



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Analisi linguistikoen eskuzko sorkuntza eta kontsulten hobekuntza definizio-eskemetatik abiatuta.

Egilea: Zuhaitz Beloki Leitza

Tutoreak: Xabier Artola Zubillaga, Aitor Soroa Etxabe

HIZKUNTZAREN AZTERKETA ETA PROZESAMENDUA

Hizkuntzaren Azterketa eta Prozesamendua Masterreko titulua lortzeko bukaerako
proiektua

2012ko iraila

Sailak: Lengoia eta Sistema Informatikoak, Konputagailuen Arkitektura eta Teknologia,
Konputazio Zientziak eta Adimen Artifiziala, Euskal Filologia, Elektronika eta Telekomu-
nikazioak.

Laburpena

Master bukaerako lan honetan Ixa taldearen analisi-tresnetan erabiltzen den informazio linguistikoaren inguruko ekarpenak egin dira. Egindako lana bi ataletan bereizi da: batetik, analisi linguistikoaren formalizazioaren ingurukoa, analisi horien egitura zehazki definitu baita definizio-eskema batean. Bestetik, definizio hori erabili da anotazio linguistikoak kudeatzeko erabiltzen den Armiarma aplikazioaren bi funtzionalitate garatzeko. Funtzionalitate horiek analisi berrien eskuzko sorkuntzarako laguntza dinamikoa eta testu-corpusen gaineko kontsultak definitzeko interfaze dinamikoa dira, bai sorkuntza eta bai kontsultak erabilgarri eta dinamiko egiteko analisisien definizio-eskemak erabiltzen baitira oinarrian.

Gaien aurkibidea

1	Sarrera eta helburuak	5
2	Aurrekariak	7
2.1	Anotazio linguistiko banatua: XML dokumentuen amarauna	7
2.1.1	Ezaugarri-egiturak	7
2.1.2	Amarauneko dokumentuen egitura	8
2.1.3	Amaraunaren inplementazio bat: informazioa datu-baseetan antolatzen	14
2.1.4	Datu-baseen elikatzea	16
2.2	Armiarma: anotazio linguistiko konplexuak kudeatzeko web-aplikazioa	17
2.2.1	Interfaze grafikoa	17
2.2.2	Sistemaren arkitektura	20
3	Analisi linguistikoaren formalizazioa	23
3.1	Motibazioa: RELAX NG eskemak eta erabilera	23
3.2	Analisiaren XML egitura definituko duten RNG eskemen sorkuntza automatikoa analisi multzo batetik abiatuta	24
3.2.1	Betekizunaren analisisa	24
3.2.2	Algoritmoa	26
3.2.3	Arazoak	30
3.3	Analisiaren normalizazioa	30
3.4	Analisi linguistikoaren definizioa	31
3.4.1	Segmentazioa	31
3.4.2	Morfosintaxia	34
3.4.3	Lematizazioa	39

4	Armiarman: Analisi linguistikoaren sorkuntza eta kontsulta-ahalmena	43
4.1	RNG eskemak kudeatzeko liburutegia	43
4.2	Analisiaren eskuzko sorkuntza Armiarman	45
4.2.1	Algoritmoa	46
4.3	Analisi-kontsultaren hobekuntza Armiarman	49
4.3.1	Algoritmoa	49
5	Ondorioak	51
6	Etorkizuneko lanak	53
7	Eranskinak	55
7.1	Analisi linguistikoaren ezaugarrien antolaketa eta multzokatzea	55
7.2	Analisiaren Relax NG definizio-eskemak	57

1 Sarrera eta helburuak

Hizkuntzaren prozesamendu automatikoa beharrezkoa da gaur egun erabiltzen diren hainbat tresna garatu ahal izateko. Itzultzaile automatikoak, galdera-erantzuneko sistemak, informazioaren erauzketa egiteko tresnak... denek dute oinarrian informazio linguistiko konplexuz osatutako biltegi bat. Informazio hori erabiliko dute, ondoren, beren eginkizunak burutzeko, dela testu bat hizkuntza batetik beste batera itzultzeko edo historiari buruzko galdera bati erantzuna emateko.

Mota horretako tresna batek behar duen informazio-kantitatea handia da oso. Horregatik, testu-iturri bat hartu eta corpus bat eraikitzen da bertatik lortutako testuekin. Horrek ezagutzaz hornituko du tresna, adibidez, erabiltzaile batek egindako mundu errealeko galdera bati erantzuteko gai izateko. Baina bidean arazo handi bat sortzen da, izan ere, konputagailuak nekez interpretatu baitezake testu gordina.

Aipatutako tresnek testuekin lan egin ahal izateko testuak prozesatu egin behar izaten dira lehenbizi, hau da, testua landu, analizatu, linguistikoki anotatu, desanbiguatu etab. egin behar da. Behin prozesu hori amaituta, eta testua informazioz eta anotazioz aberastu dugula, orduan izango gara gai landutako testu horiekin lan egiteko gai den tresna linguistiko bat garatzeko.

Testuen prozesamenduan urrats ezberdinak ematen dira: testuko osagaiak identifikatu eta hitzetan edo tokenetan banatu, hitzen osagaiak edo morfemak identifikatu, hitz edo morfema bakoitzari dagokion informazio linguistikoa esleitu (kategoria, kasua, numeroa...), desanbiguatu... batzuetan analisi sintaktikoa ere egiten da, eta, gaur egun, analisi semantikoa egiteko zenbait urrats ere eman dira. Prozesuaren fase horietako bakoitza tresna batek egiten du. Beraz, prozesuaren sarreran testu-gordina izango dugu, eta irteeran, prozesatutako eta anotatutako testua. Baina tarteko pausoetan informazioa eraldatzen, edo gehitzen, doa.

Informazio hori guztia adieraztea eta antolatzea ez da berehalako lana, bere izaera konplexua baita askotan, eta informazio konplexuarekin lan egingo duten tresnak garatu ahal izateko ezinbestekoa da informazio hori behar bezala egituratuta egotea. Hor kokatzen da masterreko tesi honen zati handi bat, Ixa taldean anotaziorako erabiltzen den Amarauna (Artola et al., 2009, 2005) izeneko formatua fintzean. Zehazki, analisi linguistikoaren egitura definitzea izan da lanik garrantzitsuenetakoa. Izan ere, duela urte asko definizio hori egin zen, baina orduan erabilitako lengoaiaren gabeziak, batetik, eta urteak pasatu ahala zaharkituak gelditu izana, bestetik, definizioak Relax NG lengoaiaren hutsetik berregin dira.

Masterreko tesi honen beste zatia anotazio-eredu horri etekina ateraz Armiarma aplikazio linguistikoari funtzionalitate berriak gehitzean datza. Armiarma hizkuntzalariei corpusak aztertzen laguntzeko interfaze bat eskaintzen die, kontsultak definitzeko aukera emanez, ondoren, emaitzak modu txukun batean jasotzeko. Masterreko tesi-lan honetan berrikuntza bat diseinatu eta egin zaio Armiamari, eta kontsultak egiteko sistema hobetu

da.

Berrikuntza analisi linguistikoak eskuz sortzeko funtzionalitate bat izan da, beti ere, analisisien egitura kontuan hartuz analisisien sorkuntza intuitiboa eta erosoan izan dadin. Horretarako, aurreko puntuan esan bezala, analisi linguistikoek egitura definitu behar izan da XML eskema-lengoaia jakin bat, Relax NG, erabiliz. Ondoren, programa bat garatu da eskema horietatik abiatuz erabiltzaileari unean uneko informazioa eskatzeko. Hau da, programak, nolabait, badaki une bakoitzean zein informazio behar duen analisi zuzen bat sortzeko. Kontuan izan behar da analisi baten egitura oso konplexua dela, eta mota honetako laguntzarik gabe eskuz sortzea oso lan konplikatu izango litzatekeela. Gainera, behin hori eginda, analisisien egitura aldatzen bada ere ez da programan aldaketarik egin behar izango, nahikoa izango baita definizio-eskema bera eguneratzea.

Kontsulta-sistema ere bide beretik jarraituz hobetu da. Kontsultak egiteko galdetegia dinamikoa izango da orain, analisisien definizio-eskemetan oinarrituz beren edozein eremu edo ezaugarriaren araberrako kontsultak egin ahal izango baitira.

2 Aurrekariak

Ixa taldearen lan gehien oinarrian hizkuntzaren prozesaketa dago. Prozesatutako testuak, dagokion informazio linguistiko guztiarekin batera, proiektu gehienen abiapuntu izan ohi dira. Hori dela-eta, ezinbestekoa da prozesaketa horri arreta berezia eskaintzea, xehetasun guztiak zainduz, prozesu hori zuzena eta fidagarria izan dadin.

Hainbat dira prozesuan parte hartzen duten tresnak: Euskararen Datu-Base Lexikala (EDBL) (Aldezabal et al., 2001), tokenizatzailea, segmentatzailea, analizatzaile morfosintaktikoa (Alegria et al., 1996; Aranzabe et al., 2004), hitz anitzeko unitate lexikalak (HAUL) ezagutu eta prozesatzen dituen tresna (Alegria et al., 2004), eta lematizatzailea (Aduriz et al., 1996). Tresna horien arteko heterogeneotasuna dela-eta, ezinbestekoa da tresna horien sarrera-irteerak formalizatu eta eredu estandarizatu bat eskuan izatea, informazio linguistikoa prozesuan zehar tresna ezberdinen artean igaro dadin. Horixe da atal honetan azalduko dena hain zuzen, tresnen artean trukatu beharreko informazio linguistiko konplexu hori adierazteko Ixa taldean diseinatu den formatu estandarra zertan datzan. Horrekin batera, Ixa taldean fitxategietan antolatu ohi den eredu hori datu-baseetan nola antola daitekeen erakusten duen eredu ere azalduko da.

