

Bir Veri Saklama Biçimi Olarak XML

Gürer Özen
<madcat@e-kolay.net>

Bu belgeyi Free Software Foundation tarafından yayınlanmış bulunan GNU Özgür Belgelendirme Lisansının 1.2 ya da daha sonraki sürümünün koşullarına bağlı kalarak kopyalayabilir, dağıtabilir ve/veya değiştirebilirsiniz. Lisans'ın bir kopyasını <http://www.gnu.org/copyleft/gfdl.html> adresinde bulabilirsiniz.

Bu belgedeki bilgilerin kullanımından doğacak sorumluluklar ve olası zararlardan belge yazarı sorumlu tutulamaz. Bu belgedeki bilgileri uygulama sorumluluğu uygulayana aittir.

Tüm telif hakları aksi özellikle belirtilmediği sürece sahibine aittir. Belge içinde geçen herhangi bir terim, bir ticari isim ya da kuruma itibar kazandırma olarak algılanmamalıdır. Bir ürün ya da markanın kullanılmış olması ona onay verildiği anlamında görülmemelidir.



Extensible Markup Language (XML), XML metinleri adı verilen veri yapılarını tanımlamaya ve yaratmaya yarayan, SGML'den türemiş bir anlamlandırma dilidir.

Tasarlanma amacı

- Internet üzerinde kullanılabilen
- Çeşitli uygulamalara uygun
- SGML ile uyumlu
- Programlar tarafından kolayca işlenebilen
- İnsanlar tarafından okunup anlaşılabilen
- Dillerin ve metinlerin hızla hazırlanabildiği bir dil yaratmak.

W3C (World Wide Web Consortium) XML Working Group tarafından 1996 da geliştirildi.

XML dökümanı, metinden (character data) ve anlamlandırma etiketlerinden (markup) oluşur:

Merhaba < isim > Memet < / isim > nasılsın?

Metin içerisinde özel karakterler varlık (entity) adı verilen elemanlar yardımıyla kullanılabilir:

<açıklama> a < b matematikte a, b'den küçüktür anlamına gelir </açıklama>

Önceden tanımlı varlıklar:

&	&amp;	<	&lt;	>	&gt;
“	&quot;	'	&apos;		

Karakterler sayısal kodları ile de çağırılabilirler:

Ӓ ****

Metin içerisindeki boşlukların (boşluk karakteri, tab, satır dönüşleri, vs) anlam taşıyıp taşımadığı dökümanı kullanacak olan uygulamaya bağlıdır. İmler içinde boşluklar özel bir anlam taşımaz, birbirleri yerine kullanılabilir.

Metin uygulamaya verilirken satır sonları için aşağıdaki çevrim yapılır:

\r\n **->** **\n**

Etiketler (tag) metnin istediğimiz kısımlarına isimler ve özellikler atamamızı sağlarlar:

BASIC dilinde <komut> PRINT </komut> ile ekrana yazı yazdırabilirsiniz.

** Linux Şenliği Web Sayfası **

<resim dosya='foto.jpg' parlaklık='1.2' />

XML metni içinde okuyanlara yönelik açıklamalar konabilir:

<!-- bu bir açıklamadır -->

İşlem komutları (PI – Processing Instruction) adı verilen etiketler metnin parçası olmamakla birlikte uygulamalara aynen aktarılır. XML adıyla başlayan işlem komutları standarda aittir ve özel bir tanesi döküman başlangıcı olarak kullanılır:

<?xml version='1.0' encoding='UTF-8' standalone='yes'?>

Döküman başlangıcında şu parametreler verilebilir:

version	dökümanın hangi XML standardına uyduğunu gösterir
encoding	dökümanda kullanılan karakter setini belirtir
standalone	dökümanın doğru işlenebilmesi için harici bir döküman tipi tanımlı gerekli mi?

Karakter Setleri

UNICODE, kar amacı gütmeyen Unicode Consortium organizasyonu tarafından geliştirilen ve her karaktere bir sayı değeri karşılığı tanımlayan bir standarttır. Evrensel Karakter Seti (UCS) olarak bilinen ISO/IEC 10646 standardı her iki organizasyonun işbirliği ile aynı sayısal karşılıkları taşımaktadır.

Unicode kodlarından oluşan karakter dizilerini bilgisayarda verimli bir şekilde saklayabilmek amacıyla bazı karakter kodlamaları (encoding) geliştirilmiştir:

UTF-8 kodlaması UCS karakterlerini 1 ile 6 byte uzunluğunda diziler halinde kodlar. ASCII kodlaması içinde kalan 0 ile 127 arasındaki karakterler aynen kendi kodları ile kullanılır. Diğerleri byte dizileri haline gelir:

UCS karakterleri	UTF-8 byte dizileri
0000 0000 - 0000 007F	0xxxxxxx
0000 0080 - 0000 07FF	110xxxxx 10xxxxxx
0000 0800 - 0000 FFFF	1110xxxx 10xxxxxx 10xxxxxx
0001 0000 - 001F FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

UTF-8 geriye uyumluluk ve batı dilleri için en verimli şekilde bellek kullanımı sağlar.

UTF-16 ise her karakter için iki byte kullanır. Karakterlerinin büyük kısmı ASCII sınırları dışında kalan dillerde UTF-8 e göre büyük bellek tasarrufu ve hız sağlar.

