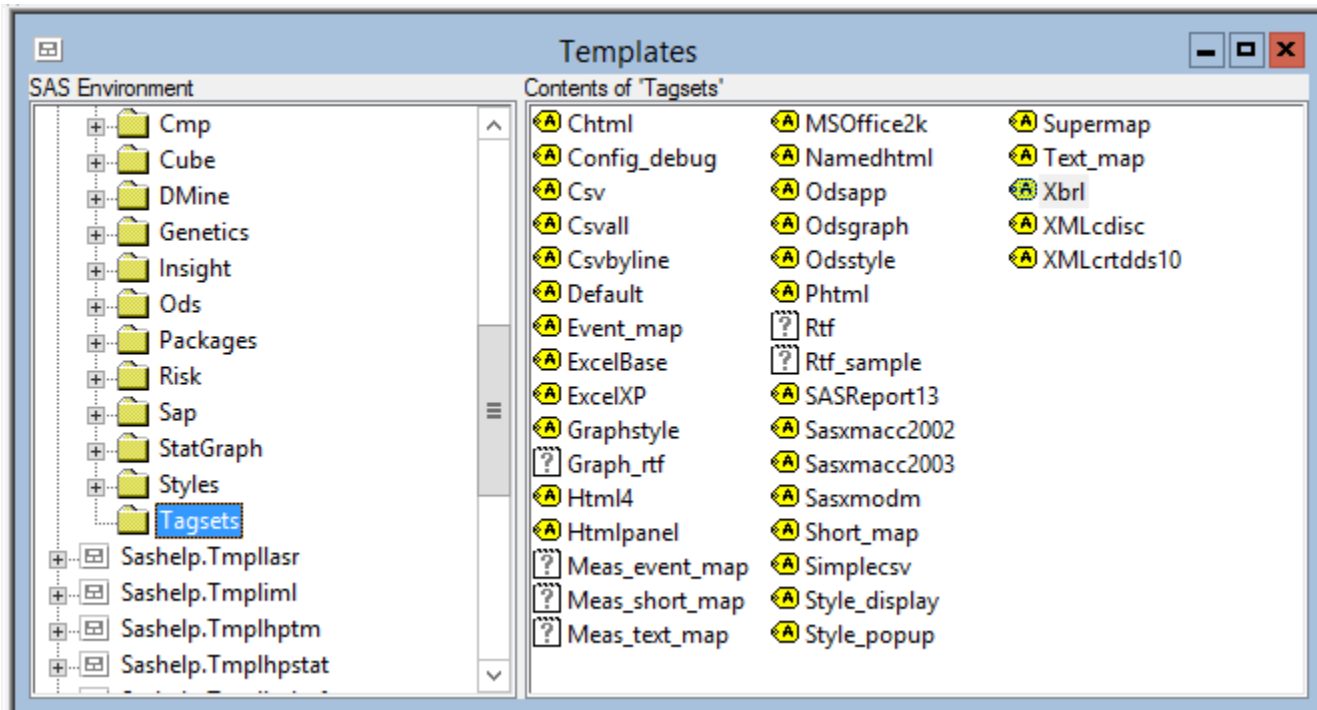
```
ods markup file='c:\temp\demo.xml';  
proc print data=sashelp.class; run;  
ods markup close;
```

- Tagset voidaan määritellä, jos halutaan esim xblr tai cdisc muotoa. Tai oma.

Tagsettien tarkastelu

- Tagsettejä voi tarkastella avaamalla odstemplates-komennolla template-ikkunan komentoriviltä.
- Template-ikkuna saadaan esiin tagsets-valinnalla pikavalikosta results-ikkunasta.