2.1 Anotazio linguistiko banatua: XML dokumentuen amarauna

Atal honetan, informazio linguistiko adierazteko sarreran aipatu den formatua azalduko da. Formatu hau eredu banatuari dagokio, informazioa dokumentu desberdinetan gordezten baita dagokion motaren arabera. Gainera, dokumentu horietako bakoitza XML dokumentu bat denez, XML dokumentuen amaraun bat eratzen da. Hortik datorkio, hain zuzen, bere izena: Anotazio linguistiko banatua: XML dokumentuen amarauna.

Atal honetan XML dokumentuen amaraun batean zein dokumentu mota aurki daitezkeen eta bakoitza zertan datzan azalduko da, eredu honen oinarrian dauden ezaugarri-egiturak azaltzetik hasiz.

2.1.1 Ezaugarri-egiturak

Informazio linguistiko oso konplexua izan daiteke. Gainera, lantzen ari garen informazio motaren arabera, oso desberdina ere bai. Adibidez, tokenizatzailearen eta analizatzaile morfosintaktikoaren irteerak oso desberdinak izango dira, informazio mota desberdina emango baitute biek. Hori dela-eta, XML dokumentuak oso egokiak dira informazio mota hau kodetzeko. Informazioa adierazteko XML egitura zehaztea, ordea, ez da berehalako lana. Lan horren konplexutasuna dela-eta, aurrez eginda dagoen lana aprobetxatzeko erabakia hartu zen, izan ere, TEI erakundeak (Ide eta Véronis, 1995; Sperberg-McQueen eta Burnard, 2002) informazio linguistiko kodetzeko ezaugarri-egitura izenekoak proposatzen

baititu (Carpenter, 1991). Horretaz baliatuz, hizkuntzaren prozesaketan parte hartzen duen tresna bakoitzaren sarrera-irteeretan erabiliko diren ezaugarri-egitura definitu ziren, Feature System Declaration (FSD) izeneko dokumentuetan, eta egitura horiek erabili ziren tresna guztiak integratzeko.

Adibide gisa, *azala* hitz-formaren lematizazioari dagokion XML dokumentua ikus daiteke jarraian:

```
1 <fs id="L-A-IZE-ARR-114" type="lematizazioa">
2   <f name="forma">
3     <str>azala</str>
4   </f>
5   <f name="lema-osatua">
6     <str>azal</str>
7   </f>
8   <!-- Ezaugarriak -->
9   <f name="ezaugarri-morfologikoak">
10    <fs type="goimailako-ezaugarri-lista">
11      <f name="KAT">
12        <sym value="IZE"/>
13      </f>
14      <f name="AZP">
15        <sym value="ARR"/>
16      </f>
17      <f name="BIZ">
18        <minus/>
19      </f>
20      <f name="KAS">
21        <sym value="ABS"/>
22      </f>
23      <f name="NUM">
24        <sym value="S"/>
25      </f>
26      <f name="MUG">
27        <sym value="M"/>
28      </f>
29    </fs>
30  </f>
31 </fs>
```

Ikus daitekeen bezala, hainbat ezaugarri-egitura (fs) ager daitezke analisi bakarrean, eta ezaugarri-egitura guztiak, ezaugarri (f) osatuta daude. Beraz, elementuen izenak ezaugarri motak bereizteko adierazgarriak ez direnez, type eta name atributuak gehitzen dira ezaugarri-egituren eta ezaugarrien motak adierazteko, hurrenez hurren.

2.1.2 Amarauneko dokumentuen egitura

Anotazio linguistikorako aukeratu den eredua anotazio linguistiko banatuarena da. Modu honetan, jatorrizko testuak ez du aldaketarik jasaten prozesatzen denean, aparteko dokumentuetan jasotzen baita testuari dagokion informazio guztia. Honela, dokumentuen amaraun bat osatzen da, non dokumentu bakoitzeko informazioa lotuta baitago bestetakoekin. Hori lortzeko, dokumentu batzuk informazio linguistikoa jasotzeaz arduratzen

diren moduan, beste batzuk jatorrizko testuko osagai bakoitza dagokion informazio linguistikoarekin lotzeaz arduratzen dira. Amaraunean aurki daitezkeen dokumentuen artean, honakoak daude:

- Jatorrizko testu-dokumentua.
- Jatorrizko dokumentuko osagaiak: dokumentu bat token bakunekin eta beste bat HAULekin¹.
- Analisi-bildumak: segmentazioak, analisi morfologikoak eta lematizazioak.
- Estekak: testuko osagai bakoitza dagozkion analisiekin erlazionatzeaz arduratzen direnak.

Bestalde, aipatu, dokumentu horiek guztiek TEI*k* proposaturiko egitura errespetatuko dutela. Batetik, informazio guztia <TEI.2> elementuaren barruan ezarriko da. Gainera, dokumentu guztiek buruko bat izango dute (<teiheader> elementua, hain zuzen), non dokumentuari buruzko informazio orokorra zehazten ahal baita. Azkenik, burukoaren ondoren, gainerako informazio guztia jasoko duen <text> elementu nagusia izango dute dokumentuek.

Jatorrizko testuak

Prozesatuko den jatorrizko testuari dagokion dokumentu mota da jarraian azalduko dena, hau da, jatorrizko testua XMLz nola kodetuko den, hain zuzen.

Testu hauek ahalik eta gutxien ukitu behar izatea da helburuetako bat, eta beraz, dagokien XML adierazpideak sinplea behar du izan. Egiten dena testua paragrafoetan banatzea da, XML dokumentuan testuaren paragrafoak <p> elementuaz adieraziz, beti ere, bakoitzak bere id atributuan identifikadore bat izanik ondoren bertako osagaiei erreferentziak behar bezala egin ahal izateko. Horrez gain, atalak eta izenburuak bezalako beste zenbait eremu ere jaso behar ditu adierazpideak. Atalak <div> elementuekin bereizten dira, eta izenburuak <head> elementuekin. Horrez gain, komatxoaren arteko elementuak, azpimarratutakoak, etab. zehazteko ere elementu bereziak erabiltzen dira.

Hona hemen adibide bat:

```
1 <TEI.2>
2 <teiheader> <!-- ... --> </teiheader>
3 <text>
4 <body>
5 <div id="idl" type="atala1">
```

¹Hitz anitzeko unitate lexikalak. Adibidez: kale-garbitzaile, laranja-zuku, hala ere...

```

6 <head id="id2">Lizarrako pilota jaialdia, zuzenean, ostiral honetan ETB-len</head>
7 <p id="id3"><hi rend="strong">Capellan-Lasa III eta Mikel Goni-Eulate arituko dira ↵
  ↳ estelarrean</hi></p>
8 <p id="id4">Eskuz binakako bi partida ikusteko aukera izango da nagusi ostiral gauean ↵
  ↳ ETB-len. Lizarrako jaialdiko bi partida emango dituzte 22:30etatik aurrera. ↵
  ↳ Lehenengoan Saralegi-Apraiz eta Cabrerizo II-Beroiz bikoteek neurtuko dituzte ↵
  ↳ indarrak. Ondoren, gaeko estelarrean, Capellan-Lasa III eta Mikel Goni-Eulate ↵
  ↳ arituko dira nor baino nor gehiago .</p>
9 <p id="id5">Xabier Euzkitzek eta Josean Tolosak egingo dute kontakizuna.</p>
10 <p id="id6"><hi rend="display: none;">&amp;nbsp;</hi></p>
11 </div>
12 </body>
13 </text>
14 </TEI.2>

```

Testu tokenizatua

Testu bat prozesatzearen lehenbiziko fasea, normalean, testua tokenizatzea izaten da. Fase honetan, testuan dauden osagaiak bereizten dira. Osagaiak, normalean, hitzak izango dira, baina ez beti. Bestelako tokenak ere badaude: komak, puntuak, kakotxak, etab. Gainera, askotan, token bakunez gain HAULak ere aurkituko dira.

Jatorrizko testuan identifikatutako tokenak biltzen dituen dokumentua da hau. Bertan bi zati nagusi bereizten dira, bi <p> elementutan banatuta: batetik, kanpo-erakusleak izango direnak. Hauek <xptr> elementuen bidez adierazten dira, eta token bat zein dokumentutan dagoen jakinarazten du. Bestetik, tokenak berak zein diren eta dokumentuaren zein posiziotan dauden adierazten duten elementuak daude, atributu baten bidez bakoitzari dagokion kanpo-erakuslearekin lotzen direlarik.

Adibidea:

```

1 <TEI.2>
2 <teiHeader> <!-- ... --> </teiHeader>
3 <text id="WDoc0001">
4 <body>
5 <p id="xptr">
6 <xptr id="Xw1" doc="TDoc0001"/>
7 <xptr id="Xw2" doc="TDoc0001"/>
8 <xptr id="Xw3" doc="TDoc0001"/>
9 <xptr id="Xw4" doc="TDoc0001"/>
10 <!-- ... -->
11 </p>
12 <p id="w">
13 <w id="w1" sameAs="Xw1" target="substring(//p[@id='id1']/text(),1,1)" type="PUNT_GALD" ↵
  ↳ >?</w>
14 <w id="w2" sameAs="Xw2" target="substring(//head[@id='id3']/text(),1,4)" type="HAS_MAI ↵
  ↳ >Aupa</w>
15 <w id="w3" sameAs="Xw3" target="substring(//head[@id='id3']/text(),6,2)">gu</w>
16 <w id="w4" sameAs="Xw4" target="substring(//head[@id='id3']/text(),9,3)">eta</w>
17 </p>
18 </body>
19 </text>
20 </TEI.2>

```

HAULen egitura

Testu tokenizatuari dagokion dokumentuan token bakunak bereizten dira. Hala ere, testuetan HAULak ere ager daitezke. Horiek dokumentu honen bidez adieraziko dira. Dokumentu honetan estekak soilik agertuko dira. Honela, HAUL baten osagai diren tokenen erreferentziak bilduko dira dokumentu honetan.

Horretarako, batetik, testuko HAULen parte diren tokenen erreferentziak biltzen dira `<xptr>` elementuen bidez, eta ondoren, `<join>` elementuen barruan, HAULa osatzen duten tokenen erreferentziak.