Metin tipi tanımlaması (DTD) bir sınıf metin için gramer kurallarını içerir. XML metinleri hangi kurallara uyduklarını **<!DOCTYPE . . . >** etiketi ile belirtirler. Kurallar metnin içinde ve/veya harici bir kaynakta bulunabilir

```
<!DOCTYPE sunum SYSTEM "sunum.dtd">
```

```
<!DOCTYPE sunum PUBLIC "-//OASIS//DTD XML Catalogs V1.0//EN"  
    "http://www.oasis-  
open.org/committees/entity/release/1.0/catalog.dtd">
```

```
<!DOCTYPE sunum [  
    <!ELEMENT sunum (lisans, slayt+, taslak?)>  
    <!ELEMENT lisans (#PCDATA)>  
    <!ELEMENT slayt (başlık, logo?, paragraf*)>  
    <!ELEMENT paragraf (#PCDATA | k | i | url | resim)*>  
    <!ELEMENT taslak EMPTY>
```

```
    <!ATTLIST paragraf      yanaşık (sol|sağ|orta) "sol">  
    <!ATTLIST resim         url CDATA #REQUIRED      alt CDATA  
#IMPLIED>  
    <!ATTLIST lisans        gfdl CDATA #FIXED "evet">  
    <!ATTLIST slayt         numara ID #REQUIRED  
>
```

Diğer DTD etiketleri:

```
<!ENTITY % draft 'INCLUDE'>
```

```
<!ENTITY % final 'IGNORE'>
```

```
<!ENTITY % font SYSTEM "font.entity">
```

```
<ELEMENT paragraph (#PCDATA | %font; )*>
```

```
<![ %draft;[
```

```
    <!ELEMENT book (comment*, title, body) >
```

```
]]>
```

```
<![ %final;[
```

```
    <!ELEMENT book (title, body) >
```

```
]]>
```

Etiketler içeren metin parçaları CDATA bölümleriyle kolayca metne eklenebilir:

```
<![CDATA[ <koşul> a < b & c = 5 </koşul> ]]>
```

Metinde kullanılmak üzere varlıklar tanımlanabilir:

```
<!ENTITY sürüm "1.0">
```

Bu metin XML standardının &sürüm; sürümüne dayanılarak yazılmıştır.

İsim alanları (Namespace) farklı uygulamalara ait XML anlamlandırmalarının tek bir XML metni içinde kullanılabilmesini sağlar. Etiketlere ve etiket özelliklerinin başına **:** karakteri ile ayrılarak eklenir:

<bk:book>

Etiketlere eklenen **xmlns** özelliği ile o etiketin ve içindeki etiketlerin kullanabileceği isim alanları belirlenir:

```
<?xml version="1.0"?>
<bk:book xmlns:bk='urn:loc.gov:books' xmlns:isbn='urn:ISBN:0-395-36341-6'>
  <bk:title>Cheaper by the Dozen</bk:title>
  <isbn:number>1568491379</isbn:number>
</bk:book>
```

Eğer bir alan ismi verilmezse **xmlns** ön tanımlı alanı belirler:

```
<?xml version="1.0"?>
<book xmlns='urn:loc.gov:books' xmlns:isbn='urn:ISBN:0-395-36341-6'>
  <title>Cheaper by the Dozen</title>
  <isbn:number>1568491379</isbn:number>
</book>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE abiword PUBLIC "-//ABISOURCE//DTD AWML 1.0 Strict//EN"
    "http://www.abisource.com/awml.dtd">
<abiword
    xmlns:fo="http://www.w3.org/1999/XSL/Format"
    xmlns:svg="http://www.w3.org/2000/svg"
    xmlns:awml="http://www.abisource.com/awml.dtd"
    xmlns="http://www.abisource.com/awml.dtd" >
<!-- This file is an AbiWord document. -->
<styles>
<s type="P" name="Normal" followedby="Current Settings" props="text-indent:0in;
margin-top:0pt; margin-left:0pt; font-stretch:normal; line-height:1.0; text-align:left;
bgcolor:transparent; lang:en-US; margin-bottom:0pt; text-decoration:none; font-
weight:normal; font-variant:normal; color:000000; text-position:normal; font-size:12pt;
margin-right:0pt; font-style:normal; font-family:Times New Roman"/>
</styles>
<pagesize pagetype="Letter" orientation="portrait" width="8.500000"
    height="11.000000" units="in" page-scale="1.000000"/>
<section props="page-margin-footer:0.5in; page-margin-header:0.5in">
<p style="Normal">Merhaba <c props="font-style:italic">Linux</c> !!!</p>
</section>
</abiword>
```

XML Schema XML metinlerinin uyması gereken biçim kurallarını tanımlayan bir XML dilidir.

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="istek" type="istekTipi"/>
  <xsd:complexType name="istekTipi"><xsd:sequence>
    <xsd:element name="alıcı" type="xsd:string"/>
    <xsd:element name="ürün">
      <xsd:simpleType><xsd:restriction
base="xsd:positiveInteger">
        <xsd:maxExclusive value="100"/>
      </xsd:restriction></xsd:simpleType>
    </xsd:element>
  </xsd:sequence></xsd:complexType>
</xsd:schema>
```

XPath

XML metinleri içinde belirli kısımları adresleyebilmeyi sağlar.

```
/kitap/sayfa/*
//kitap/descendant::sayfa
//*[starts-with(name(),'s')]
//*[count(resim)=2]
```

XPointer

XPath'ın uzak kaynaklara bağlantıları göstermek için geliştirilmiş bir türevidir.

http: // www.foo.org/bar.xml#xpointer(article/section[position()<=5])

XLink

XML metinleri arasında bağlantıları işaretlemek için kullanılır.

<link xlink:type="simple" xlink:href="..." xlink:show="embed" />

<link xlink:type="extended">

<a xlink:type="locator" xlink:href="cars.xml#BMW"

xlink:label="car"/>

<b xlink:type="locator" xlink:href="tires.xml#Pirelli"

xlink:label="tire"/>

<arc xlink:type="arc" xlink:from="car" xlink:to="tire"/>

</link>

XQuery

XML metinleri üzerinde sorgulamalar yapmaya yarayan bir sorgu dilidir.

<neyemeli> { distinct-values(

for \$r in document("tarifler.xml")//tarif

for \$i in \$r//malzeme/@isim

for \$j in document("buzdolap.xml")//yiyecek[text()=\$i]

return \$r/isim

) } </neyemeli>

XSLT

Bir XML metnini başka bir XML metnine dönüştürecek transformasyonları yazmaya yarayan bir XML dilidir.

```
<?xml version='1.0'?>
<xsl:stylesheet version='1.0'
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:template match="/">
    <html><body><h1>CD Listem</h1>
    <xsl:apply-templates/>
    </body></html>
  </xsl:template>

  <xsl:template match="cd"><p>
    Albüm: <b> <xsl:value-of select="title"/> </b>
    Sanatçı: <i> <xsl:value-of select="artist"/> </i>
  </p></xsl:template>

</xsl:stylesheet>
```

XSL-FO

XML etiketlerinin görsel olarak nasıl gösterileceğini belirlemeye yarayan bir XML dilidir.

Karmaşık ya da ilerde karmaşılaşabilecek veriler, yerelleştirme desteği gerektiren veriler etiketler ile gösterilmelidir. Ayrıca DTD ya da Schema ile kontrol edilmesi istenen veriler için etiket tercih edilmelidir.

