TEI: testu-kodeketarako gidalerroak

J.M. Arriola (*), X. Artola, K. Gojenola, A. Soroa

Lengoaia eta Sistema Informatikoak

Informatika Fakultatea, P.K. 649 20080 Donostia

E-mail: jibaregj@si.ehu.es

Laburpena

Testu-kodeketarako ekimena (Text Encoding Initiative, TEI) delakoak testu elektronikoak kodetzeko eta trukatzeko proposaturiko gidalerroak zertan diren azaldu nahi da artikulu honen bidez: historia pixka bat egin, oinarriak eta helburuak aipatu, eta, adibide xume batzuen bidez, testu bat, gidalerroon arabera kodeturik nola geratzen den erakutsi nahi izan da. Horrez gain, euskarazko testuen kodetze-lanetan dihardauen ororen jakinmina piztea ere balitzateke gure asmoetako bat, sinetsita baikaude, edozein dela ere testu mota, etorkizunari begira testuok era kontsistente eta ahalik eta zabalduenean kodeturik edukitzeak daukan garrantzi handiaz.

1. Sarrera gisa

Testu-kodeketarako ekimena (Text Encoding Initiative, TEI[1]) delakoak testu elektronikoak kodetzeko eta trukatzeko bere gidalerroak —TEI P3[2]— 1994ko maiatzean eman zituen argitara (1). Sei urtetan zehar mundu zabaleko hamaika ikertzaile eta ikertalderen lanaren fruitu diren mila eta hirurehun orrialdeko gidalerro horien helburua ondokoa bezain sinple eta, aldi berean, handinahikoa da: ezaugarri desberdineko testu mota zabal bat era kontsistente eta optimo batean kodetzeko bideak eskaintzea.

TEI P3 gidalerro horiek zertan diren azaltzea da artikulu labur honetan dugun asmo nagusia. Horrez gain, euskarazko testuen kodetze-lanetan dihardauen ororen jakinmina piztea ere balitzateke gure asmoetako bat, sinetsita baikaude, edozein dela ere testu

1 http://www.uic.edu/orgs/tei/index.html

2 *Guidelines for Electronic Text Encoding and Interchange*, TEI P3 direlakoak.

(*) Eusko Jaurlaritzako Bekaduna

mota, etorkizunari begira testuok era kontsistente eta ahalik eta zabalduenean kodeturik edukitzeak daukan garrantzi handiaz.

Beraz, artikuluaren edukia honako hau da: aurrena, TEI aurkeztu, historia pixka bat, oinarriak eta helburuak, etorkizuna eta TEIren kodetze-eskemaren diseinuaren nondik-norakoak argitzen ahaleginduko gara; ondoren, TEIren oinarrian dagoen SGML markatze-lengoaia deskribatuko dugu labur, halako lengoaia baten ezaugarriak eta zertarakoa azaldu eta adibide batzuk emanez; eta azkenik, gure taldean gidalerrook aplikatzen hasiak garenez, zertarako eta nola erabiltzen ditugun azalduko dugu, erabilera horrek dakarzkigun abantailak aipatuz.

2. TEI: Text Encoding Initiative

2.1 Historia eta helburuak

Edozein testuk, konputagailuen laguntzaz aztertuko bada, konputagailuak irakurri ahal izango duen moduan kodeturik egon behar du. Kodekera estandar baten ezean, eta kodekera diogunean software eta hardware bateragarriaz ere ari gara, ezinbestean, testuak ikergai dituzten hainbat arlotako ikertzaileek, testu horien tratamendua erraztearren, hamaika era desberdin asmatu eta erabili izan dituzte. Nork berea —eta bere modura— egin duelarik, ordea, azkenean batek egindakoaz beste bat baliatu nahi izan denean aurretik egindako lan guztia ez da suertatu izan nahi litzatekeen bezain lagungarri, maiz, erabat erabili ezina ere bai. Egoera tamalgarri horren aurrean, saio bat baino gehiago jo izan da azken hogei urte honetan testuen kodeketarako estandar baten bila, fruituak heldu ez badira ere.

1987an, Association for Computers and Humanities (ACH) elkarteak bilera batera deitu zituen 30 aditu (New York-eko Vassar eskolara), testuen kodeketaren estandarizazioa aztergai. Aditu horiek ados etorri ziren esatean aurrera jotzeko ihardupide komun bat ez ezik, literatura- eta hizkuntza-datuak kodetzeko eta trukatzeko gidalerro batzuk ere zehaztea behar-beharrezkoa zela. Eta oraingoan, arrazoiak arrazoi[3], gaiak aurrera egin zuen, eta horren ondorio dira orain ia hiru urte argitara emandako TEI P3 gidalerro horiek. Vassar-ekoaren ondoren, ACHri Association for Literary and Linguistic Computing (ALLC) eta Association for Computational Linguistics (ACL) batu zitzaizkion eta hortik sortu zen TEI izeneko ekimena. TEIren helburuak erdiestea, giza zientzien arloan ez ezik, hizkuntzaren industriaren munduan, oro har, ere

[3] Kodeketaren arazoak eta oinarrizko printzipioak hobeto ezagutzen ziren, aurreko ekimenak izan zirenean baino; garrantzizko erakunde eta ikertokien ordezkaritza zabalagoa lortu zen honetan aurrekoetan baino; SGML lengoaia bazen jada; eta, azkenik, premiaren larritasuna inoiz baino handiagoa zen, itxuraz.

beharrezkoa zela ikusi zen berehala, ikerkuntzan zein industrian ezinbesteko bihurtu baita, edozein testu hainbat aplikaziotan erabili edota, hobe, berrerabili ahal izatea, testua kodetu zen garaian imajinatu ere egin ez ziren aplikazioetan barne.