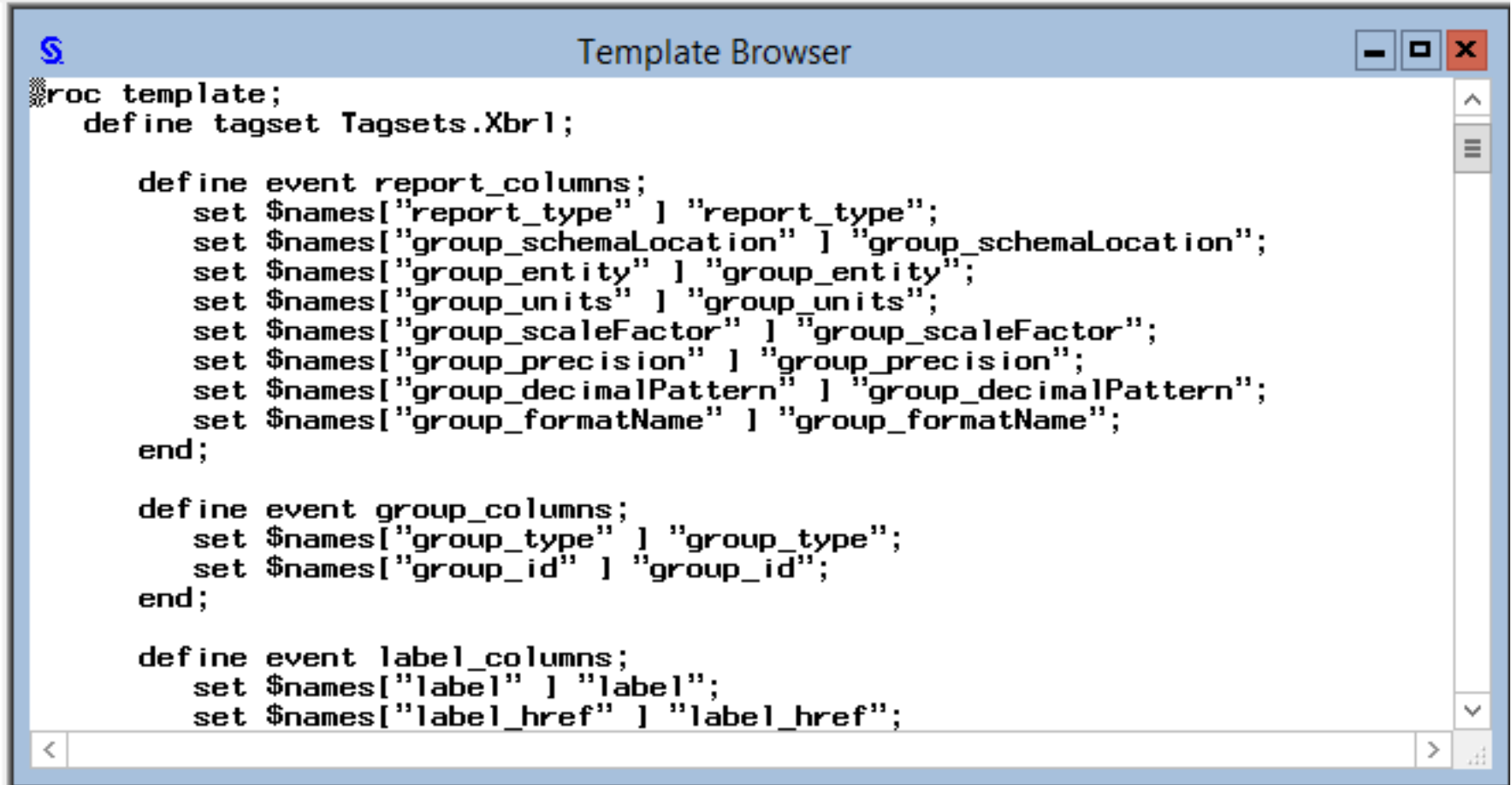
Tagsetit



Tagsetin rakenne

- Proc template – proseduuri, jolla luodaan tagsetti.
- Define tagset tagsets.demo; - määritellään
- Default_event = 'basic'; jos ei muuta tehdä
- Define event doc; ... end;
- Kaikki tarvittavat eventit pitää määritellä, muuten tehdään default_event.
- Laukaisut ja toiminnot ehdollisesti.

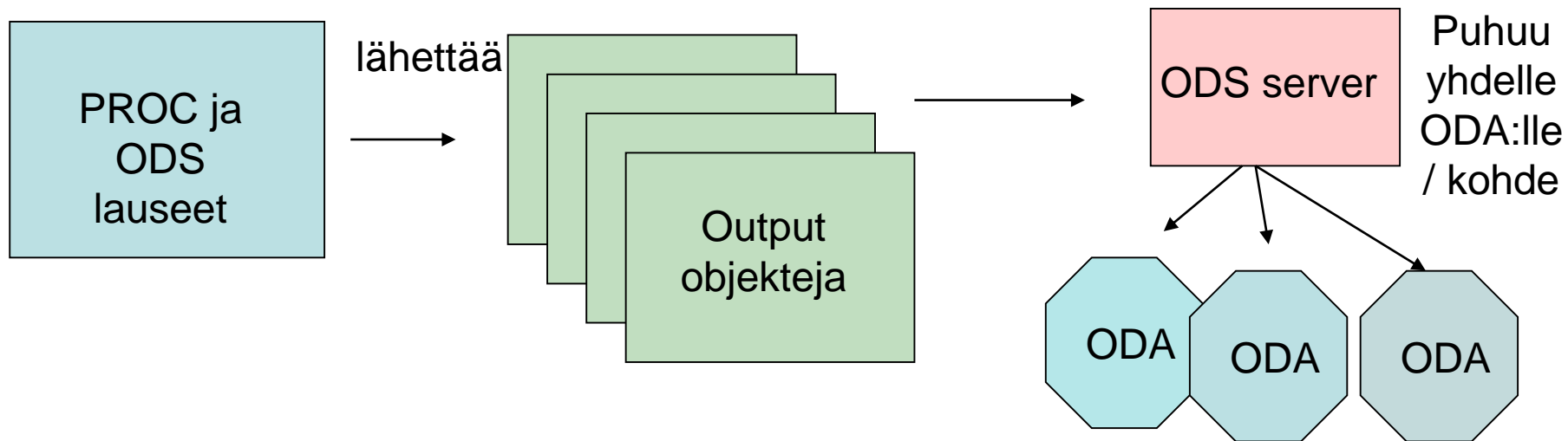
Xblr tagsetti



```
proc template;  
  define tagset Tagsets.Xblr;  
  
    define event report_columns;  
      set $names["report_type" ] "report_type";  
      set $names["group_schemaLocation" ] "group_schemaLocation";  
      set $names["group_entity" ] "group_entity";  
      set $names["group_units" ] "group_units";  
      set $names["group_scaleFactor" ] "group_scaleFactor";  
      set $names["group_precision" ] "group_precision";  
      set $names["group_decimalPattern" ] "group_decimalPattern";  
      set $names["group_formatName" ] "group_formatName";  
    end;  
  
    define event group_columns;  
      set $names["group_type" ] "group_type";  
      set $names["group_id" ] "group_id";  
    end;  
  
    define event label_columns;  
      set $names["label" ] "label";  
      set $names["label_href" ] "label_href";  
    end;  
  end;  
end;
```


ODS käsittely ja eventit

- Eventit linkataan dokumenttihierarkiaan.
- Tagsettien käyttö voi olla hyvinkin raskasta, sillä ODS käyttää Output Delivery Agent (ODA):ia hallinnoimaan output objekteja ja dokumentin hierarkiaa.