Adibidea:

```

1 <TEI.2>
2 <teiHeader> <!-- ... --> </teiHeader>
3 <text id="MWDoc0001">
4   <body>
5     <p id="xptr">
6       <xptr id="Xw87" doc="WDoc0001" from="ID(w87)"/>
7       <xptr id="Xw88" doc="WDoc0001" from="ID(w88)"/>
8       <xptr id="Xw119" doc="WDoc0001" from="ID(w119)"/>
9       <xptr id="Xw120" doc="WDoc0001" from="ID(w120)"/>
10    </p>
11    <p id="joinGrp">
12      <joinGrp type="Haul" targOrder="y">
13        <join id="mw1" type="Correct">
14          <target name="wId" value="Xw87"/>
15          <target name="wId" value="Xw88"/>
16        </join>
17        <join id="mw2" type="Correct">
18          <target name="wId" value="Xw119"/>
19          <target name="wId" value="Xw120"/>
20        </join>
21      </joinGrp>
22    </p>
23  </body>
24 </text>
25 </TEI.2>

```

Segmentazio mailako informazio linguistikoa

Dokumentu hau segmentatzaile morfologikoaren irteera da. Besterik gabe, ezaugarri-egitura bakoitzak segmentazio bat adierazten du, eta `<p>` elementu baten barruan kokatzen da.

Segmentazio mailako analisi bakar batez osatutako dokumentuaren adibidea, *lor*tu formari dagokiona:

```

1 <TEI.2>
2 <teiHeader> <!-- ... --> </teiHeader>

```

HAP masterra

```
3 <text id="MWDoc0001">
4   <body>
5     <p>
6       <fs id="S-A-ADI-SIN-9" type="segmentazioa">
7         <!-- ... -->
8       </fs>
9     </p>
10  </body>
11 </text>
12 </TEI.2>
```

Analisien gorputzaren egitura aurrerago aztertuko da xehetasunez, 3.4 atalean.

Morfologia mailako informazio linguistikoa

Dokumentu hau analisi morfosintaktikoaren irteera da. Segmentazioaren kasuan gertatzen zen moduan, <p> elementu baten barruan doa analisi bakoitza, ezaugarri-egitura batekin adierazita. Analisi morfosintaktiko honek, segmentazioa bere barruan adierazteaz gain, ondoko prozesuetarako informaziorik garatuena ematen du.

Dokumentuen egitura segmentaziokoen berbera da, baina analisiak morfosintaxi mailakoak izanik.

Lematizazio mailako informazio linguistikoa

Lematizatzailearen irteera da dokumentu hau. Bertan, analisi morfosintaktikoetatik abiatuta, desanbiguazio-lanak egin (Ezeiza et al., 1998) eta bertsio sinplifikatu bat itzultzen du. Hori adierazteko, lematizazio-analisi bakoitza ezaugarri-egitura batekin adierazten da, <p> elementu baten barruan.

Segmentazio- eta morfosintaxi-analisen dokumentuen egitura berbera dute lematizazioko analisen dokumentuek ere.

Tokenen eta dagokien informazio linguistikoaren arteko estekak

Tokenak eta analisiak biltzen dituzten dokumentuak nola antolatzen diren ikusi dugu orain arte, baina dokumentu horietan bai tokenak eta bai analisiak independenteak dira. Hau da, ez dago loturarik testuko tokenen eta analisen artean. Anotazio-eredua egokia izatekotan, token bakoitzari dagozkion analisiak zein diren adierazi beharra dago nolabait, eta bai alderantziz ere, analisi bakoitza zein tokeni dagokion. Izan ere, anbigutasuna dago hemen. Token bakoitzak analisi posible bat baino gehiago izan ditzake, hitz batek, adibidez, aditz edo izen moduan joka baitezake testuinguruaren arabera.

Tokenak eta dagozkien analisiak lotzeko, beraz, lotura-dokumentuak sortzen dira. Do-

kumentu hauetan tokenen erreferentziak eta dagozkien analisisen erreferentziak elkartzen dira. Honela, token batek erreferentzia bat baino gehiago baditu, anbigua dela esaten da. Desanbigua egiten denean, token bakoitza analisi soil batekin lotuta utzi behar da, hau da, dagozkion analisi guztietatik bakarra utzi behar da zuzen bezala markatuta.

Mota honetako hiru dokumentu sortzen dira: segmentazio mailako analisientzat, morfosintaxi mailakoentzat eta lematizazioentzat bana. Hirurek formatu berbera dute. Batetik, <xptr> elementuen bidez tokenen eta segmentazio, morfosintaxi edo lematizazio mailako analisisen erreferentziak eskuratzen dira. Ondoren, erreferentzia horietaz baliatuz, <linkgrp> elementuaren barruan, token bakoitza dagozkion analisiekin lotzen da. Horretarako <link> elementuak erabiltzen dira, targets atributuan token baten eta analisi baten erreferentziak zehazten baitira. Gainera, type atributuan esteka zuzena den edo ez adierazten da, hau da, testuinguru horretan tokenari analisi jakin bat dagokion edo ez.

Segmentazio, morfosintaxi eta lematizazioari dagozkien esteken dokumentuek egitura aldetik desberdintasunik ez dutenez, adibide bakarra jarri da, lematizazioari dagokiona.

```

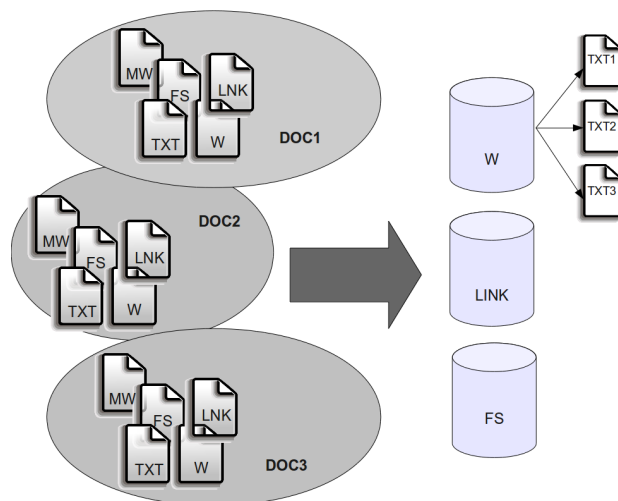
1 <TEI.2>
2 <teiHeader> <!-- ... --> </teiHeader>
3 <text id="LNKDoc0004">
4 <body>
5 <p id="xptr">
6 <!-- Tokenen erreferentziak -->
7 <xptr id="Xw2" doc="WDoc0001" from="ID(w2)"/>
8 <xptr id="Xw3" doc="WDoc0001" from="ID(w3)"/>
9 <xptr id="Xw4" doc="WDoc0001" from="ID(w4)"/>
10 <!-- ... -->
11
12 <!-- Lematizazioen erreferentziak -->
13 <xptr id="XL-A-ADI-SIN-207" doc="LemDoc0002" from="ID(L-A-ADI-SIN-207)"/>
14 <xptr id="XL-A-ITJ-6" doc="LemDoc0002" from="ID(L-A-ITJ-6)"/>
15 <xptr id="XL-A-IOR-PERARR-7" doc="LemDoc0002" from="ID(L-A-IOR-PERARR-7)"/>
16 <xptr id="XL-A-LOT-JNT-8" doc="LemDoc0002" from="ID(L-A-LOT-JNT-8)"/>
17 <xptr id="XL-A-LOT-MEN-3" doc="LemDoc0002" from="ID(L-A-LOT-MEN-3)"/>
18 <!-- ... -->
19 </p>
20
21 <p id="linkGrp">
22 <linkGrp type="Token-Analisi" targOrder="y">
23 <link type="NoCorrect" targets="Xw2 XL-A-ADI-SIN-207"/>
24 <link type="Correct" targets="Xw2 XL-A-ITJ-6"/>
25 <link type="Correct" targets="Xw3 XL-A-IOR-PERARR-7"/>
26 <link type="Correct" targets="Xw4 XL-A-LOT-JNT-8"/>
27 <link type="NoCorrect" targets="Xw4 XL-A-LOT-MEN-3"/>
28 </linkGrp>
29 </p>
30 </body>
31 </text>
32 </TEI.2>

```

2.1.3 Amaraunaren implementazio bat: informazioa datu-baseetan antolatzen

Orain arte azaldu dugun amarauna eredu teoriko bat besterik ez da, informazioa nola egituratu azaltzen duena. Eredu horrek implementazio desberdinak izan ditzake, adibidez, fitxategi edo datu-baseetan. Ixa taldean, normalean, fitxategitan antolatzen da. Atal honetan azalduko dugun implementazioa 2.2 atalean azalduko den Armiarma aplikazioarekin lotuta dago, aplikazio horretan erabiltzeko diseinatu baitzen. Bertan, informazioa datu-baseetan antolatzen da.

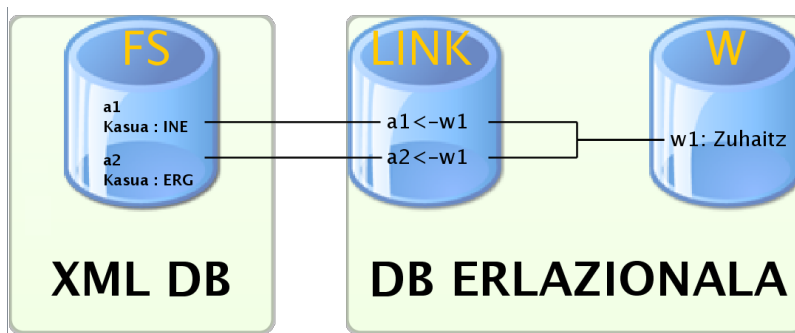
Amaraunaren implementazio honek zenbait hobekuntza eskaintzen ditu fitxategietan oinarritzen denarekin alderatuta. Batetik, atzipenak eta aldaketak egitea erosoagoa eta eraginkorragoa da. Baina, garrantzitsuena, fitxategien ereduari agertzen den errepikatutako informazio ugaria desagertzen da. Corpusak dokumentuz dokumentu prozesatzen dira, eta prozesu horien irteerak fitxategietan gordetzen dira. Dokumentuen prozesamenduak independenteak direnez elkarrekiko, ez da kontuan hartzen une bakoitzean sortzen den irteera agian aurrez prozesatutako dokumentu batean dagoeneko adierazita dagoela. Hori dela eta, corpuseko testu-dokumentuetan errepikatzen diren elementu guztiak, eta horiei dagokien informazio linguistiko guztia, agerpen adina aldiz errepikatuta agertzen da fitxategietan. Datu-baseetan, ordea, arazo hori desagertu egiten da, izan ere, token, HAUL, analisi edo estekem biltegi bakoitza corpus mailakoa baita, eta ez dokumentu mailakoa (1. irudia).