Hona hemen Vassar-eko eskolako bilkura hartan ezarritako printzipio batzuk (2):

- Gidalerroen helburua giza zientzietako ikerkuntzan datu-trukerako eta testuen kodeketarako formatu estandarra eskaintzea da.
- Gidalerroek gomendatu behar lukete sintaxi jakin bat formatu horretarako, testu-kodetze eskemak deskribatzeko metalengoaia bat definitu, eta formatu berria deskribatu metalengoaia horretan eta hizkuntza arruntean.
- Aldez aurretik kodetutako testu konputagailuz irakurgarriak formatu berrira itzultzeak, haien kodeketa-sintaxia formatu berrikora aldatzea esan nahi du, baina ez da eskatuko testu haietan ez zegoen informaziorik eransterik.

Gidalerroen garapenean, TEIk identifikatu zituen askotariko ikertzaileek zer-nolako kodetze premiak zituzten informazioaren elkartrukeari zegokionean, horretan oinarritu zituen orokor izan nahi zuen eskema batek betebeharreko kodeketa-printzipioak, eta identifikatu zituen zein ziren kodetze-arauak behar zituzten testu-klase eta -ezaugarriak. TEIk eskaintzen dituenetan arakatzen hasita, hona hemen batzuk:

- SGML (Standard Generalized Markup Language) markatze-lengoaia egokitzat jotzea gidalerroen garapenerako oinarri gisa.
- SGML erabiltzeko gomendioak —zenbait murriztapen—, bere orokortasunari eta malgutasunari eutsiz aldi berean.
- Testu-datuak kodetzerakoan beharrezko diren kategoria eta ezaugarrien identifikazioa eta analisia, maila askotan.
- Testu-egitura definizio orokorren multzo malgu eta hedagarria.
- Testu elektronikoak dokumentatzeko metodo bat, biblioteketan erabiltzen den katalogatze-arauekin bateragarria dena.
- Kodetze-arauak testu-mota eta -ezaugarri desberdinetarako: karaktere-multzoak, hizkuntz corpusak, linguistika orokorra, hiztegiak, datu terminologikoak, ahozko testuak, hypermedia, literatur prosa, olerkia, antzerkia, iturburu historikoak, eta testu-kritikarako aparatua.

Hasiera-hasieratik TEI eskema diseinatu zen hardwarea, softwarea eta aplikazioetatik independente izateko helburuarekin. Aplikazioetatik independente izan nahi horrek izugarrizko garrantzia du, gure ustez, eta testu baten ikuspegi desberdinak kodetu ahal izateko aukera ematen digu. Izan ere, testu bat har baitaiteke objektu fisikoen bilduma bezala (liburukiak edo paper-orri solteak), edo objektu tipografikoen segida bezala (karaktere-sekuentziak, letra-molde eta marjina-eskema desberdinen arabera antolatuak), edo objektu linguistikoen sekuentzia bezala (grafema edo fonemak, morfemak, unitate lexikalak, sintagmak, ...), edo objektu formalez osatutako egitura bezala (estrofak, lerroak, kapituluak, atalak, ...), eta abar eta abar. TEI gidalerroek helburu orokorreko kodetze-eskema bat definitzen dute, ikuspegi desberdin horiek guztiak era desberdinetan kodetzeko aukera ematen duena, eta, nahi izanez gero, aldi berean gainera.

Erabiltzaile askoren premiak betetzera datoz TEI P3 gidalerro hauek: zientzia eta giza arloko ikertzaileak, argitaratzaileak, bibliotekariak, eta, oro har, dokumentuen bilaketa eta biltegitzearekin zerikusia duten guztiak. Erantzun bat ematen dio, orobat, "hizkuntzaren teknologiaren" arloko jendeari, orotariko testu-corpus eta lexikoen pilatzeari emanak baitaude azken aldi honetan, hizkuntzaren ulerkuntza, sorkuntza eta itzulpenari dagokion ikerkuntzan sartuak.

TEI gidalerroak SGMLren gainean "eraikiak" dira, beraz. SGMLk testuak markatzeko oinarrizko tresneria baino ez du eskaintzen, gauzak egiteko hamaika bide zabalik utziz. Izan ere, SGML marka-multzo bat baino marka-multzoak espezifikatzeko metalengoaia baita, eta metalengoaia horretaz baliatuz diseinatu dira TEI gidalerroak. SGML, gainera, beste eskema batzuk baino egokiagoa da ikerkuntzari begira egindako testu-kodeketarako, eta 1987tik aurrera oso onarpen zabala jaso du mundu osoan administrazio, industria zein talde akademikoen aldetik. Hortaz, TEI gidalerroek, SGML bere egitean, beste kodetze-eskema askorekiko bateragarritasuna lortu dute de facto.

2.2 TEIren etorkizuna

TEIk lortu du halako mugarri bat ezartzea testuen kodeketari dagokionez, eta gaur egun munduan testuen kodeketaren arazoari aurre egin behar dionak kontuan dauka gidalerroon existentzia. Diseinatutako eskemak erantzun bat ematen die jada kodetze-proiektu gehienen oinarrizko premiei; hala ere, gidalerroak zabaldu egin behar dira eta, batik bat, erabiltzaileak behar-beharrezko diren laguntza-tresnez hornitu behar dira. Izan ere, estandar bat “saltzeko” modurik hoberena estandar horren erabilera erraztuko duten software-tresna eta baliabideak garatzea baita.

Beraz, TEIren lanik garrantzitsuena egun da gidalerrook zabaltzea, erabiltzaileei begira ez ezik, erabilpenei begira —sakon aztertu gabe geratu diren arloetan-eta, batez ere—, eta beren probaketa, zuzenketa eta hedapena ahalbideratuko duten mekanismoen inplementazioari ekitea.