XML libname

Datan kirjoittaminen ja lukeminen
SASilla

XML libname engine

- XML tiedostoa luetaan SASissa peräkkäin.
- XML kirjastoviittauksella yhdistettyyn dataan ei voi tehdä sorttausta tai lukea vain määrättyjä havaintoja.
- Sort ja SQL sorted by käsittelyä varten kannattaa datasta tehdä workkitiedosto.

Miten SASilla kirjoitetaan XML:ää

- XML:ää voidaan lukea ja kirjoittaa XML enginen kautta

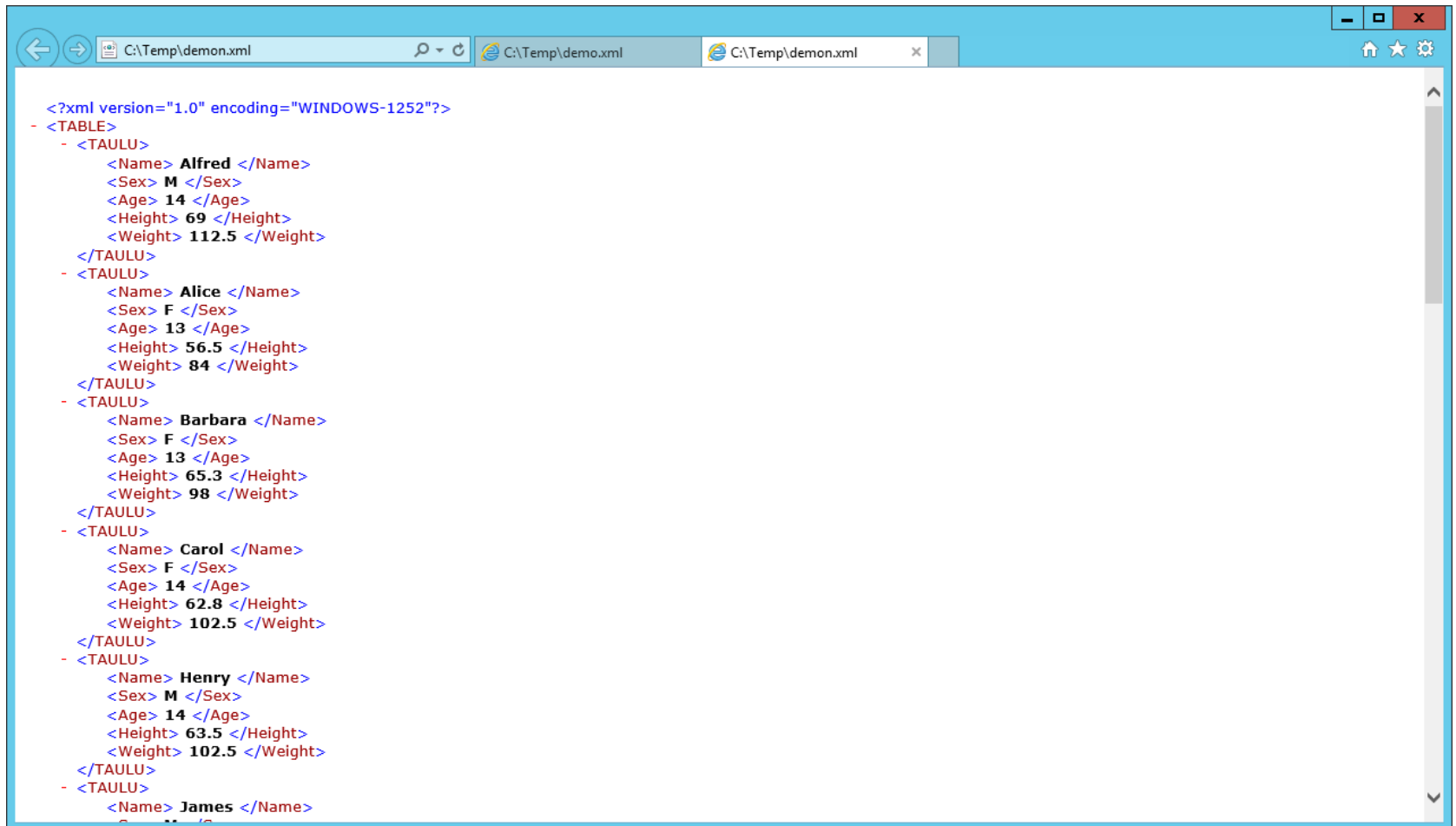
```
Libname esim xml 'c:\temp\demon.xml';
```

```
Data esim.taulu;
```

```
Set sashelp.class;
```

```
Run;
```

XML muotoinen data



The image shows a web browser window with three tabs, all displaying the same XML file located at C:\Temp\demon.xml. The XML content is as follows:

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<TABLE>
  <TAULU>
    <Name> Alfred </Name>
    <Sex> M </Sex>
    <Age> 14 </Age>
    <Height> 69 </Height>
    <Weight> 112.5 </Weight>
  </TAULU>
  <TAULU>
    <Name> Alice </Name>
    <Sex> F </Sex>
    <Age> 13 </Age>
    <Height> 56.5 </Height>
    <Weight> 84 </Weight>
  </TAULU>
  <TAULU>
    <Name> Barbara </Name>
    <Sex> F </Sex>
    <Age> 13 </Age>
    <Height> 65.3 </Height>
    <Weight> 98 </Weight>
  </TAULU>
  <TAULU>
    <Name> Carol </Name>
    <Sex> F </Sex>
    <Age> 14 </Age>
    <Height> 62.8 </Height>
    <Weight> 102.5 </Weight>
  </TAULU>
  <TAULU>
    <Name> Henry </Name>
    <Sex> M </Sex>
    <Age> 14 </Age>
    <Height> 63.5 </Height>
    <Weight> 102.5 </Weight>
  </TAULU>
  <TAULU>
    <Name> James </Name>
```

Encoding

- Luotavan XML tiedoston encoding-arvo voidaan määrittää kirjastoviittauksen yhteydessä.
- Mikäli encoding-arvo puuttuu, on se utf-8.
- Windows-ympäristössä on oletusarvona windows-1252.
- XMLENCODING=utf8 voidaan asettaa arvo utf-8:ksi.
- XMLENCODING=latin1 tekee iso-8859-1 koodauksen.

Metatiedot erikseen

- On mahdollista tallettaa Schema-tiedot erilliseen tiedostoon XML-datan luonnin yhteydessä:

```
filename xsd 'c:\temp\XML\luokka.xsd';  
libname output xml 'c:\temp\XML\luokka.xml'  
xmltype=msaccess  
    xmlmeta=schemadata xmlschema=xsd;  
data output.luo;  
set sashelp.class;  
run;  
ods _all_ close;
```

xmlv2

Mahdollisuus tehdä koneesta riippumatonta
XML:ää

```
libname outputti xmlv2 'c:\temp\XML\luokkav2.xml' ;
```

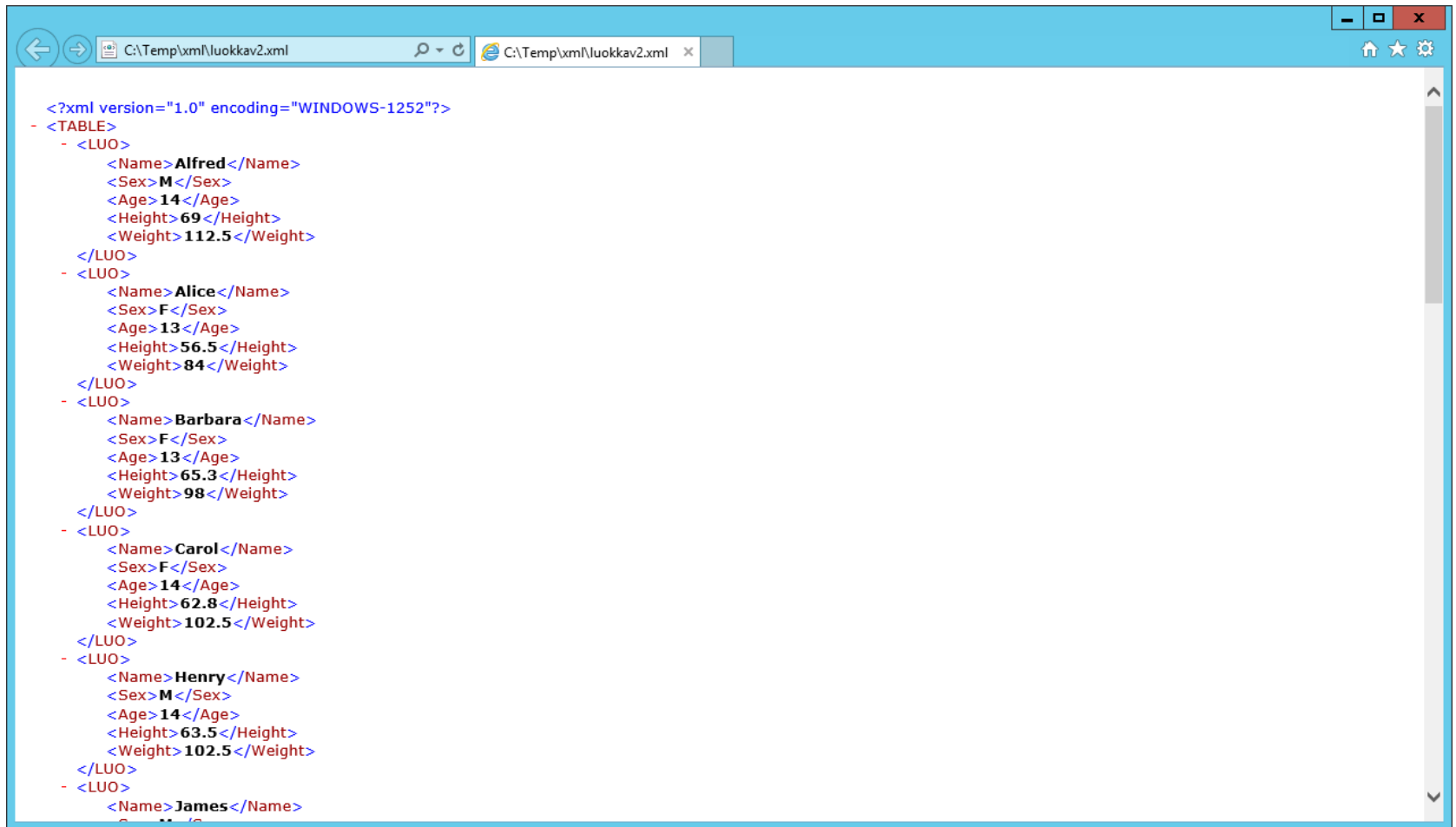
```
data outputti.luo;
```