```
<mektup önem='acil' gönderen='Ali Can' başlık='Meeting Call'>
```

```
...
```

```
</mektup>
```

```
<mektup>
```

```
  <acil/>
```

```
  <başlık>Meeting Call</başlık>
```

```
  <başlık xml:lang='tr'>Toplantı Çağrısı</başlık>
```

```
  <gönderen>
```

```
    <isim>Ali Can</isim>   <görev>müdür yardımcısı</görev>
```

```
  </gönderen>
```

```
...
```

```
</mektup>
```

Özellikler bu durumların dışında XML metnini yalın ve okunaklı kıldıkları durumlarda kullanılmalıdır.

```
<a><href>senlik.linux.org.tr</href><b>Linux Şenlik Sayfası</b></a>
```

```
<a href='senlik.linux.org.tr'>Linux Şenlik Sayfası</a>
```

Etiket, özellik ve özellik değeri isimlerini seçerken kısaltmalardan kaçınmalı, anlamlı isimler tercih edilmelidir.

```
<a3f01 1='a'/>
```

```
<m g='Ali Can'><b>Merhaba</b> . . . </m>
```

Varolan standartlar olabildiğinde kullanılmalı, herşeyi yeniden icat etmekten kaçınmalıdır.

XML-FO SVG DOCTYPE MATHML

Özgür olmayan yazılımlarda çok rastlanan kötü kullanım örnekleri:

```
<?xml version='1.0'?>
```

```
<mydoc>2af34evr4htr4ugdf9uj4tj . . . </mydoc>
```

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

```
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
```

```
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
```

```
<plist version="1.0"><dict>
```

```
<key>AnimateSnapToGrid</key>
```

```
<string>>true</string>
```

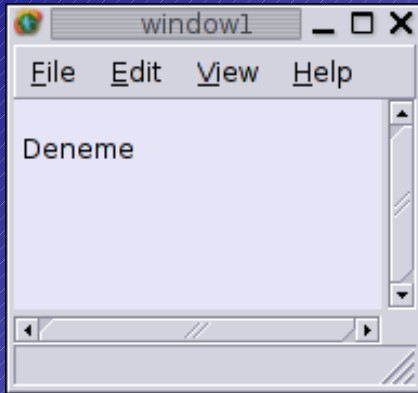
```
<key>EmptyTrashProgressWindowLocation</key>
```

```
<string>79, 44</string>
```

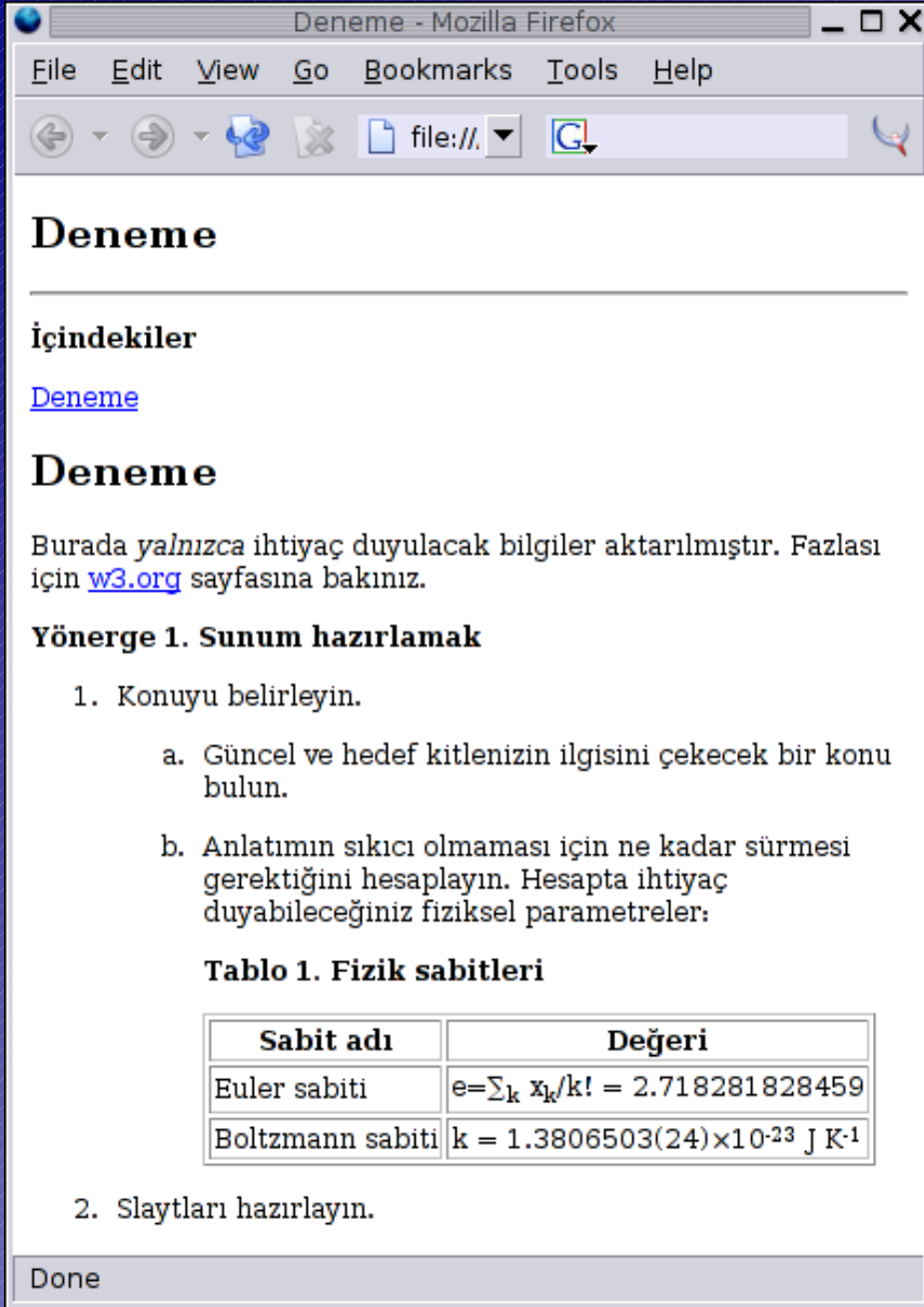
```
</dict></plist>
```

Verinin bir ağaç yapısında olduğu durumlarda XML veri ile gösterim arasında birebir uyum sağlar.

Glade



