EKI kohanimede andmebaasi allikfailide konverteerimine XMLiks translaatori abil

Kristian Kankainen, MTÜ Keeleleek
Peeter Päll, Eesti Keele Instituut

2016

1 Sissejuhatus

Kohanimeandmebaas (KNAB) on Eesti Keele Instituudis arendatav süstemaatiline kohanimeandmete kogu, mis hõlmab nii Eesti kui ka välismaa nimesid. Selle eesmärk on kaasa aidata nimede uurimisele ja nende normimisele, pakkudes teavet nimede ajaloo ja tänapäeva kohta. Andmebaas on kavandatud filoloogiakeskselt, võimaldamaks selle põhjal mitmesuguste nimeloendite ja sõnastike koostamist.

Arvutiandmebaasi kavandamist alustati 1988. aastal omaaegsete suurarvutite baasil. Selle ajaloo kohta loe rohkem leheküljel http://www.eki.ee/knab/kb_yld.htm.

Kohanimeandmebaasi interneti-versioonis on kättesaadav vaid integreeritud osa. Andmete ümberviimine XMLi kujule lihtsustab andmete kättesaadavust ja võimaldab andmete paindliku esitamise.

Mis formaadis olid need enne konverteerimist? Kuidas olid need koostatud?

Töö põhikomponent ehk translaator loodi Tartu Ülikooli Arvutiteaduse instituudi aine „Automaadid, keeled ja translaatorid" (MTAT.05.085, 6 EAP) raames 2016. aasta kevadsemestril. Töö lähtekood on litsentseeritud GNU GPLv3 alusel ja programmeerimistöö autor on Kristian Kankainen.

2 Allikfailide transleerimine XMLiks

Allikfailid süstematiseeritud struktuur võimaldab nende automaatset töötlemist. Sõelumise juures kehtestatakse loogiline struktuur andmete üle. Selline loogiline struktuur on kirjeldatav formaalse grammatikaga. Transleerimise juures käivitatakse loogilise struktuuriosade puhul mingi tegevus. Antud töö raames moodustuvad sellised tegevused suuresti struktuuriosade sisu ümberkirjutamine vastavaks XML väljundile.

Töös valitud formaalse grammatika võimaluste piires on tarbilik jagada konverteerimine mitmeks etapis. Siin peatükis kirjeldatud etappi võib mõista kui allikfailide makrostruktuuri konverteerimine. Järgmises peatükis käsitletakse konverteerimise mikrostruktuuri, ehk reasisest konverteerimist

Translaatorite ehitamine on võimalik automatiseerida, mille puhul lihtsustatakse programmeerimist suuresti kaheks komponendiks: sisendi loogilise struktuuri esitamiseks formaalse grammatika kujul ja transleeri tegevuste deklareerimiseks. Töös on kasutatud ANTLR 4 translaatori ehitamiseks. Kirjutise autor tahab püsida neutraalsena tarkvara valikus. Valik tehti TÜ aine sisust ajendatuna.

```
 !‘Aandu k <”sk,”s1,”s0> [EE1945n; EE1970n; EE1978n]
/p20:#H1021:#X96000808/RP:Khl/Hag//G:590916N−244306E
$k(:)R468
$pRP:Khl [1970; 1978]; HR:Khl:Kohila+kn [1945]; HR:Khl [1922]
$jLohu+ms [1725; 1913]
( Oando k [HR1926k]; Aando k [EE1922n]; Oandu k <!kat> [EE1930k_SK]
( Ando, (Df)+ <de> [Ri1913]; Aando, (Df)+ <de> [EE_P1871k:4t]
( Aanda, (Df)+ <de> [Me1798]; Aanda, Dorff+ <de> [HR1725n]
(    , (.)+ <ru,*w> [EE1900k_1v:4t]
3 A0# −ndu
5 Päevati k [1977]; Rootsi k [1977]; Kadaka k [1977]
9 EKNR
```

Joonis 1: KNABi allikfaili struktuurinäidis

Töövoo visualiseerimine on esialgu näidatud järgmiselt. Hiljem tuleb siia ilusa pildi koostada. Allikfail –[transleerija]–> XML –[Xquery]–> XML^2 ... XML^n –[Xquery]–> XML^{n+1}

Allikfaili tekstikuju näide on toodud joonises 1.

Allikfailide abstraktse (makrostruktuuri) kirjeldus on veebiaadressil.

Esmase transleeritud XMLi puustruktuuri kirjeldus on toodud joonises 2. Struktuur peegeldab allikfailide struktuuri.

2.1 Sõeluja kitsaskohad

ANTLRi metodoloogiliste valikute tõttu ei suudeta formaalses grammatikas väljendada üht allikfailides esinenud tõika. Nimelt esinesid järgridade klassifitseerimiseks kasutatud erimärgid (tokenid) ka muudes kohtades allikfaili struktuuris. Kuna need esinesid ainult allikfailide vaba teksti osades, sai need puudused kõrvaldatud allikfailide tekstide ümberkirjutamisega (relevantsed reaalgused asendati kahe sidekriipsuga algavatega).

Need kitsaskohad kehtivad ainult allikfailide puhul ja ei puuduta andmete täiendamist XMLi toimetamissüsteemi juures.

3 XML struktuuri edasine transleerimine Xquery abil

Konverteerimist on lihtsam teha etapiliselt, kus iga etapiga teisendatakse mingi konkreetne struktuur ümber teiseks ja XML-põhiseks. Kui eelmises peatükis kirjeldatud konverteerimine on mõistetav kui allikfaili makrostruktuuri teisendamisena, siis edaspidi tuleb juttu mikrostruktuuri teisendamisest, ehk allikfaili ridadesisesest teisendamisest.