2.3 Kodetze-eskemaren diseinua dela-eta bi hitz

Kodetze-eskemaren diseinuan ezarritako helburuak hauek izan ziren:

- gidalerroek ikerketarako beharrezko diren testu-ezaugarriak errepresentatzeko aski izan behar dute;
- sinpleak, argiak eta zehatzak izan behar dute;
- erabilterrazak, software berezirik gabe erabili ahal izateko modukoak;
- testuak zehazki definitu eta eraginkorki prozesatzeko aukera eman behar dute;
- erabiltzaileari hedaketak egiteko aukera eman behar diote; eta,
- bat etorri behar dute dauden estandarrekin.

Geroak esango du, noski, helburu horiek bete diren ala ez. Gauza batzuk argi daude, ordea. Argitasuna eta ulerterraztasuna direla eta, egileek berek aitortzen dute hasieran espero zutena baino askoz konplexuagoak irten direla, azkenean, gidalerroak, baina diote, eta arrazoi osoz gure ustez, eskema sinpleago batek nekez erantzungo liekeela P3 honek bezala ikertzaileen premiei. Gidalerroek ikerketarako beharrezko diren testu-ezaugarriak errepresentatzeko aukera asko eta asko ematen dituzte, eta oro har aitortzen zaie egun diren premia gehienetarako baliagarritasuna. Bestalde, gidalerroen diseinuak berak beren hedagarritasuna bermatzen du, eta, horretara, estandarrean definituriko elementuekin-eta aski ez duen erabiltzaileak aukera guztiak ditu DTDak bere premietara egokitzeko (gidalerroetan, gainera, adierazten zaio nola egin gauzak nahasten ibili gabe). Software berezirik gabe erabili ahal izateaz den bezainbatean, berriz, esan beharra dago izan, posible dela editore estandarrez-eta baliatuz lan egitea, baina aitortu behar da askoz ere emaitza hobeak eta eraginkortasun handiagoa lortuko dela inondik ere editore bereziak eta, oro har, software berezitua erabiliz gero.

3. SGML: testuak markatzeko lengoaia estandar eta orokorra

1978. urtean, American Standard National Institute (ANSI) erakundeak testu-prozesamenduan ari ziren hainbat talde jarri zituen harremanetan, edozein motatako testuak kodetzeko, egituratzeko eta elkarren artean trukatzeko balioko lukeen lengoaia estandar eta orokorra definitzeko helburuarekin; 1980 urterako lengoaia horren lehenengo txostenak argitaratu baziren ere, 1985 urtean elkar-lanaren emaitzaren azken bertsioa argitaratua izan zen, International Standard for Organization (ISO) erakundeak estandartzat onartu zuena: ISO 8879 edo SGML (Standard Generalized Markup Language) lengoaia.

SGML lengoaian, testu barnean markatze-kodeak txertatzen dira eredu bati jarraituz eta modu deskriptiboan egituratzen da, programazio-lengoaien edo aplikazioen menpe sortzen diren egitura berezituak saihesten direlarik.

3.1 Markatze deskriptiboa

Markatze-sistema deskriptiboek markatze-kodeak erabiltzen dituzte dokumentuaren zatiak izendatzeko. Horrela bada, *<esaldia>* edo *</esaldia>* bezalako kodeak dokumentu batean txertatzen ditugunean, testu horren zati baten hasiera edo bukaera adieraziko genuke.

Izan ere, testu arruntak idazterakoan (honako hau, kasu) markatze-kodeak erabiltzen ditugu beti, testuari egitura inplizitu bat emanez (hitz larriak esaldi hasieran, puntuazio-markak esaldien zatiak bereizteko, lerro hutsak eta tabuladoreak paragrafoak bereizteko, etab.); are gehiago, testu bat ulertzerakoan ere markatze-kodeak erabili ohi ditugu (esaldien zati garrantzitsuenak azpimarratuz, etab).

Bestalde, markatze-sistema prozedurala deritzoguna dugu. Markatze-sistema prozeduraletan, testuan ipintzen diren kodeak konputagailu-programa baten instrukzioak dira, zeintzuetan aplikazio horri ekintza batzuk egikaritzea eskatzen zaion (Adib. mugitu ezkerreko marjina 5 zentimetro eskuinera, etab).

Azken mota horretako markatze-lengoaiak testu-prozesadoreek lehendik erabiltzen bazituzten ere (RTF, LaTex, Postscript), modu horretan kodetuak dauden fitxategiek ez dute estandarren oinarrizko baldintza bat betetzen, hots, ez dira aplikazioekiko independenteak. Izan ere, estandar izateko bokazioa duen lengoaia orok, makinekiko, aplikazioekiko eta teoria linguistikoekiko independentea den testu-egituratzerako sistema eskaini behar du nahitaez. Horrez gain, markatze prozedurala bideratzen duten lengoaia horien helburua testuen "itxura" (letra-motak, marjinak, etab) , ez beren barne-egitura, kodetzea litzateke eta beraz testu horiek kudeatzeko ahalmen eskasa eskaintzen dute.

3.2 DTD: dokumentu-motaren definizioa

SGMLk dokumentu-motaren nozioa ezartzen du: dokumentu bakoitza mota batekoa izango da eta dokumentu-mota batek dokumentu-multzo bat definituko du. Dokumentu-mota hau DTD (Document Type Definition) delako fitxategian definitzen da eta dokumentu guztiek DTD bati esleituak egon behar dute.