```
<glade-interface>
<widget class="GtkWindow" id="window1">
  <property name="title"
translatable="yes">window1</property>
  <child>
    <widget class="GtkVBox" id="vbox1">
      <child>
        <widget class="GtkMenuBar" id="menubar1">
          . . .
        </child>
        <child>
          <widget class="GtkScrolledWindow"
id="scrolledwindow1">
            <child>
              <widget class="GtkTextView" id="textview1">
                <property name="text">Deneme</property>
              </child>
            </widget>
          </child>
          <child>
            <widget class="GtkStatusbar" id="statusbar1">
              . . .
            </widget>
          </child>
        </child>
      </widget>
    </child>
  </widget>
```



```
<?xml version='1.0'?>
<!DOCTYPE article PUBLIC "-//OASIS//DTD DocBook XML V4.2//EN"
"http://www.oasis-open.org/docbook/xml/4.2/docbookx.dtd">
<article lang='tr'><title>Deneme</title><sect1><title>Deneme</title>
<para>Burada <emphasis>yalnızca</emphasis> ihtiyaç duyulacak bilgiler
aktarılmıştır.
Fazlası için <ulink url="http://w3.org">w3.org</ulink> sayfasına bakınız.
</para>
<procedure><title>Sunum hazırlamak</title>
  <step performance="required">  <para>Konuyu belirleyin.</para>
    <substeps>
      <step performance="required">
        <para>Güncel ve hedef kitlenizin ilgisini çekecek bir konu
bulun.</para>
      </step>
      <step performance="optional">
        <para>Anlatımın sıkıcı olmaması için ne kadar sürmesi gerektiğini
hesaplayın. Hesapta ihtiyaç duyabileceğiniz fiziksel
parametreler:</para>
        <table frame="all"><title>Fizik sabitleri</title> <tgroup cols="2">
          <thead><row>
            <entry>Sabit adı</entry><entry>Değeri</entry>
          </row></thead>
          <tbody>
            <row>
              <entry>Euler sabiti</entry>
              <entry>e=<sum;<subscript>k</subscript></subscript>
                = 2.718281828459</entry>
            </row>
            <row>
              <entry>Boltzmann sabiti</entry>
              <entry>k = 1.3806503(24)&times;10<superscript>-
23</superscript>
                J K<superscript>-1</superscript></entry>
            </row>
          </tbody>
        </tgroup></table>
      </step>
    </substeps>
  </step>
  <step performance="required">  <para>Slaytları hazırlayın.</para>
</step>
</procedure>
</sect1></article>
```


Verinin çeşitli uygulamalar ve farklı mimariler üzerinden erişilebilmesi XML ile çok kolaydır.

OpenOffice.org

OpenOffice.org dosyaları aşağıdaki dosya ve dizinleri barındıran birer ZIP dosyalarıdır.

META-INF/manifest.xml Diğer dosyaların adları ve tiplerine ilişkin bilgileri içerir.

```
<manifest:file-entry manifest:media-type="text/xml" manifest:full-path="content.xml"/>
```

mimetype application/vnd.sun.xml.impress

meta.xml Belgenin yazarı, hazırlanma tarihi gibi bilgiler.

styles.xml İçerikte kullanılan biçim stilleri.

```
<style:style style:name="standard" style:family="graphics">
  <style:properties draw:stroke="solid" svg:stroke-color="#000000" fo:margin-left="0cm"/>
  <text:list-style>
    <style:style style:name="gr1" style:family="graphics" style:parent-style-name="standard">
      <style:properties draw:stroke="none" draw:fill-gradient-name="Linear blue/white"/>
    </style:style>
  </text:list-style>
</style:style>
```

content.xml

Belgenin içeriğini taşır.

```
<draw:page draw:name="Başlık" draw:style-name="dp1" draw:id="1"
  draw:master-page-name="Arkaplan">
  <draw:text-box draw:style-name="gr1" draw:text-style-name="P2"
draw:layer="layout"
  svg:width="26.847cm" svg:height="3.427cm" svg:x="0.698cm" svg:y="2.141cm">
    <text:p text:style-name="P1">
      <text:span text:style-name="T1">Bir Veri Saklama Biçimi Olarak
XML</text:span>
    </text:p>
  </draw:text-box>
  . . .
</draw:page>
```

OpenOffice.org dosya biçiminin getirileri:

- Linux (x86/ppc), Solaris (x86/sparc), Windows, MacOS X mimarilerinde çalışıyor.
- Metin, sunum ve hesap tablosu verilerini tek tek veya bir arada tutabiliyor.
- Dış verileri (resim, multimedya, vs) taşıyabiliyor.
- Erişim performansı yüksek, dosyalar az yer kaplıyor.
- Varolan standartları kullanıyor, genişlemelere açık.
- Diğer uygulamalar tarafından kolayca erişilebiliyor ve yaratılabiliyor.
- Basit shell komutları ya da perl programları ile bile kolayca dosya içeriklerine ulaşmak mümkün.

III. Linux ve Özgür Yazılım Şenliği

Jabber

- Jeremie Miller'in 1999 yılında başlattığı bir özgür yazılım projesi (<http://jabber.org>).
- Uygulamaların email benzeri bir sunucular/istemciler mimarisinde XML yoluyla veri aktarımı yapabilmelerini ve diğer uygulamalar tarafından bulunabilirliklerini sağlar.
- IETF'nin 2002 yılında kurduğu XMPP (eXtensible Messaging and Presence Protocol) çalışma grubu tarafından geliştirilip bir internet protokolü durumuna getirildi.
- Jabber bu protokol üzerinde çalışan bir anında iletişim (instant messaging) sistemi, her platformda istemcileri, çeşitli diller için protokol kitaplıkları ve çeşitli sunucuları var (<http://jabberstudio.org>).

İstemci