```
set sashelp.class;
```

```
run;
```

```
ods _all_ close;
```


xmlv2



```
<?xml version="1.0" encoding="WINDOWS-1252"?>
- <TABLE>
  - <LUO>
    <Name>Alfred</Name>
    <Sex>M</Sex>
    <Age>14</Age>
    <Height>69</Height>
    <Weight>112.5</Weight>
  </LUO>
  - <LUO>
    <Name>Alice</Name>
    <Sex>F</Sex>
    <Age>13</Age>
    <Height>56.5</Height>
    <Weight>84</Weight>
  </LUO>
  - <LUO>
    <Name>Barbara</Name>
    <Sex>F</Sex>
    <Age>13</Age>
    <Height>65.3</Height>
    <Weight>98</Weight>
  </LUO>
  - <LUO>
    <Name>Carol</Name>
    <Sex>F</Sex>
    <Age>14</Age>
    <Height>62.8</Height>
    <Weight>102.5</Weight>
  </LUO>
  - <LUO>
    <Name>Henry</Name>
    <Sex>M</Sex>
    <Age>14</Age>
    <Height>63.5</Height>
    <Weight>102.5</Weight>
  </LUO>
  - <LUO>
    <Name>James</Name>
```

Elementti tai attribuutti

- Data voidaan kirjoittaa joko elementtinä tai attribuuttina.

XMLDATAFORM=ELEMENT

- Elementtinä