DTDan dokumentuaren mota formalki definitzen da, hau da, bere osagaiak eta egitura esplizituki adierazten dira. Adibidez, txosten-motako dokumentu bat definitzerakoan, hasieran egilearen izena etorriko dela esan beharko dugu, ondoren laburpen bat eta, azkenik, txostenaren muina daukaten hainbat paragrafo. Esaten da SGML lengoaia meta-lengoaia dela, DTDen bidez (edo, SGML terminologia erabilita, "aplikazioaren" bidez) azpi-lengoaiak definitzen baitira. Egun hypermedia munduan eta Internet bezalako sare informatikoetan hain ospetsua eta erabilia den HTML lengoaia, SGMLren DTD bat besterik ez da.

3.2.1 Adibidea.

Adibide gisa, demagun poemak gordetzeko barne-egitura definitzen dugula.

Antologia bat hainbat *poemaz* osaturik egongo da.

Poemek, *izenburu* bana eta hainbat *estrofa* izango dituzte.

Estrofak hainbat *lerroz* osaturik egongo dira.

SGML lengoaian kodetuta, egitura horrekin bat datorren testu batek horrelako itxura izango luke:

```
<antologia>
  <poema>
    <izenburua>Potaren galdatzia</izenburua>
      <estrofa>
        <lerroa>Andria, Ieinkoak drugatzula; orai berdi
                girade</lerroa>
        <lerroa>ni errege balin baninz, erregina zinate;</lerroa>
        <lerroa>pot bat, othoi, egidazu; etzaitzula herabe;</lerroa>
        <lerroa>nik zugatik dudan penek hura merexi dute.</lerroa>
      </estrofa>
      <estrofa>
        <lerroa>Eia horrat, apart´ adi; nor uste duk
                nizala?</lerroa>
        <lerroa>Horlako bat eztuk uste nik ikusi dudala;</lerroa>
        <lerroa>horrelako hitz gaixtorik niri eztarradala;</lerroa>
        <lerroa>berzer erran albaitzita; enuk uste duiana.</lerroa>
      </estrofa>
  </poema>
  <poema>
    <izenburua>...
      ...
  </poema>
  ...
</antologia>
```

Kontuan hartu adibide honetan testuaren zatiak era batera kodetuak daudela, baina horrek ez du esan nahi testuaren zatiak bereizteko bestelako modurik ez dagoenik. Erraz asma daitezke kodetzeko beste ezaugarri batzuk non, gure interesen arabera, informazio desberdina bereiz daitekeen (adibidez, ahozko corpusak kodetzeko). Horregatik esaten da, hain zuzen ere, DTDekin testuaren *interpretazioak* kodetzen direla.

Testu honi dagokiokeen DTDa honako hau litzateke:

DTD baten adibidea

```
<ELEMENT antologia      - -   (poema+) >
<ELEMENT poema          - -   (izenburua?, estrofa+) >
<ELEMENT izenburua      - O   (#PCDATA) >
<ELEMENT estrofa  - O   (lerroa+) >
<ELEMENT lerroa   O O   (#PCDATA) >
```

Ikusten denez, DTDan testuan erabiliko diren markatze-kodeak edo, SGML terminologia erabilita, elementuak definitzen dira. Elementu bat definitzerakoan, 3 zati bereiz daitezke: elementuaren izena, minimizazio-erregelak eta eduki-eredua.

3.2.2 Elementuaren izena.

Lehenengo eremua (ELEMENT hitz erreserbatuaren ondoren) deklaratzen ari den elementuaren identifikatzailea da.

3.2.3 Minimizazio-erregelak.

Definizioaren bigarren zatiak —minimizazio-erregelak—, izenaren ondoren datozen bi eremuz osatua dago, eta elementuaren hasiera-marka eta bukaera-markaren beharra adierazten digu. Eremu hauen balio posibleak '-' edo 'O' dira. Normalean testuaren zatiak bereizteko hasiera-marka eta bukaera-marka erabili behar dira. Gerta daiteke, hala ere, marka batzuen beharra ez edukitzea; adibidean, *izenburua* elementuak bukaera-markarik ez du behar, gramatikaren arabera argi geratzen baita izenburuaren ondoren bakarrik estrofak etor daitezkeela. Horrelako kasuetan, 'O' (hautazkoa) balioa ipintzen zaio dagokion eremuari eta beste guztietan '-' (derrigorrezkoa) balioa.

3.2.4 Eduki-eremua.

Azken zati honetan elementuaren osagaiak eta haien distribuzioa azaltzen da. Osagaiak elementuak edo entitate "bereziak" (hitz erreserbatuen bidez adierazitakoak) izan daitezke, hitz erreserbatuen artean usuen erabili ohi dena *#PCDATA* (edozein karaktere adierazten duen entitatea) delarik. Adibidean, *antologia* elementua *poema* elementuaz dago osatua eta *izenburua* elementua, ostera, *#PCDATA* karakterez.

Eduki-eremu honen bitartez dokumentuaren gramatika deskribatzen da. *#PCDATA* elementuak gramatikako terminalak dirateke.

Osagaien distribuzioa definitzeko konektoreak erabiltzen dira. Konektore asko dauden arren, guztien esanahiaren deskribapena ez da artikulu honen helburua eta beraz konektore garrantzitsuenen deskribapena besterik ez da emango:

Konektorea	**Deskribapena**
Osagaia+	Osagaia behin edo gehiagotan azaltzen da.
Osagaia*	Osagaia zero edo gehiagotan azaltzen da.
Osagaia?	Osagaia zero edo behin azaltzen da.
Osagaia1 , Osagaia2	Bi osagai konektatzen ditu bata bestean atzetik. Osagaia1 eta gero Osagaia2
Osagaia1 | Osagaia2	OR konektorea. Edo Osagaia1 edo Osagaia2 (biak ez)
Osagaia1 & Osagaia2	AND konektorea Osagaia1 eta Osagaia2 (ordenak ez du garrantzirik)

Adibidez, (izenburua? , (lerroa+ | estrofa*)) espresioak zera adierazten du: izenburua (edo ez) etor daitekeela lehendabizi, eta gero hainbat lerro edo zero edo gehiago estrofa etorriko direla.