```
<?xml version='1.0'?>
<stream:stream to='example.com' xmlns='jabber:client'
  xmlns:stream='http://etherx.jabber.org/streams'
  version='1.0'>

<message to='zeynep@jabber.org'>
  <body>Merhaba, nasılsın?</body>
</message>

</stream:stream>
```

Sunucu

```
<?xml version='1.0'?>
<stream:stream from='example.com'
  xmlns='jabber:client'

  xmlns:stream='http://etherx.jabber.org/streams'
  version='1.0' id='someid'>

<message from='zeynep@jabber.org'>
  <body>iyiyim, sen nasılsın?</body>
</message>

</stream:stream>
```



```
<message from='yalçın@js.com.tr' type='chat'>
  <subject>XML</subject>
  <body>Hi, have you prepared your presentation?</body>
  <body xml:lang='tr'>Selam, sunumunu hazırladın mı?</body>
  <smiley xmlns='http://foo.org/jabber/smiley.dtd'>telaşlı</smiley>
</message>
```

```
<presence to='ahmet@firma.com' type='subscribe'/>
```

```
<presence from='madcat@jabber.org'>
  <show>dnd</show>
  <status>çalışıyorum</status>
</presence>
```

```
<iq type='get' id='1234'><query xmlns='jabber:iq:roster'></iq>
```

```
<iq type='result' id='1234'><query xmlns='jabber:iq:roster'>
  <item jid='ahmet@firma.com' name='Ahmet Can' subscription='both'/>
  <item jid='madcat@jabber.org' name='Gürer'
subscription='to'><group>arkadaşlar</group></item>
</query></iq>
```

```
<iq type='set'><query xmlns='jabber:iq:roster'>
  <item jid='ahmet@firma.com' subscription='remove'/>
</query></iq>
```


Veriye ait meta verinin her veri parçasında farklı olduğu ve/veya öngörülmecek biçimde değiştiği

durumlarda XML basit bir gösterim şekli, geriye ve ileriye uyumluluk sağlar.

<ayarlar>

<yedekleme> <otomatik süre='5' /> <uzantı>.bak</uzantı> </yedekleme>

<arabirim>

<toolbar/><ipuçları/>

<renkpaleti>yüksek-kontrast</renkpaleti>

</arabirim>

</ayarlar>

Özellikle uygulamanın diğer veriler için de XML kullandığı durumlarda ayar dosyalarının işlenmesi bedavaya gelmektedir. Bununla birlikte ayarların çok basit olması, fazla değişmeyecek olması veya genel ayarlar olmaları durumunda INI dosyası biçimi kullanmak basitlik sağlar ve ayarların temel metin komutlarıyla işlenmesini çok daha kolaylaştırabilir.

basit INI ayar dosyası

otoyedekleme=5

uzantı=.bak

araççubuğu=açık

ipuçları=açık

renkpaleti=yüksek-konstrast

Verinin meta veri içermediği durumlarda XML herhangi bir getiri sağlamadan kaynak israfına yol açar.

- Grafik, ses, multimedya verileri meta verilerini de içeren kendi başlarına veri biçimlerine sahiptir.
- Sayı dizisi ve matrislerinden oluşan verilerin meta verileri verinin bütününe kapsar ve çok azdır.
- Liste ve tablo gibi veri yapılarının meta verisi dizideki tüm elemanlar için sabittir.

SELECT yazar FROM kitaplar

Kitaplar.xml

Kitaplar İlişkisel Veritabanı

<u>Yazar</u>	<u>Kitap</u>
Nicolai Hartmann	Ontolojinin Işığında Bilgi
Gürsel Korat	Zaman Yeli
0. byte	20. byte 100. byte

ptr += 100 (tek cpu komutu ile sonraki kayıt)

ptr = yazar_ptrs[++index]
(bir arttırma, bir bellek erişimi ile sonraki kayıt)