```
<Name> Alfred </Name>
```

- Attribuuttina

```
<COLUMN name="Name" value="Alfred" />
```

XML mapper

- Jos XML tiedosta on olemassa Schema, skeema, voi tiedon rakennetta tarkastella XML mapperin avulla.
- Pohjana voi käyttää myös XML tietoa.
- XML tiedosto ja Schema voivat sisältää useampia 'taulukkoita'.

XML mapper

SAS XML Mapper

File Tools Help

Condensed Full Schema

<no data> [1] (1)

Properties

Name: SXLEMAP

Description:

Path:

End Path: Begin/End

Retain

SXLEMAP

[None]

Namespaces

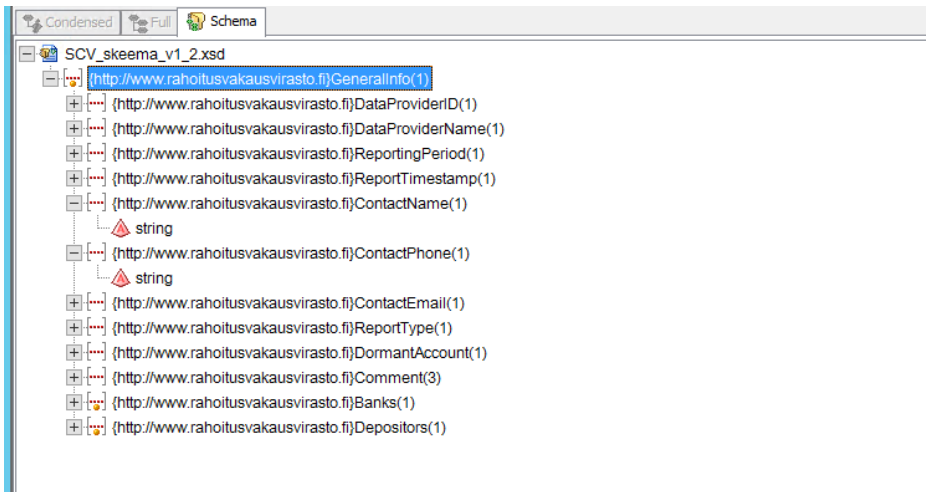
XML source XML Schema source XMLMap SAS Code Example Table view Contents Validate Log

<no file>

Tyhjässä mapperissa kannattaa avata joko XML tiedosto tai schema-tiedosto, jotta map voidaan muodostaa

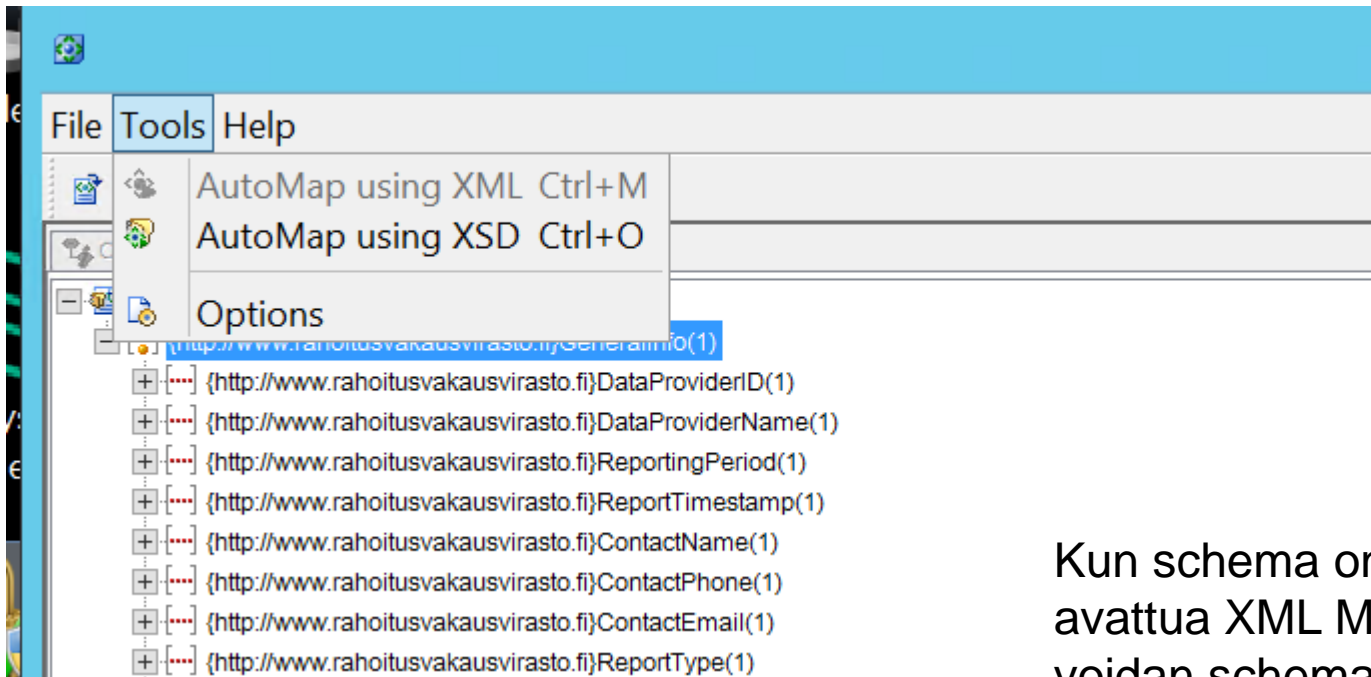
XML Mapper initialization: application mode

Tiedot mapperissa



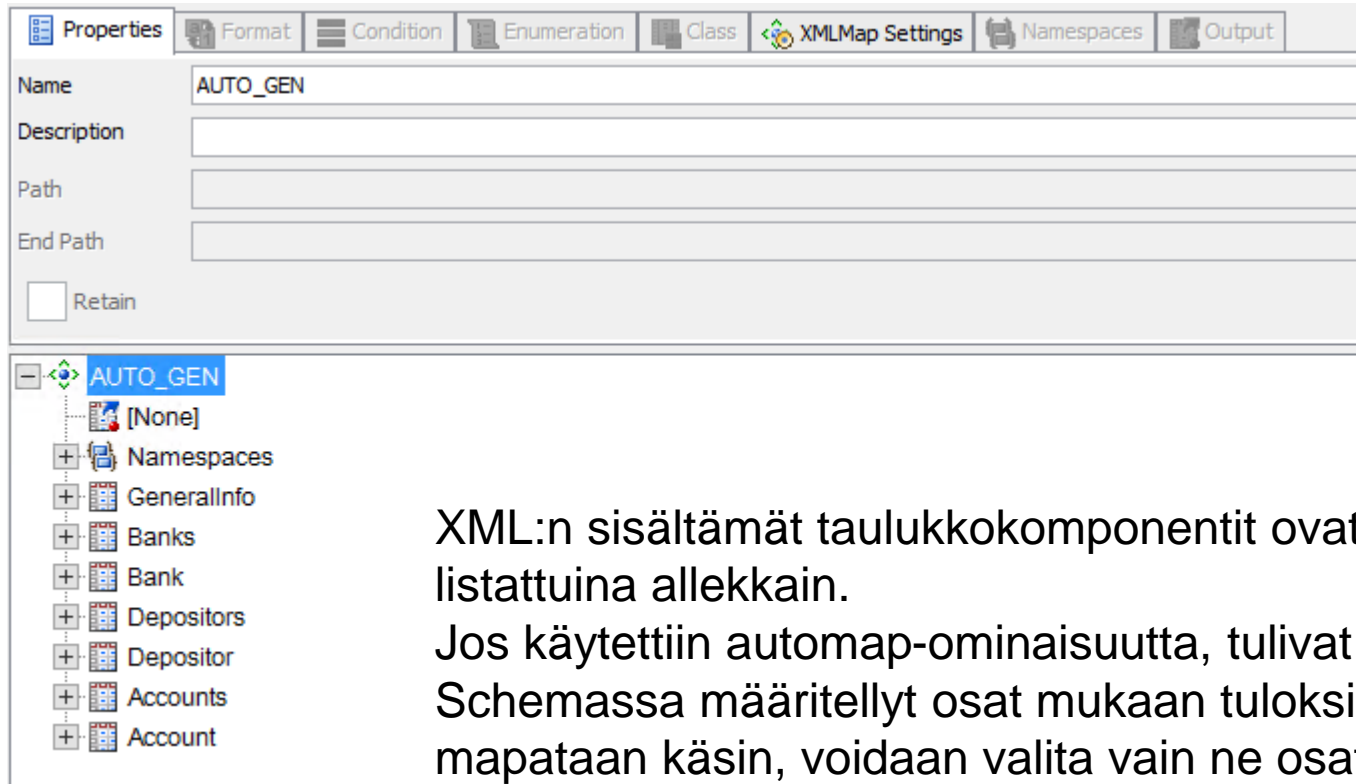
Scheman rakenneosat löytyvät vasemmassa ylälaudassa olevasta ikkunasta. Tässä voidaan tarkastella joko XML pohjaisesti tai Schema-pohjaisesti luotua kuvaa XML tiedon rakenteesta.

Scheman perusteella työskentely



Kun schema on saatu avattua XML Mapperiin, voidaan scheman perusteella luoda MAP-tiedosto valitsemalla Tools / AutoMap using XSD.

Automapin tulokset



The screenshot shows the Automap application interface. The top toolbar includes buttons for Properties, Format, Condition, Enumeration, Class, XMLMap Settings, Namespaces, and Output. The Properties window is open, showing the following fields:

- Name: AUTO_GEN
- Description: (empty)
- Path: (empty)
- End Path: (empty)
- Retain:

Below the Properties window is a tree view showing the project structure:

- AUTO_GEN
 - [None]