1 irudia: Amarauna fitxategien eredutik datu-baseen eredura pasatzearen abantailak.

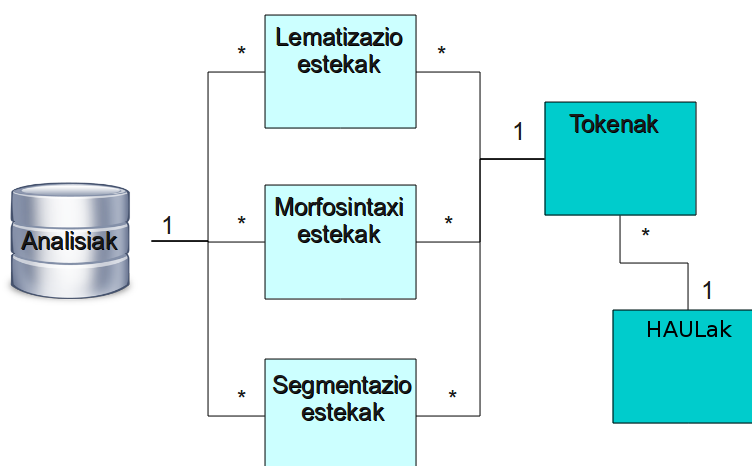
Amaraun-ereduan datu mota desberdinak ditugu: tokenak, HAULak, analisi linguistikoak eta estekak. Hori dela-eta, bakoitzarentzat datu-base eta taula egokiak diseinatu behar dira. Kasu honetan, bi datu multzo desberdin sailkatu dira beren izaeraren arabera: analisiak batetik, eta tokenak, HAULak eta estekak bestetik. Tokenak, HAULak eta estekak oso egokiak dira datu-base erlazionaletan gordetzeko. Izan ere, hauen egitura sinplea baita, atributu zerrenda batekin adierazten ahal baitira. Horregatik, MySQL

motako datu-base erlazionalak erabiltzen dira horiek gordetzeko. Analisisien egitura, aldiz, oso desberdina da, zuhaitz baten egitura baitaude. Mota horretako informazioa gordetzeko ohiko datu-base erlazionalak ez dira egokiak. Aldiz, XML datu-baseak aukeratu dira analisiak biltegitratzeko. Zehazki esanda, Berkeley XML DB datu-basea.



2 irudia: Amarauna datu-baseetan antolatuta.

2. irudian ikusten den bezala, hiru multzotan banatu da informazioa: tokenak, HAU-Lak eta estekak datu-base erlazional baten tauletan, eta analisiak XML datu-base batean. Hiru analisi-maila desberdin direla-eta, analisiak gordetzeko hiru XML datu-base erabiltzen dira, berdinak izan arren, analisi mota bakoitza besteengandik bereizita mantentzeko. Ondorioz, eta eredu kontsistente mantentzearen, estekak ere hirutan banatu dira. Hone-l, estekentzat egitura bereko hiru taula erlazional erabiltzen dira. Eskema osoa 3. irudian ikus daiteke.



3 irudia: Datu-baseen eskema.

XML datu-basearen barne-egiturak ez du misteriorik. Mota horretako datu-baseak dokumentutan banatzen dira, XML dokumentutan, dokumentu bakoitzak identifikadore

bat daukalarik. Kasu honetan, analisi linguistiko bakoitza datu-baseko dokumentu batean biltegitratzen da, analisiaren identifikadorea bera dokumentuaren identifikadore izanik.

Token, HAUL eta esteken taulak konplexuagoak dira, elementu horiek atributu zerrenda baten bidez adierazi behar baitira. Eredu garrantzitsuenak aipatuz, tokenek dagokien dokumentuaren izena, dokumentu mailako identifikadore bat eta tokenaren forma bera izan behar dute. Horrez gain, tokenak dokumentuan duen posizioa, paragrafoaren identifikadore, token mota² eta tokena XML dokumentuan kokatzen duen XPath espresioa ere gortetzen dira taulan. Taula mailako identifikadoreak dokumentu-izenak, identifikadoreak eta posizioak osatzen dute. HAULen taulak atributu gutxiago ditu, izan ere, HAUL bat hainbat tokenen elkarketa baita. Beraz, HAULaren identifikadoreak, tokenaren dokumentu-izenak eta tokenaren dokumentu mailako identifikadoreak osatzen dute HAULen taulako sarrera bakoitzaren identifikadorea. Horrez gain, HAULA zuzena edo okerra den adierazten duen eremu bat dauka taulak. Azkenik, esteken taulan, tokenaren identifikadoreak eta dagokion analisiarenak osatuko dute esteken identifikadorea: dokumentu-izenak, tokenaren dokumentu mailako identifikadoreak eta analisiaren identifikadoreak. Horrez gain, token horrentzat analisi jakin hori zuzena den edo ez adierazten duen boolearra ere badauka.

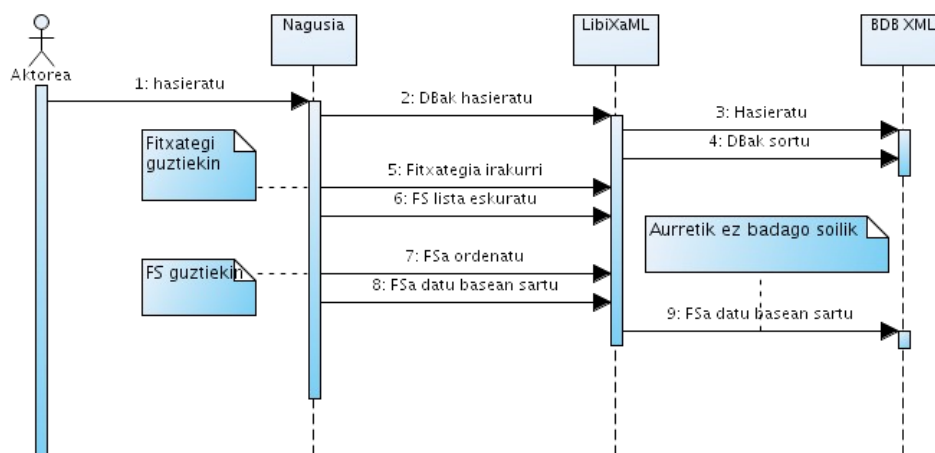
2.1.4 Datu-baseen elikatzea

Aurreko atalean ikusi dugu amaraunaren implementazio bat, 2.2 atalean aztertuko den Armiarma aplikaziorako diseinatu zena. Implementazio hori datu-baseetan oinarritzen da, eta ikusi dugu datu-baseak eta taulak nola antolatzen diren. Jatorriz, ordea, Ixa taldean informazio hori fitxategietan mantentzen da, eta beraz, datu-baseak elikatu egin behar dira fitxategi horien edukiarekin.

Prozesu horretan laguntza handia eskaintzen du LibiXaML (Artola et al., 2002) liburutegiak. Liburutegi hau Ixa taldean garatu zen, amarauneko informazioarekin lan egiten laguntzeko. Oraingoz, fitxategietan antolatutako edukiekin bakarrik egiten du lan, baina etorkizunean datu-baseetan antolatutako ereduak ere integratzea da asmoa. Dena dela, kasu honetan, fitxategietatik jasoko da datu-baseetan sartuko den informazioa, eta beraz, laguntza handia eskaintzen du liburutegiak token, HAUL, analisi eta estekak fitxategietatik irakurtzeko garaietan. LibiXaMLk klase asko eskaintzen ditu, baina datu-baseak elikatzeko prozesuan `W`, `Join`, `FS` eta `Link` dira interesgarrienak, token, HAUL, analisi eta estekak maneiatzeko erabiltzen direnak baitira, hurrenez hurren.

Prozesua antzekoa da lau elementu motentzat. Lehenik, datu-baseko taula sortzen da (dagoeneko sortuta ez badago). Ondoren, fitxategiaren edukia kargatzen da. Horretarako ere LibiXaMLk zenbait klase eskaintzen ditu: `WXMLDoc`, `MwXMLDoc`, `FSXMLDoc` eta `LnkXMLDoc`. Behin edukia objektu batean daukagula, objektu horiek berek eskaintzen

²Token mota desberdinak bereizten dira amaraun-ereduan, hala nola, puntuazio-ikurrak (puntu, koma, galdera ikur...), letra-larriz hasten direnak etab.



4 irudia: Analisisien (FS) biltegitzearen sekuentzia diagrama.

duten metodo batekin token, HAUL, analisi edo esteka zerrenda bat lortzen da. Azkenik, elementuak banan-banan datu-basean sartzen dira, objektu horiek elementuaren atributu guztiak eskuratzeko metodoak eskaintzen baitituzte. Elementua aurrez datu-basean bazegoen edo ez ere aztertzen da. 4. irudian analisisien biltegitze-prozesuaren sekuentzia-diagrama ikus daiteke. Gainontzekoek ere egitura berdina dute.