```
<?xml version='1.0'?>
<kitaplar>
  <kitap>
    <yazar>Nicolai Hartmann</yazar>
    <ad>Ontolojinin Işığında
Bilgi</ad>
  </kitap>
  <kitap><yazar>Gürsel Korat</yazar>
  <ad>Zaman Yeli</ad></kitap>
</kitaplar>
```

Kayıtları dolaşmak kaç cpu işlemi gerektirir?

Seri Erişim (SAX - Serial Access)

- XML metni işlenirken karşılaşılan her element için programın önceden belirttiği yordamlar çağrılır.
- Veriler üzerinde geriye veya ileriye dönük referanslar tutmadığı için bellek kullanımı en az düzeydedir.

<?xml version="1.0"?>

<kitap ad="Kedi Beşiği" yazar="Kurt Vonnegut">

etiket (kitap, “ad”, “Kedi Beşiği”
, “yazar”, “Kurt Vonnegut”)

Daha genç olduğum sıralarda, adını
sıralarda, adını”)

metin (“Daha genç olduğum

<k>Dünyanın son günü</k> koymaya
karar verdiğim bir kitap için belgeler
toplamaya başladım.

etiket (k)

metin (“Dünyanın son günü”)

etiket_sonu (k)

metin (“koymaya karar verdiğim bir kitap”
“için belgeler toplamaya başladım”)

</kitap>

etiket_sonu (kitap)

Metin Nesne Modeli (DOM - Document Object Model)

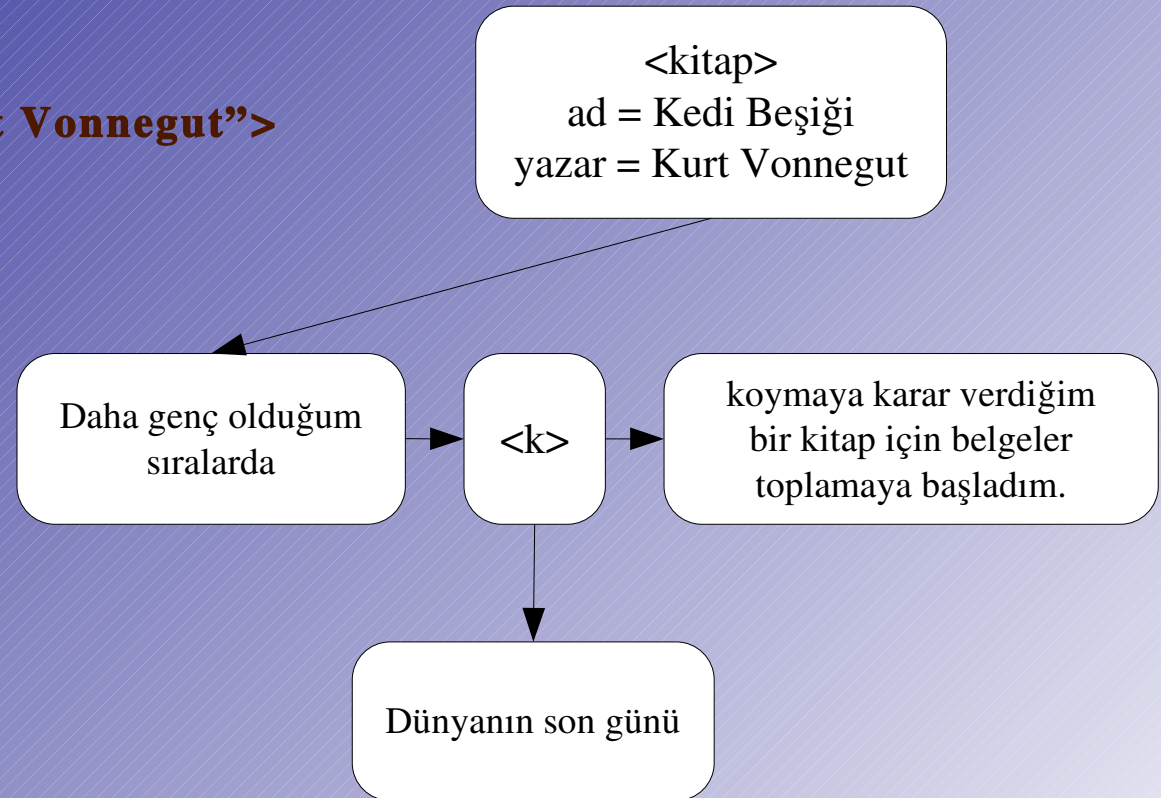
- XML metnini bir ağaç yapısına dönüştürüp programa aktarır.
- Dosyadaki veriler ve ağaç yapısının meta verileri bellekte yer kaplar.

<?xml version="1.0"?>

<kitap ad="Kedi Beşiği" yazar="Kurt Vonnegut">

**Daha genç olduğum sıralarda, adını
<k>Dünyanın son günü</k> koymaya
karar verdiğim bir kitap için belgeler
toplamaya başladım.**

</kitap>



libxml

<http://xmlsoft.org>

- Standart GNOME XML kitaplığı
- XML, HTML, URI, SAX, DOM, XML Reader, DTD, XML Catalog, XPath, XPointer, XInclude, Iconv
- libxslt, libxmlsec gibi ekstra kitaplıklar, C++, python, perl arabirimleri

```
int main(int ac, char **av)
{
    xmlDocPtr doc;
    xmlXPathObjectPtr res;
    xmlXPathContextPtr ctxt;

    doc = xmlReadFile (av[1], NULL, 0);
    ctxt = xmlXPathNewContext (doc);
    ctxt->node = xmlDocGetRootElement (doc);

    res = xmlXPathEvalExpression(BAD_CAST av[2], ctxt);
    ...
    xmlXPathFreeObject(res);
    xmlXPathFreeContext(ctxt);

    xmlFreeDoc (doc);
    xmlCleanupParser();
    return(0);
}
```

```
int main (int ac, char **av)
{
    xmlTextReaderPtr reader;

    reader = xmlReaderForFile (av[1], NULL, 0);
    while (xmlTextReaderRead (reader) == 1) {
        printf ("Etiket %s Derinlik %d Tip %d\n",
                xmlTextReaderConstName (reader),
                xmlTextReaderDepth (reader),
                xmlTextReaderNodeType (reader));
    }
    xmlFreeTextReader (reader);
    xmlCleanupParser();
    return 0;
}
```

iksemel

<http://iksemel.jabberstudio.org>

- SAX, DOM, XML Stream, Jabber desteği
- Modüler, tutarlı, kolay öğrenilen
- ANSI C dilinde, Posix, Windows, Symbian uyumlu sistemlerde çalışıyor
- XML 1.0, DTD desteği yok, yalnızca UTF-8

```
int bas;

int etiket (void *u, char *ad, char **ozel, int tip) {
    if (strcmp (ad, "bas") == 0) {
        if (tip == IKS_TYPE_OPEN)
            bas = 1;
        else
            bas = 0;
    }
}

int metin (void *u, char *veri, int boy) {
    if (bas) puts (veri);
}

int main (int ac, char *av[]) {
    iksparser *prs;
    int err;
    prs = iks_sax_new (NULL, etiket, metin);
    iks_parse (prs, av[1], 0, 1);
    iks_parser_delete (prs);
    return err;
}
```

```
int main (int ac, char *av[]) {
    iks *x;
    int err;
    x = iks_tree (av[1], 0, &err);
    if (err == IKS_OK) {
        printf ("TEST etiketinin içeriği:\n%s\n", iks_string (NULL, iks_find (x, "TEST")));
    }
    return err;
}
```