4. TEI eta SGM

Zer esan nahi dugu, beraz, TEI gidalerroak SGML lengoaian oinarrituta daudela esaten dugunean? Bada TEI gidalerroak SGMLren aplikazio bat besterik ez direla, hots, DTD multzo bat non ezaugarri desberdinetako testu motak kodetzeko kodeak eta beraien distribuzioa definitzen baitira.

Adibidearekin jarraituz, testua kodetzeko guk egindako DTDaren ordez TEIk poesiarako eskaintzen duen DTDa erabiliko bagenu, adierazpide ahalmena aberatsagoa izateaz gain, TEI gidalerroei jarraitzen zaizkien beste testu guztiekin bat etorriko litzateke. Horren garrantzia oso handia da, haientzat eraikitako tresnak gure testurako baliagarriak izango baitira, eta, bai haien testuak, bai gureak, berrerabilgarriak izango direlako. Hauek dira, azken finean , estandarizazioaren abantailak.

5. TEI gidalerroen aplikazioak

Gorago esan bezala, TEI gidalerroen helburua datu-trukerako eta testuen kodeketarako formatu estandar bat eskaintzea da. Beraz, TEI zertarako erabil litekeen pentsatzen jarriz gero, aukera amaigabea da, eta areago testuak euskarri elektronikoan ematea —CD-ROMak, Internet, e.a.— gero eta normalago den kinka honetan.

TEI-P3k bere mugak ditu, ordea, gidalerroak arlo batzuetarako bakarrik baitaude garatuta, eta ez pentsa litezkeen guztietarako. Dena den, esan behar da arlo asko izan dituztela kontuan, eta espero izatekoa da etorkizunean iritsiko diren hedakuntzek beste batzuk ekarriko dituztela.

Hona hemen, bada, eta gidalerroen liburutik bertatik zuzenean aterata (Sperberg-McQueen & Burnard, 1994), TEI-P3k zein erabileratarako eskaintzen dituen gomendioak:

- Prosa, bertsoa, eta drama motetako testuak.
- Ahozko hizketaren transkribapena.
- Hiztegiak.
- Datu-base terminologikoak.
- Testuen edo testu-zatien arteko loturak, segmentazioa eta alineamendua kodetzeko.
- Hizkuntz analisia.
- Ezaugarri-egiturak (feature structures).
- Iturburu primarioen transkribapena (eskuzkribuak-eta).
- Testu-kritikarako aparatua.
- Izenak eta datak.
- Taulak, formulak eta grafikoak.
- Hizkuntz corpusak.

Horrez gain, TEI-P3k testu guztiek amankomunean dituzten hainbat arazoren trataera nola egin ere azaltzen du, hala nola karaktereen errepresentazioa, zerrendak, aipuak, erreferentzia gurutzatuak, termino teknikoak, izenak, datak, zenbakiak, eta abar.

Bukatzeko, eta SGMLren Interneteko orrira[4] jo beza gehiago jakin nahi duen irakurleak handik zuzenean harturik baitaude ondorengoak, eman ditzagun SGMLren aplikazio diren lan batzuen erreferentziak.

TEI-P3 gidalerroekin bat datozen proiektu guztien artean banaka batzuk aipatzearren, aipa ditzagun Oxford Text Archive (OTA), idazle handien literatura-lanak (2000 titulu inguru, hizkuntza anitzetan: grekoa, latina, ingelesa, ...) biltzen dituena, HTI American Verse Project, XIX. mendean Estatu Batuetan idatzitako poesiaz osatua, British National Corpus Project (BNC), 100 milioi hitzeko corpus erraldoia, ahozko zein idatzizko hizkuntzaren bilgune, Oxford Univ. Press-en gidaritzapean garatzen ari dena, eta European Corpus Initiative (ECI), corpus eleanitz handi bat biltzea helburu duena, gero ikerketa zientifikoan erabili ahal izateko (dagoeneko 100 milioi hitzetatik gorakoa).

SGMLren beste aplikazio interesgarri bat —kasu honetan TEIrekin bat ez badator ere— University of Waterloo Centre for the New OED delakoak Oxford English Dictionary-ren inguruan garatzen ari dena da.

Aipa dezagun azkenik, eta Euskal Herrian corpus elebidunak edukitzeak itzulpengintzarako eta bestetarako eduki dezakeen garrantziaz jabeturik, Osloko Unibertsitateko English-Norwegian Parallel Corpus Project delakoa —hau ere TEIrekin bat datorrena—, helburu duena ingelesezko eta norvegierazko testu-corpus paralelo bat biltzea hizkuntzaren tratamendu automatikoan erabiltzeko, eta testu paraleloen azterketarako tresna multzo bat prestatzea.

6. TEIren erabilera IXA taldean

IXA taldea Euskal Herriko Unibertsitateko Informatika Fakultatean diharduen ikertaldea da. Bere ikerkuntza hizkuntzaren tratamendu automatikoaren arloan egiten du, eta azken urteotan garatutako lanak hizkuntzaren tratamendurako oinarria finkatzea izan du helburu.

Helburu honi erantzuteko zutabea Euskararen Datu-Base lexikala (EDBL) (3), lexikoari buruzko informazio-biltegia dugu, 60.000 sarreratik gora dituen datu-basea, bakoitza bere kategoria eta informazio morfologikoarekin. Funtzioari buruzko hainbat informazio ere gehitu da, sintaxia eta semantikarako bidea irekiz. Hizkuntza-tratamendurako tresnei dagokienez, berriz, EDBL euskarri duten segmentatzaile eta analizatzaile morfologikoa ditugu, eta lematizatzaile orokorra. Euskal morfologiaren tratamendu automatikoa gauzatzeko bi mailatako morfologiaz baliatu gara (4).