2.2 Armiarma: anotazio linguistiko konplexuak kudeatzeko web-aplikazioa

Aplikazio hau Ixa taldean garatu da, Lexikoaren Behatokia proiektuaren baitan. Proiektu hau Euskaltzaindiari dagokio eta UZEI, Elhuyar eta Ixa taldearekin batera garatzen ari da. Helburua hizkuntzaren ikerketarako eta azterketarako testu-corpus handi bat osatzea du, horretarako Argia, Berria, EITB.com eta beste zenbait komunikabideren eskutik lortutako testuak erabiliz. Corpus horrek interfaze bat behar du, kontsultak ez ezik, desanbiguazio-eta aberaste-lanak ere egiteko modua izateko. Armiarma da aplikazio horren izena, eta aurrez Ixa taldean garatutako EULIA (Artola et al., 2004) tresnaren oinordekoa da. Masterra proiektu honek lotura handia dauka Armiarmarekin, izan ere, hemen egindako lanak aplikazio hori garatu eta osatzeari begira egin baitira neurri handi batean.

2.2.1 Interfaze grafikoa

Armiarmaren web-interfazeak aukera eman behar du testu-corpus baten gainean kontsultak egin, analisisien desanbiguazio-lanak egin eta analisi berriak eskuz sortu eta corpusean txertatzeko. Hori dena, erabiltzaile eta baimenen kudeaketa bat ginez, noski.

HAP masterra

The screenshot shows the ARMIARMA web interface. At the top left is the logo 'ixa' and the title 'ARMIARMA'. Below the logo is a 'Hasiera' button. On the right, there are input fields for 'Izena:' and 'Pasahitza:' with a 'Sartu' button. The main content area is divided into three vertical panels on the left, a central results panel, and a right analysis panel.

Left Panel (Filters):

- Datu-basea:** lem
- Analisiak:** Denak
- Hitza 1:** Mota: Lema, Hitza: beakazkeria
- Hitza 2:** Gaitu (checked), Distantzia: 2, edonon, Mota: Forma, Hitza: eta
- Hitza 3:** Gaitu (unchecked), Distantzia: 2, edonon, Mota: Lema

Emaitzak (6):

gauza baña . Harrokeria eta beakazkeriaz , abokoa baino gehiago izate (Testu osoa)
 Taldearen izenean . Harrokeria eta beakazkeria dira bekatu bi hauek oso (Testu osoa)
 bertso berria . Pasioa , beakazkeria eta traizioa ditu hizpide Emilio (Testu osoa)
 bertso berria . Pasioa , beakazkeria eta traizioa ditu hizpide Emilio (Testu osoa)
 eta aferagile harraparik pairamena , beakazkeria eta desjabetze ekonomikoa zuzentasunaren izenean (Testu osoa)
 erre zituzten , askotan inguruko beakazkeria eta mendekuen ondorioz . Sorgin (Testu osoa)

Analisiak:

beakazkeria eta
 beakazkeria
 L-A-IZE-ARR-2971571
 Forma = beakazkeria
 Lema = beakazkeria
 AZP = ARR
 BIZ =
 KAS = ABS
 KAT = IZE
 MUG = M
 NUM = S

Correct
 L-A-IZE-ARR-2971570
 L-A-IZE-ARR-2971682

5 irudia: Armiarmaren kontsulta-interfazea.

Kontsulta-sistema

Kontsultena da Armiarmaren oinarriko funtzionalitatea (5. irudia). Erabiltzaileak hainbat terminoren arabera kontsulta bat defini dezake, aplikazioak corpusean bilaketa egin eta emaitzak KWIC³ moduan itzuliko dituelarik. Kontsultak definitzeko formulario bat eskaintzen du interfazeak ezkerreko aldean, lau zatitan bananduta. Lehenbiziko zatian bi parametro aukeratu daitezke: segmentazio, morfosintaxi edo lematizazio mailan lan egin nahi dugun, eta analisi guztiak edo zuzenak bakarrik kontuan hartzea nahi dugun. Beste hiru zatietako bakoitzean termino bat definitzen ahal da. Termino bat definitzeko, forma edo lemaz gain, kategoria, azpikategoria eta beste edozein ezaugarri ere zehazteko aukera dauka erabiltzaileak. Azkenik, definitutako termino bakoitzaren kokapenaren inguruko murriztapenak gehitu daitezke, beti ere, aurrez definitutako terminoarekiko erlatiboak. Hau da, bigarren terminoak lehenbizikoaren ondoren agertu behar duela esan dezakegu, asko jota hiru hitzeko distantziara; edo lehenbizikoaren hurrengo edo aurreko hitza izan behar duela, etab. Baldintza guztiak konjuntzio modukoak dira, eta, beraz, baldintzetako bat betetzen ez bada, agerpena baztertu egiten da zuzenean.

³KWIC formatuak, (Key Word In Context, ingelesez) hitz gako bat edo gehiago testuinguruan kokatuta erakusten ditu. Kasu honetan, kontsultan definitutako terminoa erdigunean kokatuko du, inguruan hitz horren aurreko eta ondorengo hitzak ere erakutsiz.

ixa **ARMIARMA**

Hasiera Dokumentua X

Izena: zbelok001 Pasahitza: ●●●●●● Sartu

BIRA. Plazera

Egilea: BORDA, Itxaro

Bira
Itxaro Borda
Plazera

Krisia da. Aburu ikerketen arabera familiek egubertrako ohiko zazpi ehun euroen ordean, soilik lau ehunen xahutzeko xedea dagarte: sakelak zulo dabilzak eta gerrikoak trinko. Errealitate mingotsa da denda hutsak eta kontsumitzeko kemen ahulak sumatzen. Baina ele zuriz harpatu pantailatan ardura aditzen den perpausa behar diogu elkarrizketa plazer egin teherria dugu, magazin handietako atal karioak erakusten dizkiguten bitartean.

Santizpitu zubitik etxera nentoren ilunbean gogoeta hasi nintzen plazera kontzeptua ardatz. Funtsean zergatik hainbeste errepikatzen da hitz hori? Gobernuan jarri dituzten arazo gaztigatzaileak, haren pulinda mukiak eta aferagile harrapariak palramena, **bekezkeria** eta desjabetze ekonomikoaren zuzentasunaren izenean justifikatzen dute. Herritarrak sufritzen du, lepo zainak tiran, beti bektoros, eta halaz, noibait onargarria da pentsatzea noiztenka plazer txikiak zilegi zaizkiola ez dagoen dirutik baldin bada ere.

Ideologikoki buruak apal herrestan bizitzearen auto-saria al da kontsumitzeko plazera? Aita zenak lansario gogorren ondorengo beti erraren zuen: ano tllikaat mehexitiatan ba'araziz...

Plazer horren ukatzea oldar-ekintza bat balitz?

6 irudia: Agerpen bati dagokion testu osoa ikusteko estekan klikatuz, erlantz berri bat zabaltzen da testua erakutsiz.

Kontsulta batek eskaintzen duen informazioa ez da hor bukatzen, ordea. Emaitzetako agerpen bakoitzean, kontsultan definitu diren termino bakoitzaren analisi linguistikoak ere ikus daitezke horietakoren batean klikatuz gero. Analiak eskuineko aldean agertuko dira, testuinguruaren arabera zuzen bezala markatuta dagoena berdez, eta okerrekoak gorritz.

Kontsultatu daitekeen beste gauza bat agerpenari dagokion dokumentu osoa da. Kasu honetan, dokumentu bakoitza komunikabide bateko artikulua bat da normalean. Artikulu osoa ikusteko, bere egile, izenburu, data eta testu osoarekin, agerpenaren eskuin aldean “testu osoa” jartzen duen estekan klikatu behar da. Honela, erlantz berri bat zabalduko da interfazean artikulua erakutsiz (6. irudia).

Desanbiguazio-sistema

Lexikoaren Behatokia corpusa linguistikoki prozesatu eta anotatu da. Hala ere, prozesu hori automatikoki egin da bere osotasunean. Horrek esan nahi du zenbait kasutan informazio linguistikoa ez dela zuzena izango. Hori dela-eta, interfazeak analisisien eskuzko desanbiguazioa egiteko funtzionalitatea eskaintzen du. Erabiltzaileak, horretarako, baime-nak esleituta izan behar ditu sisteman. Prozesua sinplea da, hitz baten analisiak erakustea

HAP masterra

The screenshot shows the Armiarma web application interface. At the top left is the 'ixa' logo and the title 'ARMIARMA'. Below it are navigation tabs for 'Hasiera' and 'Dokumentua'. On the right, it says 'Kaixo zbeloki (Itzen)'. The main interface is divided into three main sections:

- Left Sidebar (Filters):**
 - Datu-basea:** Dropdown menu set to 'lem'.
 - Analisiak:** Dropdown menu set to 'Denak'.
 - Hitza 1:** 'Mota' dropdown set to 'Lema', 'Hitza' input field containing 'bekaizeria', and a link for 'Informazio morfologikoa'.
 - Hitza 2:** 'Gaitu' checkbox checked, 'Distantzia' dropdown set to '2', 'edonon' dropdown, 'Mota' dropdown set to 'Forma', 'Hitza' input field containing 'eta', and a link for 'Informazio morfologikoa'.
 - Hitza 3:** 'Gaitu' checkbox unchecked, 'Distantzia' dropdown set to '2', 'edonon' dropdown, 'Mota' dropdown set to 'Lema'.
- Central Area (Emaitzak (6)):** A list of search results with words and their frequency in parentheses:
 - gauza baita . Harrokeria eta bekaizeriaz , alboko bano gehiago izate (Testu osoa)
 - Taldearen izenean . Harrokeria eta bekaizeria Dira bekatu bi hauek oso (Testu osoa)
 - bertsio berria . Pasioa , bekaizeria eta traizioa ditu hizpide Emilio (Testu osoa)
 - bertsio berria . Pasioa , bekaizeria eta traizioa ditu hizpide Emilio (Testu osoa)
 - eta aferagile harrapariak pairamena , bekaizeria eta desjabetze ekonomikoa zuzentasunaren izenean (Testu osoa)
 - erre zizutzen , askotan ingurukoan bekaizeria eta mendekuen ondorioz . Sorgin (Testu osoa)
- Right Sidebar (Analisiak):**
 - Search input fields containing 'bekaizeria' and 'eta'.
 - Dropdown menu set to 'eta'.
 - Analysis types: 'L-A-LOT-MEN-1' (with a green dot) and 'L-A-LOT-JNT-1' (with a red dot).
 - Buttons: 'Gorde' (Save) and 'Garbitu' (Clear).