 - Namespaces
 - GenerallInfo
 - Banks
 - Bank
 - Depositors
 - Depositor
 - Accounts
 - Account

XML:n sisältämät taulukkokomponentit ovat listattuina allekkain.

Jos käytettiin automap-ominaisuutta, tulivat kaikki Schemassa määritellyt osat mukaan tuloksiin, jos mapataan käsin, voidaan valita vain ne osat XML tiedostosta, joista ollaan kiinnostuneita.

Taulun rakenne

Properties Format Condition Enumeration

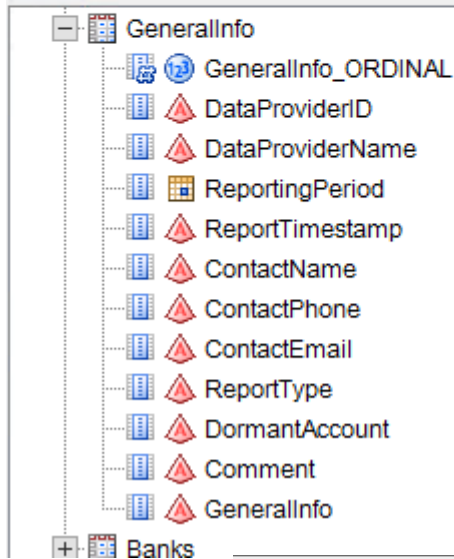
Name: AUTO_GEN

Description:

Path:

End Path:

Retain



XML:n perusteella luotu rakennekuva sisältää metadatan taulusta, sarakkeiden nimet ja tyypit.
Formaatit ja ehtorakenteet

Properties Format Condition Enumeration Class XMLMap Settings Namespaces Output

Type: character Length: 32

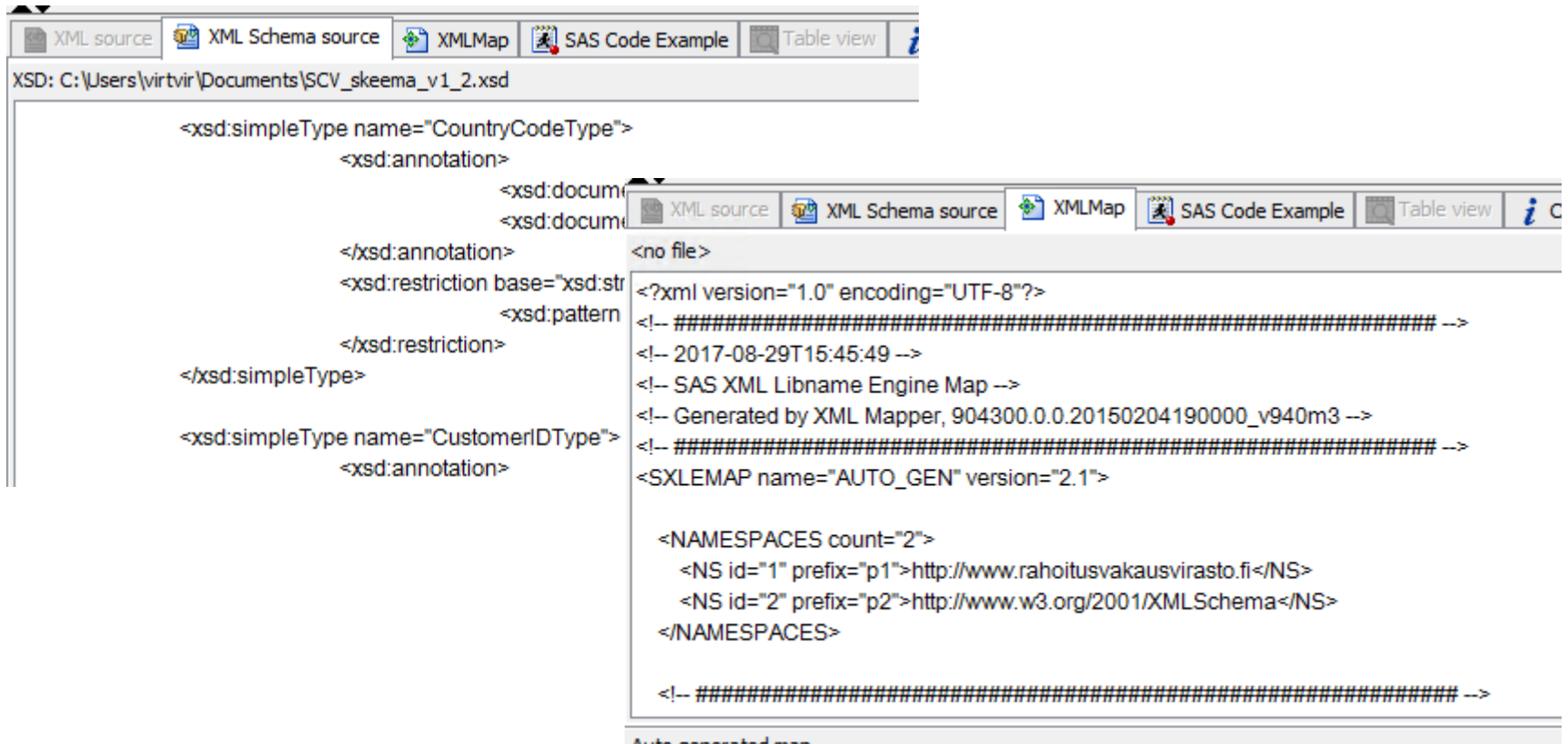
Data type: string

Name: Width: 0 NDec: 0

Informat: Width: 0 NDec: 0

Näkyvyys XML:ään

Kuvaruudun alalaidassa on mahdollista tarkastella Scheman tietoja tai vastaavasti siitä muodostetun Mapin tietoja tai SAS koodia, jolla mappi ja kirjastoviittaus luotaisiin.



The screenshot displays two overlapping windows from the SAS software interface. The top window, titled 'XSD: C:\Users\virtvir\Documents\SCV_skeema_v1_2.xsd', shows an XML Schema Definition (XSD) for 'CountryCodeType'. The visible code includes an annotation, a restriction based on 'xsd:string', and a pattern. The bottom window, titled '<no file>', shows the SAS code generated by the XML Mapper. This code includes an XML declaration, a comment with a timestamp and version, namespace declarations for 'http://www.rahoitusvakuusvirasto.fi' and 'http://www.w3.org/2001/XMLSchema', and another comment with the 'SXLEMAP' name and version.