4 http://www.sil.org/sgml/acadapps.html

Taldeak helburu du, orobat, euskararen normalizazioari eragitea, sistematizazio eta formalizaziorako tresna lagungarriak garatuz lengoaia naturalaren prozesamenduaren ildotik (LNP). LNPrako aipatu oinarriak TEIko gidalerroen proposamenetara egokitzea on litzatekeela uste dugu, horrela euskara aztergai edo lan-tresna duen edonorentzat baliagarri izango baitira.

Horrela bada, oraingoz bi proiektutan ari gara bide horretan, TEIren araberako kodeketa-lanean, alegia: batetik, lexikografia konputazionalean, Hauta-Lanerako Euskal Hiztegiaren (HLEH) kodeketari ekin diogu; bestetik, EDBL, segmentatzailea, analizatzaile morfologikoa (Morfeus) eta lematizatzailearen (EusLem) arteko komunikazio osoa ere TEIren arabera kodetzeko beharrezko definizioak egin ditugu, barne-prozesuen arteko informazio-truke bateraturako ez ezik, kanpora begira ere informazio-molde estandarizatua eskaintzeko asmoarekin. Esan gabe doa hizkuntz tresna horien sarrera izango diren testuek eta corpusek ere TEIren arabera kodetuta joan beharko dutela, nahiz eta oraindik ez diogun heldu horri.

6.1 HLEH hiztegia TEIko gidalerroei jarraituz

EDBL aberasteko eta osatzeko Hauta-Lanerako Euskal Hiztegia (5)[5] hartu dugu "ezusteko" iturri lexikal legez, kontuan harturik informazio-iturri joria dela. "Ezusteko" diogu berez hiztegia giza erabiltzaileari zuzenduta dagoelako eta ez baitu gure helburuetarako nahi genukeen egitura, hots, ordenadore bidez zuzenean probetxatzeko modukoa. Beraz, lehenbiziko lana, prozesamendu automatikoari heldu aurretik hiztegi-testuaren egituratzea izan da.

Giza erabiltzaileari zuzenduriko hiztegi elebakar zein eleanitzak kodetzeko oinarrizko etiketa-multzoa proposatzen da TEI gidalerroetan. Hauetan batez ere kodeketa arazo orokorrak planteiatzen dira, kontuan izan behar baitira hiztegien konplexutasuna (bai tipografikoki bai egitura aldetik) eta erabiltzaile desberdinen interes konfliktiboak. Bi arazo nagusi aipatzen dira:

1) Hiztegi-sarreren egitura aldakorra, bai hiztegi berean bai hiztegitik hiztegira. Kodekera-eskemak egitura horiek guztiak jaso ditzan, elementu orori hiztegi-sarreraren edozein posiziotan azaltzeko aukera emango zaio. Dena den, hiztegi gehienak egiturazko printzipio berdintsuei jarraitzen zaizkie, ideialki etiketatzeko gomendioek isla ditzakete printzipio horiek. Bi elementu-mota definitzen dituzte hiztegi-sarrerak errepresentatzeko: <entry> zeinak hiztegi

5 Behin-betiko kodeketa, baina, I. Sarasolak argitara berri duen Euskal Hiztegia-ren (6) gainean egingo da.

konbentzionalen ezaugarriak biltzen dituen, eta <entryFree> zeinak oinarrian elementu berdinak erabiltzen dituen, baina konbinazio askeagoak baimenduz.

2) Hiztegietako informazio ugari inplizitua da edota oso konprimitua dago. Hau dela eta, garbi izan behar da azaleko testu-egitura edo aurkezten den informazioaren barne-egitura jaso nahi den. Ikuspuntu biak interesgarri izan daitezke, bien berri emango diguten metodoak garatu behar direlarik, testu-egitura eta barne-egitura harremanetan jartzeko errepresentazio moduak ere landuz.

TEIk proposaturiko gomendioak jarraitzeari egoki deritzogu, besteak beste arrazoi hauengatik:

- Lexikografoen lanerako onuragarria delako, informazioaren egituraketari dagokionean kodeketa-sistema finkoa eskaintzen baitu. Ez da inola ere gauza bera, esate baterako, testu-prozesadore batez editatutako hiztegia, testu huts dena, edo honelako adierazpide aberatsago bat erabiliz eratutakoa.
- Honen inguruan garatzen ari diren tresnak kontuan harturik, informazioaren atzipena, kontsulta-aukeren ugaltasuna eta aberastasuna, etab. ahalbidera daitezkeelako.
- Estandarra denez, informazioaren berrerabilgarritasuna eta elkartrukatzea gauza baitaitezke.

Abantaila hauek kontuan harturik, HLEHren azterketaren emaitza TEIn proposatzen diren etiketen bidez adieraztea erabaki dugu. HLEH irakurgarria da ordenadorez: testu-fitxategi baten gisara, hiztegiko artikuluak sekuentzialki editatu dira eta informazio motak banatzeko bereizgarri tipografikoak erabili dira. Horrela bada, HLEH informazio-iturri aberatsa da, baina informazio hori era gordinean dago eta, giza erabiltzaileari zuzenduta dagoenez, inprimatutako bertsioen akatsak ditu, bereziki informazioaren egituraketa desegokia LNPren ikuspegitik. Esan bezala, beraz, premiazkoa da edozein tratamendu automatikori heldu aurretik hiztegi-testua egituratzea.