7 irudia: Analisisien desanbiguazioa Armiarman.

eskatzen denean, aurreko atalean ikusi den moduan, berdez analisi zuzena agertzen da eta gorritz okerrak. Erabiltzaileak desanbiguazioa gaizki eginda dagoela ikusten badu, nahikoa du koloreen gainean klikatuz kolore horiek aldatzea. Aldaketak baieztatzeko botoia sakatzen den unean, datu-basean aldaketak gordeta gelditzen dira (7. irudia).

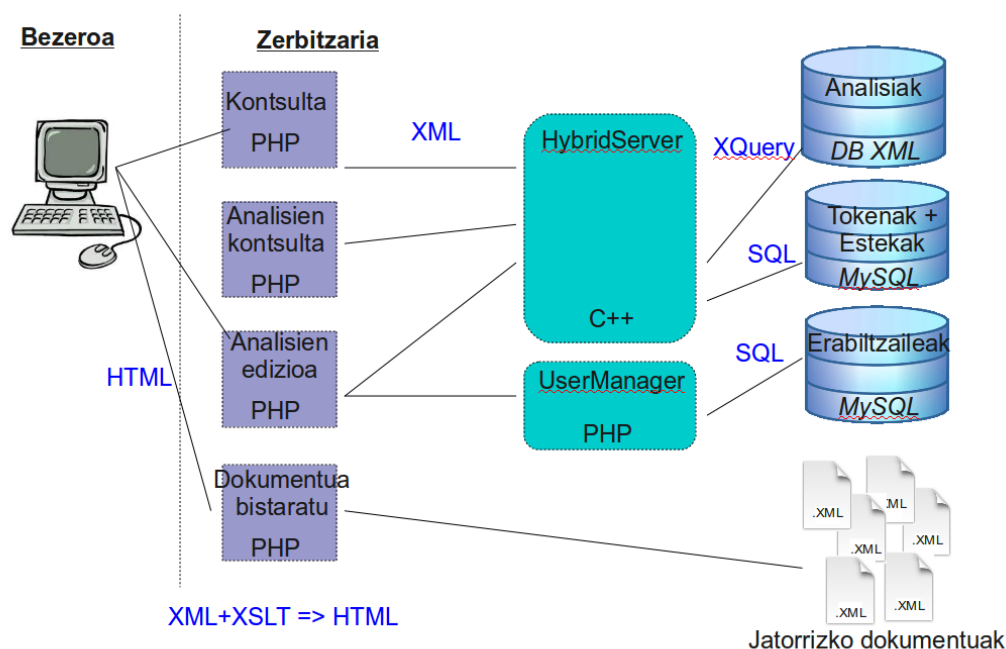
2.2.2 Sistemaren arkitektura

Armiarma aplikazioaren arkitekturak badauka konplexutasunik (8. irudia). Web-aplikazio batez ari gara. Arkitektura hiru mailatan bereiz dezakegu: datuen modeloa (datu-baseak eta fitxategiak), aplikazioaren logika eta interfazea.

Datuen modeloa aurkeztu dugu dagoeneko, 2.1.3 atalean. Laburki esanda, analisiak XML datu-baseetan gordetzen dira, eta token, HAUL eta estekak datu-base erlazionalan. Aplikazioarentzat espezifiko den beste datu-base erlazional bat ere badago, erabiltzaileen informazioa gordetzeaz arduratzen dena. Horrez gain, jatorrizko dokumentuak fitxategietan mantentzen dira, dokumentu osoa interfazean erakusteko garaian praktikoagoa baita fitxategietatik jasotzea datu-baseetatik dokumentua berreraikitzea baino.

Aplikazioaren logika bi modulu nagusik kudeatzen dute. Batetik, PHPz idatzitako

HAP masterra



8 irudia: Armiarmaren arkitektura.

UserManager moduluak, erabiltzaile eta baimenen kudeaketaz arduratzen dena. Hau da erabiltzaileen datu-basearekin komunikatuta dagoen modulu bakarra. Bestetik, aplikazioaren modurik garrantzitsu eta konplexuena den HybridServer moduluak daukagu, kontsultak eta desanbiguzioak egiteaz arduratzen dena. Modulu hau C++ lengoia idatzita dago eta analisen, tokenen, HAULen eta esteken datu-baseekin komunikatuta dagoena da. Azkenik, kontsulta- edo desanbiguzio- eskaera bakoitzeko HybridServer-en instantzia bat sortzeaz PHP scriptak arduratuko dira. Hau da, web-orriak PHPz kodetuta daudenez, eskaera bakoitzeko HybridServer moduluaren instantzia bat sortuko du, dagozkion parametroak pasatuz, eta moduluak itzuliko dizkion emaitzak jasotzeko komunikazio-kanal bat sortuz. HybridServer moduluak bere emaitza guztiak XMLz kodetuta itzultzen ditu.

Amaitzeko interfazea azaltzea gelditzen da. Aplikazioaren izaera dinamikoa dela-eta, interfazea leiho bakarrean erakusteko ahalegina egin da: kontsultaren definizioa, emaitzak, analisiak eta desanbiguzioa, dena, leiho bakarrean. Horretarako AJAX teknika erabili da, honela ez baita orri osoa birkargatu behar izaten zerbitzariarekin komunikatu nahi den bakoitzean. Bestalde, esan dugun moduan zerbitzariak emaitza guztiak XMLz itzultzen dituzenez, XSLT scriptak sortu dira XML egitura horiek HTML bihurtu eta interfazean modu txukun batean erakusteko.

3 Analisi linguistikoaren formalizazioa

3.1 Motibazioa: RELAX NG eskema eta erabilera

Ixa taldean analisi linguistikoak aspalditik sortu eta erabili izan dira. Analisiak egitura konplexuak dira, hierarkikoki antolatutako eremu desberdinez osatzen baitira. Gainera, analisi batetik bestera asko alda daiteke analisiaren itxura. Adibidez, kategoria desberdinetako hitzen analisiak ezaugarri desberdinak izango dituzte. Esaterako, aditz batek aditzoina definituta izan behar duen bezala, izen batek plurala den edota biziduna den bezalako ezaugarriak izan behar ditu definituta. Aldagarritasun horrek buruhausteak eman ditzake analisi baten egitura orokorra nolakoa den ulertzeko garaian, eta beraz, ezinbestekoa da, nolabait, analisisien egitura orokor hori formalki definitzea.

Hasieratik definitu zen analisisien egitura nolakoa zen. Definizio horiek egiteko Feature System Declaration (FSD) lengoia erabili zen (aldi berean XML lengoia bat dena). Lengoaia hau ez zen hasiera batean espero zena bezain erabilgarria suertatu. Horren arrazoi nagusietako bat FSDekin definitutako lengoian idatzitako dokumentuak balidatu ezin izatea da, ez baitago FSDekin lan egiten duen parser edo balidatzaileirik. Analisiak denboran zehar aldatu egin dira, hamaika arrazoi direla medio, eta, aldiz, definizioak (FSDak) beren horretan geratu dira, eguneratu gabe, eta pixkanaka zaharkituz joan dira. Aldaketa guztiak definizio-eskemarekin balidatu behar izanez gero, errazagoa litzateke eskema horiek eguneratuta mantentzea. Arazoa beste ikuspuntu batetik ikusita, FSD eskema gizakiak analisisien egitura ulertzeko lagungarri dira, baina ez dute balio konputagailuek egitura hori uler dezaten, esan bezala, ez baitago horiekin lan egiteko parser edo APIrik.

Urteak pasatu dira, eta FSD definizioak oso atzeratuta gelditu dira analisisien gaur egungo egiturarekin alderatuta, baina ez da hori izan eskema birdefinitzera bultzatu gaituen arrazoi bakarra. Armiarma aplikazioaren funtzionalitate berri bat analisi linguistikoak hizkuntzalari batek eskuz sortzeko laguntza ematea da. Horretarako, sistemak gidatu behar du hizkuntzalaria, une bakoitzean hautagai diren eremuak proposatuz, eta eremu bakoitzaren balio posibleak zein diren jakinaraziz, besteak beste. Hori egiteko, garbi dago sistemak analisisien egitura xehetasun guztiekin ezagutu behar duela. Hori izan da analisi linguistikoak RELAX NG (RNG) lengoiaz definitzeko arrazoi nagusia. RNG XML eskema-lengoia bat da, XML lengoia berriak definitzea ahalbidetzen duena. TEIk dokumentuak balidatzeko hobesten duen lengoia da, izatez. Kasu honetan, analisi linguistikoaren egitura definitu da RNG eskema bidez. Honela, aurrez aipatu dugun analisisien sorkuntza egiten laguntzeko sistemak nahikoa izango du RNG eskema irakurtzearekin analisisien egitura zehatza ezagutzeko. Gainera, RNG eskema balidatzeko tresna ugari daude, XML dokumentu bat eta RNG eskema bat emanda, dokumentu hori eskema horretan definitutako lengoiaarekin bat datorren edo ez esaten duenak.

3.2 Analisisien XML egitura definituko duten RNG eskemen sorkuntza automatikoa analisi multzo batetik abiatuta

Analisien RNG definizio-eskemak automatikoki sortu direla esan daiteke. Horretarako, analisi multzo batetik abiatuta, beren egitura erauzi da programa baten bidez. Atal honetan ataza hau nola burutu den azalduko da, betekizunen analisia egitetik hasi eta algoritmoa azaltzeraino.