```
iksparser *prs;
int node_hook (void *u, int tip, iks *node) {
    if (tip == IKS_NODE_START) iks_auth_sasl (prs, 0, "madcat", "şifre");
    else . . .
}

int main (int ac, char *av[]) {
    prs = iks_stream_new (IKS_NS_CLIENT, NULL, node_hook);
    iks_connect_tcp (prs, "jabber.org", IKS_JABBER_PORT);
    while (iks_recv (prs, 1) == IKS_OK);
}
```

expat

- <http://expat.sf.net>
- 1998'de James Clark tarafından geliştirildi, basit SAX tabanlı, DTD yok

xerces

- <http://xml.apache.org>
- C++, Java dilleri için, SAX, DOM, DTD, XML Schema desteği

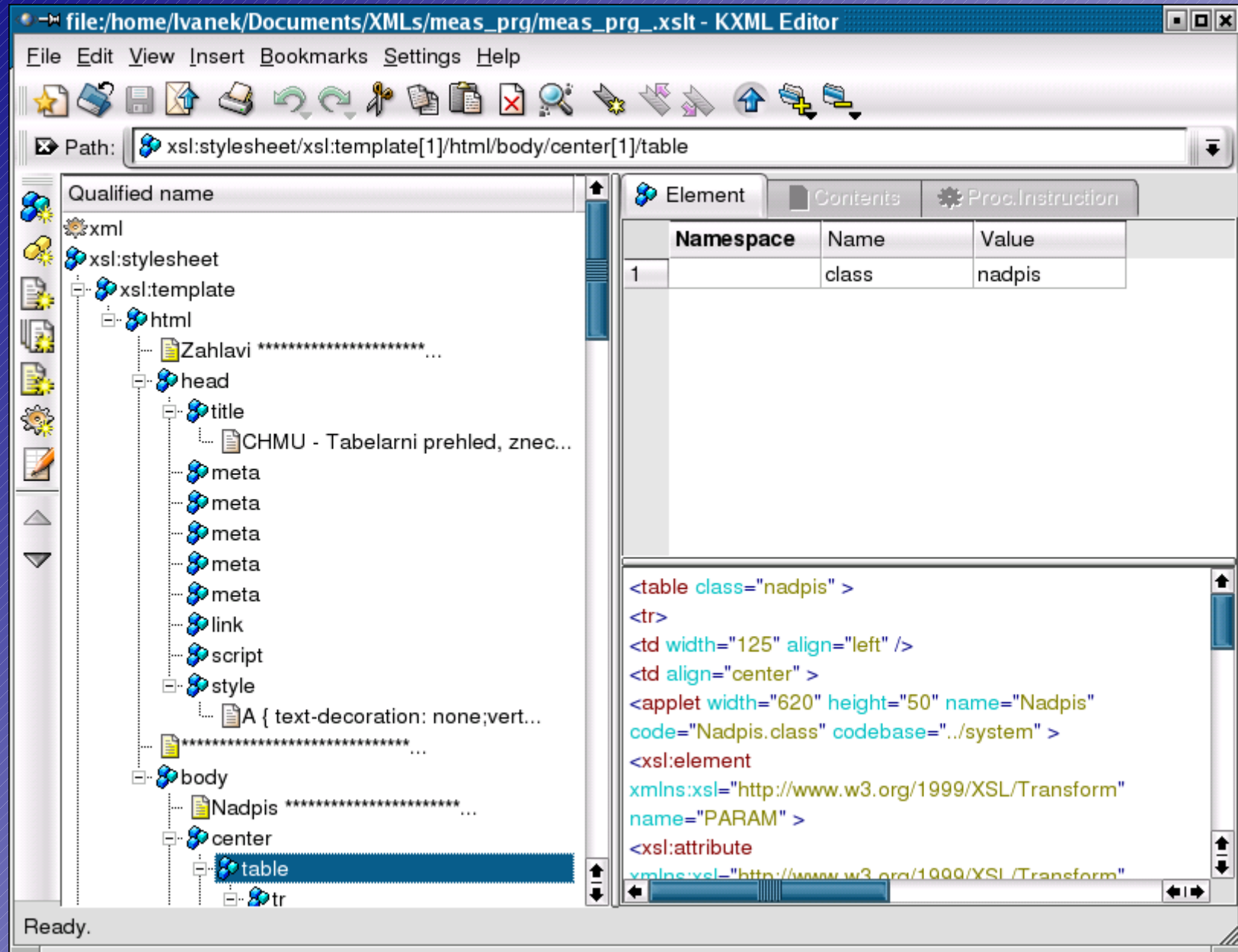
xalan xerces üzerinde XPath ve XSLT desteği sağlayan C++, Java kitaplığı

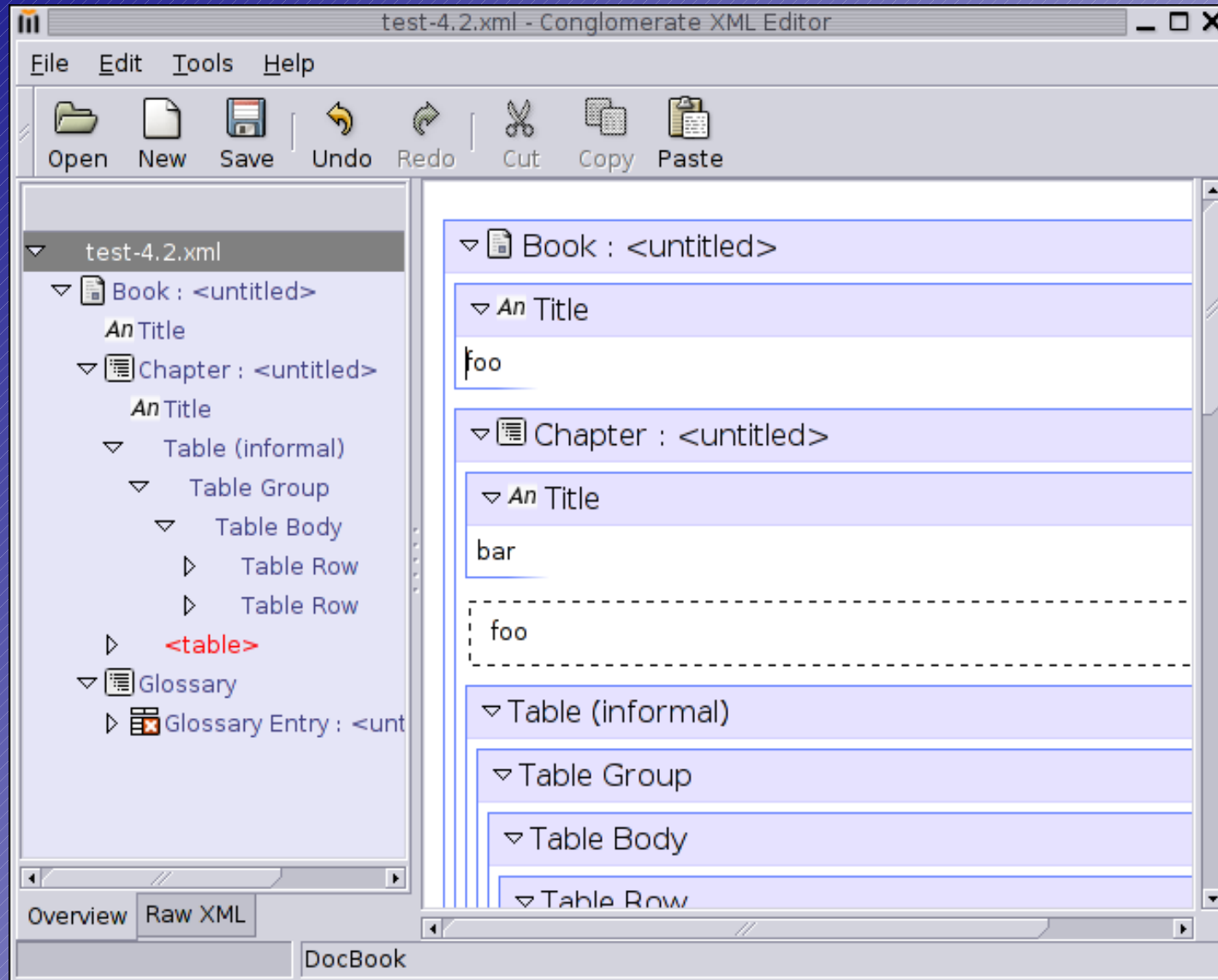