Egituratze-lanaren emaitza gisa, hiztegiaren analisia erdietsi dugu, lorturiko emaitza egituratu honetatik formatu estandarrera pasatzea erabaki da, hots, TEIn proposatzen diren etiketen bidez adieraziko dugu analisiaren irteera. Hona hemen *ahalketu* sarrera TEI formatuan jarririk:

```
<entry>
  <form><orth>ahalketu, ahalke, ahalketzen. </orth></form>
  <GramGrp><subc>da-du </subc><pos>ad. </pos></GramGrp>
  <usg type=time>(1571)</usg>
  <usg type=geo>. Ipar. edo jas. </usg>
  <sense n = 'A.'>
    <GramGrp><subc> da </subc><pos>ad. </pos></GramGrp>
    <def>Lotsatu. </def>
    <eg><q>Ahalke zaitez, zerbitzari laxoa. Erori orduko ahalketu
        ziren, eta estali zituzten beren gorputzak piko-hostoz.</q>
    </eg>
    <sense n = '#1'>
      <eg><q> Zerbaitez edo norbaitez ahalketu. Ahalketzen baita
            erokeria haiek ikusteaz eta entzuteaz.</q>
      </eg>
    </sense>
  </sense>
  <sense n = 'B.'>
  <GramGrp><subc> du </subc><pos>ad. </pos></GramGrp>
  <xr type = syn>Ik. ahalkerazi. </xr>
    <sense n = '1. '>
      <usg type=time>(1571).</usg>
      <def> Lotsatu. </def>
      <eg><q>Neure bihotzeko hasperenek ahalketzen ninduten.</q>
      </eg>
    </sense>
    <sense n = '2. '>
      <usg type=time>(XIX b. ).</usg>
      <def> Beldur erazi. </def>
      <eg><q>Mamu horrek txoriak ahalketzen ditu.</q>
      </eg>
    </sense>
  </sense>
</entry>
```

6.2 Hizkuntza-tratamendurako tresnen arteko komunikazioa: EDBL, segmentatzailea, Morfeus eta EusLem-en irteera-formatuak TEIko proposamenetara egokituz

EDBL (datu-base lexikala), segmentatzaile morfologikoa, Morfeus analizatzaile morfologikoa eta EusLem lematizatzaile orokorraren irteera-formatuak TEIko proposamenetara egokitzeko TEIk ezaugarri-egiturak kodetzeko proposatzen dituenak izan ditugu kontuan.

Ezaugarri-egiturak (feature structures) linguistika konputazionalean eta beste arlo askotan sarri erabiltzen diren adierazpidea ditugu, eta inondik ere ezin egokiagoak hizkuntz tratamendurako tresna horien informazioaren sarrera-irteerak errepresentatzeko.

EDBLtik jasotzen du informazioa segmentatzaile morfologikoak bere lana egiteko; segmentatzailearen irteera (hitzen morfema-zatiketa, morfema bakoitzari dagokion informazio morfologikoarekin) sarrera du Morfeusek; Morfeusek hitz barneko analisi morfosintaktikoa egiten du eta emaitzatzat hitzaren analisi morfosintaktikoa dei genezakeena ematen du, ondoren lematizaziorako edo analisi sintaktikorako sarrera izango dena. Lematizatzailearen emaitza, azkenik, Morfeusenaren bertsio sinplifikatua baino ez da, anbiguotasuna ahal den neurrian ebatzi eta hitz anitzeko unitate lexikalen tratamendua egin ondoren.

Gorago esan bezala, bada, tresna horien arteko informazio-trukea normalizatzeko baliatu gara TEIz, eta lau irteera-formatu definitu ditugu: irteera.edbl.fsd, irteera.seg.fsd, irteera.morf.fsd eta irteera.tag.fsd. Hau ez da tokia formatu horien xehetasunak ematen hasteko, baina nahi genuke, hori bai, formatu hauetan kodetutako adibide pare bat behintzat erakutsi. Hona hemen adibideok:

- lehenak segmentatzailearen irteera-formatuaren itxura erakusten du (.seg.tei formatua, irteera.seg.fsd-rekin bat datorrena), *softwaregileek* hitzaren kasuan (emanik hitz honek bi segmentazio ematen dituela, *softwaregile+ek* eta *software+gile+ek*);