```
<xsd:simpleType name="CountryCodeType">
  <xsd:annotation>
    <xsd:documentation>
      <xsd:documentation>
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[A-Z]{2}" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="CustomerIDType">
  <xsd:annotation>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ##### -->
<!-- 2017-08-29T15:45:49 -->
<!-- SAS XML Libname Engine Map -->
<!-- Generated by XML Mapper, 904300.0.0.20150204190000_v940m3 -->
<!-- ##### -->
<SXLEMAP name="AUTO_GEN" version="2.1">

  <NAMESPACES count="2">
    <NS id="1" prefix="p1">http://www.rahoitusvakuusvirasto.fi</NS>
    <NS id="2" prefix="p2">http://www.w3.org/2001/XMLSchema</NS>
  </NAMESPACES>

  <!-- ##### -->
```

SAS koodi

```
<no file>
Show
 PROC Datasets  PROC Contents  PROC Print  Copy to WORK

/*****
* Generated by XML Mapper, 904300.0.0.20150204190000_v940m3
*****/

/*
* Environment
*/
filename SXLELIB '<fileName>.xml';
filename SXLEMAP 'C:\Users\virtvir\Documents\talsu_skeema.map';
libname SXLELIB xmlv2 xmlmap=SXLEMAP access=READONLY;
```

SAS koodi, jolla XLM data voidaan ottaa esille XLM kirjastoviittauksen kautta on nähtävissä suoraan koodi-ikkunassa. Koodissa kerrotaan XML tiedoston sijainti, map:in sijainti ja tehdään kirjastoviittaus, jolla saadaan XML data käyttöön SASissa.

Ulkopuolisen tiedon lukeminen

- URL tai FTP metodilla on mahdollista lukea XML tiedostoja:

```
filename NHL url http://sww.sas.com/NHL.xml';  
filename MAP '/tmp/NHL.map';  
libname NHL xml xmlmap=MAP;  
proc copy indd=nhl outdd=work;  
select NHL;  
run;
```

Hierarkioiden toteuttaminen

- Yksinkertainen hierarkia voi olla helpointa toteuttaa kirjoittamalla XML tiedosto PUT-lauseilla tekstitiedostoon.
- Tekstitiedostojen yhdistäminen mahdollistaa hierarkioiden luomisen.

Puttaaminen

```
filename vanha 'c:\temp\konsti.xml';  
DATA _NULL_ ;  
SET sashelp.class end=loppu;  
FILE vanha;  
if _n_ = 1 then put '<xml>';  
PUT '<lapsi>';  
    PUT '<name>' name '</name>';  
    PUT '<age>' age '</age>';  
    PUT '<height>' height '</height>';  
    PUT '<weight>' weight '</weight>';  
    PUT '</lapsi>';  
    if loppu then put '</xml>';  
RUN;
```

Itse tehtyjen tagsettien käyttö

- On mahdollista tehdä itse tagsettejä, joilla voidaan toteuttaa haluttuja hierarkioita ja elementtien nimeämisiä.
- Itse tehtyjen tagsettien kanssa kannattaa olla tarkka, jos kaikki osakokonaisuudet eivät ole tuttuja. (XML, data, engine, yms)