xindice xerces/xalan üzerinde XML veri tabanı, sorgulamalar için XPath, değişiklikler için XUpdate kullanan Java kitaplığı, ufak ve orta büyüklükte XML metinleri için

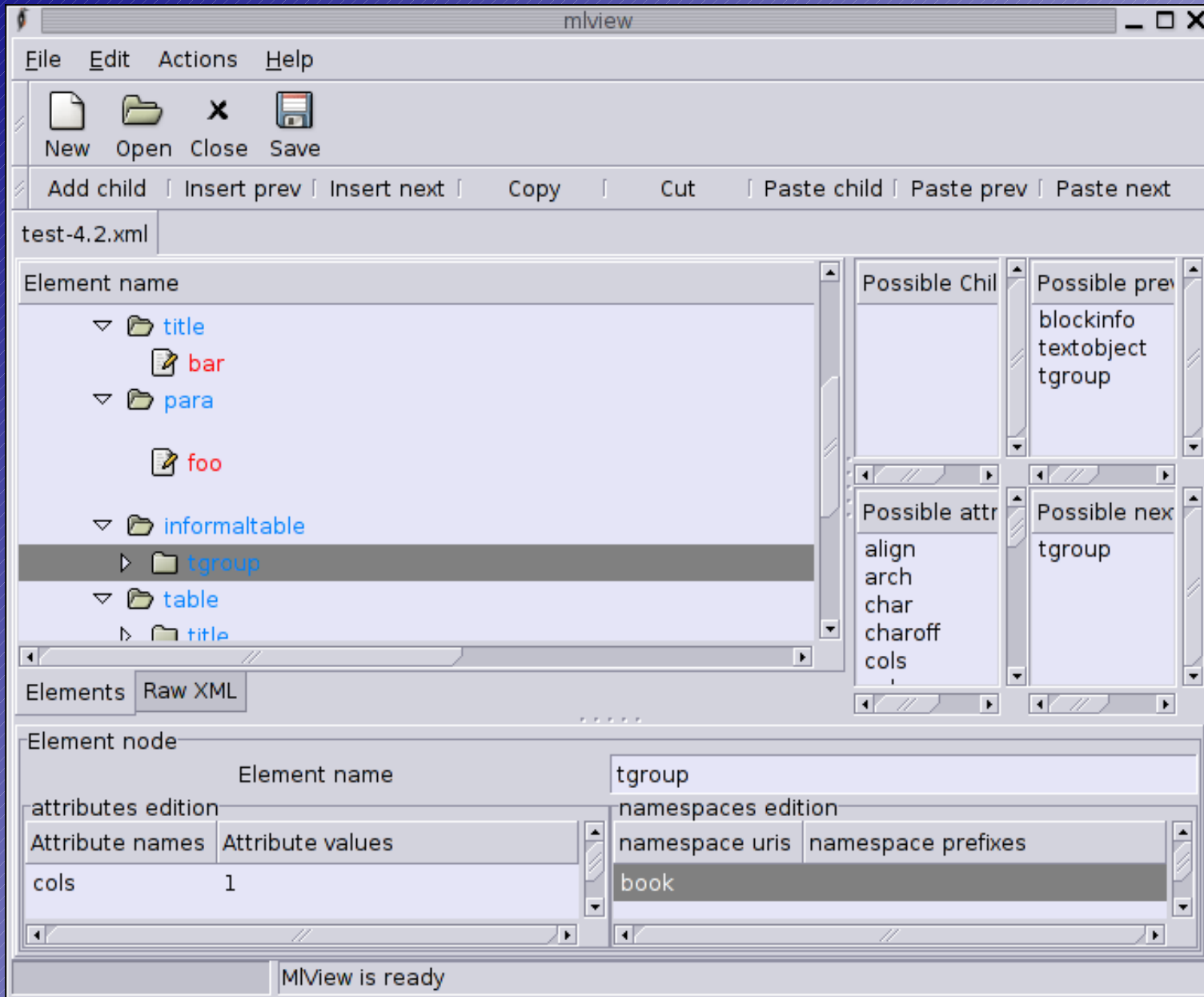
```
<xupdate:insert-after select='/adresler/adres[1]'>
  <xupdate:element name='adres'>
    <xupdate:attribute name='id'>2</>    <isim>Ali Can</isim>    <il>İstanbul</il>
<postakodu>81010</postakodu>
  </xupdate:element>
</xupdate:insert-after>
```

berkeley db xml

- <http://www.sleepycat.com/products/xml.shtml>
- BerkeleyDB içinde tutulan XML metinleri üzerinde XPath sorgulamaları yapmayı sağlar
- C++, Java dillerini destekler, xerces tabanlıdır
- Hızlı sorgulama için metinler üzerinde indeks oluşturur, metin değişiklikleri yeniden indeksleme gerektirir







Sorular