- bigarrenak hitz beraren lematizazioaren emaitza erakusten du (.tag.tei formatua, irteera.tag.fsd-rekin bat datorrena).

```
<!--Segmentatzailearen irteera (.seg.tei): softwaregileek-->
<!--Bi segmentazio:  softwaregile lexikalizatutzat, EDBLko-->
<!--sarreratzat hartuta bata;-->
<!--              software+gile bezala analizatuta bestea-->
<tei.2>
<fs type="segmentazio-kohortea">
  <f name="forma"><str>softwaregile</str></f>
  <f name="segmentazioak" org="list">
    <fs type="segmentazioa">
      <f name="lema-morfemak" org="list">
        <fs type="lema">
          <f name="twol"><str>softwaregile</str></f>
          <f name="sarrera">
            <fs type="gako">
              <f name="Sarrera"><str>softwaregile</str></f>
              <f name="Homografo-Id"><nbr value="0"></f>
            </fs>
          </f>
          <f name="ezaugarriak">
            <fs type="ezaugarri-lista">
              <f name="KAT"><sym value="IZE"></f>
              <f name="AZP"><sym value="ARR"></f>
              <f name="BIZ"><plus></f>
              <f name="OIN">
                <fs type="gako">
                  <f name="Sarrera"><str>software</str></f>
                  <f name="Homografo-Id"><nbr value="0"></f>
                </fs>
              </f>
```

```
          <f name="ATZL" org="list">
            <fs type="gako">
              <f name="Sarrera"><str>gile</str></f>
              <f name="Homografo-Id"><nbr value="1"></f>
            </fs>
          </f>
        </fs>
      </f>
    </fs>
    <fs type="morfema">
      <f name="twol"><str>ek</str></f>
      <f name="sarrera">
        <fs type="gako">
          <f name="Sarrera"><str>ek</str></f>
          <f name="Homografo-Id"><nbr value="1"></f>
        </fs>
      </f>
      <f name="ezaugarriak">
        <fs type="ezaugarri-lista">
          <f name="KAT"><sym value="DEK"></f>
          <f name="KAS"><sym value="ERG"></f>
          <f name="NUM"><sym value="P"></f>
          <f name="MUG"><sym value="M"></f>
          <f name="FSL" org="list">
            <sym value="@SUBJ">
          </f>
        </fs>
      </f>
    </fs>
  </f>
</fs>          <!-- hemen bukatzen da lehen segmentazioa-->
<fs type="segmentazioa">
  <f name="lema-morfemak" org="list">
    <fs type="lema">
      <f name="twol"><str>software</str></f>
      <f name="sarrera">
        <fs type="gako">
          <f name="Sarrera"><str>software</str></f>
          <f name="Homografo-Id"><nbr value="0"></f>
        </fs>
      </f>
      <f name="ezaugarriak">
        <fs type="ezaugarri-lista">
          <f name="KAT"><sym value="IZE"></f>
          <f name="AZP"><sym value="ARR"></f>
        </fs>
      </f>
    </fs>
    <fs type="lema">
      <f name="twol"><str>gile</str></f>
      <f name="sarrera">
        <fs type="gako">
          <f name="Sarrera"><str>gile</str></f>
          <f name="Homografo-Id"><nbr value="1"></f>
        </fs>
      </f>
```

```
          <f name="ezaugarriak">
            <fs type="ezaugarri-lista">
              <f name="KAT"><sym value="ATZ"></f>
              <f name="OINKT"><sym value="IZE"></f>
              <f name="ERAKT"><sym value="IZE"></f>
            </fs>
          </f>
        </fs>
        <fs type="morfema">
          <f name="twol"><str>ek</str></f>
          <f name="sarrera">
            <fs type="gako">
              <f name="Sarrera"><str>ek</str></f>
              <f name="Homografo-Id"><nbr value="1"></f>
            </fs>
          </f>
          <f name="ezaugarriak">
            <fs type="ezaugarri-lista">
              <f name="KAT"><sym value="DEK"></f>
              <f name="KAS"><sym value="ERG"></f>
              <f name="NUM"><sym value="P"></f>
              <f name="MUG"><sym value="M"></f>
              <f name="FSL" org="list">
                <sym value="@SUBJ">
              </f>
            </fs>
          </f>
        </fs>
      </f>
    </fs>
  </f>
</fs>
</tei.2>

<!--EusLem-en irteera (.tag.tei): softwaregileek-->
<tei.2>
<fs id="IZE-ARR-1905" type="analisia">
  <f name="lema"><str>softwaregile</str></f>
  <f name="ezaugarri-morfologikoak">
    <fs type="goimailako-ezaugarri-lista">
      <f name="KAT"><sym value="IZE"></f>
      <f name="AZP"><sym value="ARR"></f>
      <f name="BIZ"><plus></f>
      <f name="OIN"><str>software</str></f>
      <f name="ATZL" org="list">
          <str>gile</str>
      </f>
      <f name="KAS"><sym value="ERG"></f>
      <f name="NUM"><sym value="P"></f>
      <f name="MUG"><sym value="M"></f>
      <f name="FSL" org="list">
        <sym value="@SUBJ">
      </f>
    </fs>
  </f>
</fs>
</tei.2>
```

7. Bukatzeko

TEIren berri ematea izan da artikulu honen helburua. Testu elektronikoak kodetzeko eta trukatzeko gidalerroak zertan diren azaldu da, historia pixka bat egin, oinarriak eta helburuak aipatu, eta, adibide xume batzuen bidez, testu bat, gidalerroon arabera kodeturik nola geratzen den erakutsi nahi izan da.

Hasieran genioenez, euskarazko testuen kodetze-lanei ekin behar dionaren jakinmina piztea ere bagenuen asmo, idazlan hau geure buruari planteatu geniolarik. Artikulu labur bat murritz geratzen da, noski, TEIren aukera guztien berri emateko, baina asmatuko ahal genuen, bederen, jakinmin hori pizten eta gehiago sakondu nahi duenari ateak zabaltzen.

Geure aldetik, IXA taldean, garbi ikusi ditugu bide honetan abiatzeak dituzkeen abantailak eta, artikuluan azaldu bezala, ekin diogu dagoeneko, lexikografia konputazionalaren arloan aurrena, eta barne-prozesuen arteko informazio-trukerako laster.

8. Bibliografia.

(1) Sperberg-McQueen, C.M., L. Burnard, arg. 1994. *Guidelines for Electronic Text Encoding and Interchange.* Chicago & Oxford.

(2) Ide, N., J. Véronis, arg. 1995.*Text Encoding Initiative. Background and Context.* Kluwer Academic: Dordrecht.

(3) Agirre, E., Arregi X., Arriola J. M., Artola X., Insausti J. 1994. *Euskararen Datu-Base Lexikala (EDBL)* . Barne-txostena UPV/ EHU / LSI / TR8-94.

(4) Agirre E., Alegria I., Arregi X., Artola X., Díaz de Ilarraza A., Sarasola K., Urkia M.. 1989 "Aplicación de la morfología de dos niveles al euskara.*", Sociedad Española para el Procesamiento del Lenguaje Natural*, *SEPLN* **8. zb**., 87-102 or.

(5) Sarasola I.. *Hauta-lanerako euskal hiztegia.* Gipuzkoa Donostia Kutxa: Donostia, 1984-1995.

(6) Sarasola I.. *Euskal hiztegia.* Kutxa Fundazioa: Donostia, 1997.