3.2.1 Betekizunen analisia

Analisien egitura erauztea beren XML dokumentuetatik abiatuta RNG eskema lortzean datza. Hori egiteko tresnak egon badaude, baina ez dira erabilgarriak izan kasu honetan. Horren arrazoia tresna horiek oso helburu orokorrekoak izatea da, eta aplikazio honek badu ezaugarri bat helburu orokorreko tresna horiek jokoaz kanpo uzten dituen. Aipatutako tresnek bi XML elementu berdinak direla jotzen dute elementuen izena berdina bada. Analiak kodetzeko erabili den TEIk proposatutako ezaugarri-egituren izaera dela-eta, ordea, elementu gehien izena f s edo f da, eta elementuak `type` eta `name` atributuen balioen arabera desberdintzen dira. Arazoa hobeki ikusteko, adibide bat jarriko da:

```
1 <ezaugarriak>
2   <f name="KAT">
3     <sym value="IZE"/>
4   </f>
5   <f name="AZP">
6     <sym value="ARR"/>
7   </f>
8 </ezaugarriak>
```

XML dokumentu horretan ageri diren bi elementuak analisi baten bi ezaugarri desberdin dira: kategoria eta azpikategoria. Kasu horretan, kategoria “izena” litzateke eta azpikategoria “arrunta”. Bi elementu horiek, RNG erauzle arrunt batek, “ f ” izeneko elementu baten bi instantzia bezala ulertuko lituzke. Beraz, eskeman, honelako zerbait adieraziko luke: “Ezaugarrien barruan f motako elementu bat baino gehiago egon daiteke. f elementuak `name` izeneko atributu bat dauka, bere balioa `KAT` edo `AZP` izan daitekeelarik. Elementu honek, barruan, `sym` izeneko elementu bakarra izango du, eta bere balioa `IZE` edo `ARR` izango da.”. Interpretazio hori hurrengo adibidean ikusten da:

```
1 <element name="ezaugarriak">
2   <oneOrMore>
3     <element name="f">
4       <attribute name="name">
5         <choice>
6           <value>KAT</value>
7           <value>AZP</value>
8         </choice>
```



```

9     </attribute>
10    <element name="sym">
11      <attribute name="value">
12        <choice>
13          <value>IZE</value>
14          <value>ARR</value>
15        </choice>
16      </attribute>
17      <empty/>
18    </element>
19  </oneOrMore>
20 </element>

```

Guk, ordea, ez dugu interpretazio hori nahi, guretzat bi elementuak desberdinak baitira: bat kategoria eta bestea azpikategoria. Hau da lortu nahi den interpretazioa: “Ezaugarrien barruan bi elementu daude. Lehenaren izena f da eta bere name atributuaren balioa KAT. Bigarrenaren izena ere f da, baina bere name atributuaren balioa AZP. Lehenak sym izeneko elementu bat dauka, $value$ atributuaren balioa IZE izanik, eta bigarrenak ere izen bereko elementu bat dauka, bere $value$ atributuaren balioa ARR izanik.”. Hau da, elementuak ez dira beti izenaren arabera bereiziko. Izena f denean name atributuaren balioaren arabera erabakiko da elementu berdinez ari garen edo ez, eta izena f s denean, $type$ atributuaren arabera. Izena beste edozein denean, orduan bai, izenaren arabera desberdinduko dira. Hona hemen espero dugun interpretazio horri legokioken eskema:

```

1 <element name="ezaugarriak">
2
3   <!-- Kategoria -->
4   <element name="f">
5     <attribute name="name">
6       <value>KAT</value>
7     </attribute>
8     <element name="sym">
9       <attribute name="value">
10        <value>IZE</value>
11      </attribute>
12      <empty/>
13    </element>
14  </element>
15
16  <!-- Azpikategoria -->
17  <element name="f">
18    <attribute name="name">
19      <value>KAT</value>
20    </attribute>
21    <element name="sym">
22      <attribute name="value">
23        <value>IZE</value>
24      </attribute>
25      <empty/>
26    </element>
27  </element>
28
29 </element>

```

Ohiko erazleak erabiltzea eragozten duen beste arazo bat ere badago. Elementu batek testu-balio bat duenean, eskeman hori adierazteko bi aukera daude: elementu horrek mota (testua, zenbakia, boolearra. . .) jakin horretako balioaren bat izan behar duela esan, edo elementu horrek testu-balio jakin hori izan behar duela adierazi. Ikus dezagun adibide batekin. Demagun bi analisi dauzkagula. Bataren identifikadorea L-A-IZE-ARR-34 da eta bestearena L-A-ADI-SIN-495. Kasu horretan argi dago ez lukela zentzurik izango definizio-eskeman identifikadorearen balioak bietako bat izan behar duela esateak. Aldiz, identifikadoreak karaktere-kate edo `string` motakoa izan behar duela definitzea izango litzateke zuzena, analisi berri batek identifikadore berria izango baitu. Baina, adibidez, kategoriaren balioaren kasuan, desberdina gertatzen da, kategoriaren balio posibleek multzo itxi bat osatzen baitute. Aurreko bi analisi horietan, esaterako, bataren kategoriaren balioa IZE izango da eta bestearena ADI. Baina, zentzuzkoa al litzateke kategoriak `string` motakoa izan behar duela esatea? Ez, kasu horretan kategoriak bietako bat izan behar duela esango genuke, IZE edo ADI. Erabakia, identifikadoreetan izan ezik, beste guztietan balio desberdinak balio posible bezala proposatzea izan da, eta analisiaren identifikadorearen kasuan, `string` motakoa izan behar duela definitzea.

Lortu nahi den sistemak kontuan izan behar duen beste ezaugarri bat zera da, testu motako balioak izango dituzten elementu bakarrak `<str>` elementuak direla.

Azkenik, elementuen arteko ordena ere kontuan hartu beharrekoa da. Hau da, analisi bat baino gehiago aztertuta, beren elementuak beti ordena berean agertzen badira, definizio-eskeman ordena hori errespetatu behar dela zehaztuko da, hau da, elementuen arteko ordena esanguratsua dela. Aldiz, analisi desberdinetan elementuen ordena aldatzen bada, eta gainera ordena zehaztea ezinezkoa bada, elementuen ordena askea dela definituko da definizio-eskeman.

Betekizun horiek denak kontuan hartuz, algoritmo bat diseinatu da analisi multzo hendi batetik beren RNG definizio-eskema erazteko.

3.2.2 Algoritmoa

Algoritmoa bi fasetan banatu da. Lehenbizikoan, analisi guztiak irakurri eta eredu bat eraikiko da dagokion informazio esanguratsu guztia bilduz. Bigarrean, eredu hori RNG sintaxian adieraziko da, behar ditugun definizio-eskemak lortuz.

1. fasea

Prozesatu XML dokumentuak eta sortu eredu bat informazio horrekin:

```
2 definitu Elementu: izen, atributuak, umeak, hautazkoaDa, agerpenAnitzekoaDa
   , testuMotakoaDa;
3
4 /* Erro elementua errekursiboki prozesatzen du, edukia "erro" izeneko
   objektuan kargatuz. */
5 prozedura nagusia(inputXMLDoc):
6   erro = Elementu();
7   prozesatu_elementua(inputXMLDoc->root, erro);
8 amprozedura;
9
10 /* XML elementu bat prozesatzen du: 1) Atributuak. 2) Elementua bera. 3)
   Umeak.
11   "xmlElem" parametroa XML dokumentuko elementuaren instantzia da.
12   "elem" parametroa, aldiz, eredia gordeko den objektua. */
13 prozedura prozesatu_elementua(xmlElem, elem)
14   /* Atributuak prozesatu. Atributuen informazioa elementuari dagokion
   objektuan kargatu. */
15   prozesatu_atributuak(xmlElem, elem);
16   /* Elementuaren izena kargatu. */
17   elem.izen = xmlElem.name;
18   /* Elementua str bada, bere edukia testu motakoa izango da. */
19   baldin elem.izen == "str" orduan
20     elem.testuMotakoaDa = true;
21   else
22     /* Prozesatu ume-nodo bakoitza errekursiboki. */
23     bakoitzeko xmlElem->getChildren() as xmlChild egin
24     /* Elementua, analisi guztien artean, guraso horren ume gisa agertzen
   den lehenbiziko aldia bada elementu berri bat sortu eta bertan
   kargatu informazioa: */
25     baldin oraindik_prozesatu_gabe(xmlChild) orduan
26       ume = Elementu();
27       prozesatu_elementua(child, ume);
28       elem.gehituUmea(ume);
29       /* Gurasoa gehiagotan agertu bada, baina umea ez, umea hautazkoa
   dela esan nahi du. */
30     baldin ez gurasoaren_lehenbiziko_agerpena(elem) orduan
31       ume.hautazkoaDa = true;
32     ambaldin;
33     /* Ume elementu hau beste analisisaren batean, edo honetan bertan,
   agertu bada, kargatu dagokion objektua eta informazioa eguneratu.
   */
34     bestela:
35       ume = elem.lortuUmea(childXML);
36       prozesatu_elementua(ume, xmlChild);
37       /* Analisi honetan guraso jakin honen barruan dagoeneko horrelako
   beste ume bat aurkitu bada, agerpen anitzekoa da. */
38     baldin elementuaren_umeen_artearen_dagoeneko_prozesatua(elem, ume)
       orduan
39       ume.agerpenAnitz = true;
40     ambaldin;
41     ambaldin;
```

```

42     ambakoitzeko;
43     ambaldin;
44 amprozedura;