XML tehnoloogia sobitub ideaalselt allikfailide ridadesiseseks teisendamiseks. XMLi kasutamine ja ümbertöötlemine tegi läbi hüppelise arengu, kui W3C töörühmad defineerisid uue standardi selle andmestruktuurile seoses XPath 2.0 keelega. Uut andmestruktuuri oli vaja selleks, et ühtlustada W3C eri töörühmade tehnoloogiaid. Peamised XMLi töötlevad tehnoloogiad on XSLT 2.0 ja Xquery, mis mõlemad järgivad XPath 2.0 standardit.

Olgu siinkohal öeldud vaid niipalju, et XMLi töötlevad tehnoloogiad on omavahel väga komplementaarsed ja kellele üks ei meeldi, tasuks proovida teist. XSLT on deklaratiivne keel, mille maailmapilt on ülevalt-alla, see töötleb XMLi suunas suuremast üksusest väiksemale. Xquery maailmapilt on altpoolt-ülesse ja töötleb XMLi kokkupannes väiksemaid üksusi suurematekks.

```
<kirje>
  <põhirida> !‘Aandu k
    <infomärgendid>”sk,”s1,”s0</infomärgendid>
    <allikaviitemärgendid>EE1945n; EE1970n; EE1978n</allikaviitemärgendid>
  </põhirida>
  <lisarida>p20:#H1021:#X96000808/RP:Khl/Hag//G:590916N−244306E</lisarida>
  <laiendirida>k(:)R468</laiendirida>
  <laiendirida>pRP:Khl
    <allikaviitemärgendid>1970; 1978</allikaviitemärgendid>; HR:Khl:Kohila+kn
    <allikaviitemärgendid>1945</allikaviitemärgendid>; HR:Khl
    <allikaviitemärgendid>1922</allikaviitemärgendid>
  </laiendirida>
  <laiendirida>jLohu+ms
    <allikaviitemärgendid>1725; 1913</allikaviitemärgendid>
  </laiendirida>
  <rööpnimerida> Oando k
    <allikaviitemärgendid>HR1926k</allikaviitemärgendid>; Aando k
    <allikaviitemärgendid>EE1922n</allikaviitemärgendid>; Oandu k
    <infomärgendid>!kat</infomärgendid>
    <allikaviitemärgendid>EE1930k_SK</allikaviitemärgendid>
  </rööpnimerida>
  <rööpnimerida> Ando, (Df)+ <infomärgendid>de</infomärgendid>
    <allikaviitemärgendid>Ri1913</allikaviitemärgendid>; Aando, (Df)+
    <infomärgendid>de</infomärgendid>
    <allikaviitemärgendid>EE_P1871k:4t</allikaviitemärgendid>
  </rööpnimerida>
  <rööpnimerida> Aanda, (Df)+ <infomärgendid>de</infomärgendid>
    <allikaviitemärgendid>Me1798</allikaviitemärgendid>; Aanda, Dorff+
    <infomärgendid>de</infomärgendid>
    <allikaviitemärgendid>HR1725n</allikaviitemärgendid>
  </rööpnimerida>
  <rööpnimerida>    , (.)+ <infomärgendid>ru,*w</infomärgendid>
    <allikaviitemärgendid>EE1900k_1v:4t</allikaviitemärgendid>
  </rööpnimerida>
  <etümoloogiarida> A0# −ndu</etümoloogiarida>
  <alaobjektiderida> Päevati k <allikaviitemärgendid>1977</allikaviitemärgendid>; Ro
    <allikaviitemärgendid>1977</allikaviitemärgendid>; Kadaka k
    <allikaviitemärgendid>1977</allikaviitemärgendid>
  </alaobjektiderida>
  <koostajarida> EKNR</koostajarida>
</kirje>
```

Joonis 2: KNABi XMLi struktuurinäidis

Algorithm 1 Lisaridade konverteerimine Xquery abil

```
let $knab := db:open("eeufx")
for $lisarida in //lisarida
  let $osad := fn:tokenize($lisarida, "/")
  return
    replace node $lisarida
    with
      <lisarida>
        <liigikood>{$osad[1]}</liigikood>
        <praeguneHalduvuskuuluvus>{$osad[2]}</praeguneHalduvuskuuluvus>
        <ajaloolineHalduvuskuuluvus>{$osad[3]}</ajaloolineHalduvuskuuluvus>
        <lähikuuluvus>{$osad[4]}</lähikuuluvus>
        <kohaviide>{$osad[5]}</kohaviide>
      </lisarida>
```

Erinevad probleemid on tihti lihtsamini kirjeldatavad ühel või teisel moel – ja seega programmaatiliselt lihtsamini lahendatavad ühe või teise keele abil.

3.1 Lisaridade konverteerimine

Lisarea tekstis on kaldkriipsudega eraldatud viis infoüksust: liigikood, praegune halduskuuluvus, ajalooline halduskuuluvus, lähikuuluvus ja kohaviide. Algoritm 1 näidatud kood tükeldab lisarea kaldkriipsude kohtadelt ja lisab iga osise teksti vastava nimetusega XML elemendisse.

Info lisaridade konverteerimise kohta: konverditi 35269 lisarida. Ajakulu 2111.96 ms (millest Parsing: 0.75 ms - Compiling: 1.04 ms - Evaluating: 2110.11 ms - Printing: 0.07 ms).

3.2 Järgmised etapid

Töö on hetkel pooleldi ja ootab oma aega.