```

2. fasea

1. fasean eraikitako eredua hartuta, idatzi RNG sintaxia erabiliz, RNG definizio-eskemak sortzeko:

```

1
2  /* Programa nagusiak aurreko fasean lortutako eredua jaso eta goiburukoak
   idatzi ondoren, elementuari dagokion RNG kodea idatziko du errekurtsiboki
   . */
3 prozedura sortu_RNG(ereduaren_erro_elem) :
4     idatzi_rng_goiburukoak();
5     sortu_rng_elementua(ereduaren_erro_elem);
6 amprozedura;
7
8  /* Elementu bati dagokion RNG kodea errekurtsiboki idazten du. */
9 prozedura sortu_rng_elementua(elem)
10     /* Elementua optional, oneOrMore edo ZeroOrMore bada, elementuaren
       aurretik idatzi. */
11     baldin elem.hautazkoaDa eta elem.agerpenAnitzekoaDa orduan
12         idatzi('<optional>');
13     baldin not elem.hautazkoaDa eta elem.agerpenAnitzekoaDa orduan
14         idatzi('<oneOrMore>');
15     baldin elem.hautazkoaDa eta ez elem.agerpenAnitzekoaDa orduan
16         idatzi('<zeroOrMore>');
17     ambaldin;
18     /* Elementuaren etiketa zabaldu */
19     idatzi('<element name="elem.izen">');
20     /* Elementuaren atributuen kodea idatzi. */
21     bakoitzeko elem.atributuak as atributu egin
22         idatzi_atributuaren_rng(atributu);
23     ambakoitzeko;
24     /* Elementuak barruan testua badu, idatzi testu-motako eremua dela
       adierazten duen RNG kodea. */
25     baldin elem.testuMotakoaDa orduan
26         idatzi('<data type="string">');
27     ambaldin;
28     /* Umeen ordena esanguratsua ez bada, interleave elementuaren etiketa
       zabaldu lehenik. */
29     baldin ez elem.umeakOrdenatuta orduan
30         idatzi('<interleave>');
31     end_baldin;
32     /* Etiketak itxi aurretik, ume-elementu guztiekin berdina egin, banan-
       banan. */

```

```
33 bakoitzeko elem.umeak as ume egin
34     sortu_rng_elementua(ume);
35 ambakoitzeko;
36     /* Itxi zabaldutako etiketa guztiak. */
37     itxi_etiketak();
38 amprozedura;
```

Elementuen arteko ordena

Elementu baten umeen arteko ordena esanguratsua bada, hori adierazi behar da definizio-eskeman. Esanguratsua ez bada, `interleave` elementu berezia jarri behar da elementuen guraso gisa.

Hainbat analisi izanda, elementu jakin baten umeen arteko ordena esanguratsua den edo ez, eta hala bada, ordena posible bat zein den jakiteko algoritmoa azalduko da jarraian.

Hasteko, adibide pare batekin ikusiko dugu arazoa zein den. 1. kasuan, elementu guztiak hautazkoak direla suposatuz, honakoa izango litzateke a elementuaren umeen arteko ordena posible bat: $x \rightarrow z \rightarrow y \rightarrow w$. 2. kasuan, aldiz, ez dago ordena bat zehazterik. Aztertu hurrengo kodea:

```
1 <!-- 1. kasua -->
2 <root>
3   <a>
4     <y/>
5     <w/>
6   </a>
7   <a>
8     <x/>
9     <y/>
10  </a>
11  <a>
12    <x/>
13    <z/>
14    <y/>
15  </a>
16 </root>
17
18 <!-- 2. kasua -->
19 <root>
20   <a>
21     <x/>
22     <y/>
23   </a>
24   <a>
25     <y/>
26     <z/>
27   </a>
28   <a>
29     <z/>
30     <x/>
31   </a>
32 </root>
```

Kasu hauetan erraza da jakitea, baina algoritmo bat behar da kasu guztietarako balioko duena. Lehenik eta behin, informazioa nola antolatu ikusiko dugu. Izan ere, grafo baten moduan ikus dezakegu eredu. Batetik, nodoak dauzkagu: $nodoak = \{y, w, x, z\}$. Bestetik, elkarren arteko ordena adierazten duten ertz zuzenduak dauzkagu: $ertzak = \{y \rightarrow w, x \rightarrow y, x \rightarrow z, z \rightarrow y\}$.

Grafoen teoria erabiliz erraza da elkarren arteko ordena esanguratsua den edo ez adieraztea, nahikoa baita grafoa ziklikoa den edo ez esaten duen algoritmo bat erabiltzea. Grafoak ziklorik badu, ordena ez da esanguratsua, ziklorik ez badu, posible da ordena jakin bat edo gehiago definitzea.

Grafoak ziklorik ez duela ondorioztatzen bada, eta beraz, elementuen arteko ordena esanguratsua dela, ordena posible bat bilatuko duen algoritmoa behar dugu. Lan hori ere oso sinplea da grafoen teoria erabilia, nahikoa baita elementu guztietatik pasatzen den bide bat aurkitzea.

3.2.3 Arazoak

RNG eskemei dagokienez, elementuen arteko ordenarekin lotutako arazo nagusi bat izan dugu. Ordena esanguratsua denean ez dago arazorik, baina RNG eskemetan ordena askea dela adierazteko `interleave` elementu berezia erabili behar da, eta arazo bat ematen du gure analisisien egitura dela eta. Lehen esan dugun moduan, analisisien XML dokumentuetan elementu gehien izena `fs` edo `f` izango denez, atributuaren balioen arabera bereizten ditugu elementuak. RNG lengoaiak, ordea, elementuen izena bakarrik hartzen du kontuan, eta beraz, `f` izeneko elementu guztiak berdinak dira beretzat, eta gauza bera gertatzen da `fs` elementuekin ere. Hori dela eta, `interleave` patrioiak arazoak ematen ditu barruan izen bereko elementu bat baino gehiago dauzkanean, ez baitu onartzen `interleave` barruan elementurik errepikatuta agertzea.

Arazo hori gainera kentzeko aukera bakarra ikusi dugu. Aurrebaldintza bat jarri dugu analisisien egiturari: elementu ororen umeen arteko ordenak berdina izan behar du analisi guztietan. Hori errespetatzen bada, ez da `interleave` patrioiaren beharrik izango, eta beraz, arazoak ez du eraginik izango.

3.3 Analisisien normalizazioa

Urteak dira Ixa taldean testuak prozesatzen direla. Analisi linguistikoek, beraz, urteak daramatzate gure artean, eta urteetan zehar beren egitura aldatzen joan da, hobekuntzak gehituz eta gabeziak asetuz. Hasiera batean analisisien definizio-eskemak egin baziren ere, zaharkituta gelditu dira. Orain, definizio-eskema bat hutsetik sortzeko prozesuan, zenbait arazo aurkitu dira analisisien edukian. Hori dela eta, une egokia dela deritzogu arazo horiek konpondu eta analisisiek egitura zuzen eta zehatz bat izan dezaten.

Aurkitu den arazoetako bat elementuek analisisetan duten ordena izan da. Analisi desberdinetan, elementuak ordena desberdinetan agertzen dira. Horrek tresna gehien funtzionamendurako arazorik eragiten ez duen arren, koherentzia mantzentzearren bada ere, egokiena ordena bat definitzea izango litzatekeela ikusi da. Izatez, ordenatu ez ezik, ezaugarriak multzokatu ere egin dira, multzo logikotan banatu baitira ezaugarriak. Adibidez, kategoriaren araberrako ezaugarriak multzo batean bildu dira, flexio-ezaugarriak (kasua, numeroa etab.) beste batean, etab. Hartara, ordena hutsa baino pauso bat haratago doan egitura logikoago bat lortu da, ezaugarrien listei dagokienean behintzat. Ezaugarriak multzo logikotan nola banatu den ikusteko joan eranskinetako 7.1 atalera.

Hori eta beste zenbait zuzenketa egin dira diseinuan, baita inplementatutako definizio-eskeman ere, baina analisiak ezin izan dira egokitu, oraindik, proiektu honen amaierara iritsi garen honetan. Izatez, analisisien hurrengo bertsiorako konponketa horiek eginda egoitea espero da, baina bitartean analisi zaharkituak ditugu esku artean. Definizio-eskemak, ordea, bertsio berrientzat pentsatuta egin dira. Hori dela-eta, ezin izan da balidatze-prozesua burutu. Prozesu hori garrantzitsua da oso, analisisien egitura konplexua dela-eta, zaila baita esaten definizio-eskema erabat zuzena den, edo, aldiz, kontuan hartu gabeko salbuespenen bat gelditu den. Horregatik, analisisien normalizazioa egitea izan da proiektu honen lanaren zati bat. Analisisien normalizazioa deritzogu analisisien uneko bertsioei proiektu honetan zehar proposatu diren konponketak aplikatu eta analisi egokiak lortzeko prozesuari. Lortutako analisi normalizatuak proiektu honetarako bakarrik erabiliko dira, definizio-eskemak probatu ahal izateko soilik.

Hori egiteko XSLT script bat inplementatu da, sarrera gisa hainbat analisi-fitxategi jaso eta emaitza gisa analisi berrituak itzuliko dituen, proposatutako konponketak aplikatuz.

3.4 Analisi linguistikoaren definizioa

Atal honetan analisisien definizio-eskema ulertzeko adibide batzuk landuko dira. Eskema osoa eranskinetan dago ikusgai, 7.2 atalean.

3.4.1 Segmentazioa

Segmentazioa da tokenei aplikatzen zaien lehenbiziko prozesamendua. Segmentazio-prozesua amaitzean, analisi bakoitzak segmentazio-analisi bat edo gehiago izango ditu esleituta. Hitzak berak esaten duen bezala, segmentazio-prozesuan, tokenak segmentutan banatzen dira, horietako bakoitza informazio linguistikoz hornituz.

Har dezagun, adibide gisa, *guztiak* hitza. Ikusi, jarraian, hitz horren segmentazio-analisia:

```
1 <!-- Adibidea: guztiak -->
2
3 <fs id="S-A-DET-ORO-3" type="segmentazioa">
4   <f name="forma">
5     <str>guztiak</str>
6   </f>
7   <f name="lema-morfemak" org="list">
8     <fs type="lemaSeg">
9       <f name="sarrera">
10        <fs type="Gako">
11          <f name="Sarrera">
12            <str>guzti</str>
13          </f>
14          <f name="Homografo-Id">
15            <nbr value="0"/>
16          </f>
17        </fs>
18      </f>
19      <f name="ezaugarriak">
20        <fs type="ezaugarri-lista">
21          <f name="KAT">
22            <sym value="DET"/>
23          </f>
24          <f name="AZP">
25            <sym value="ORO"/>
26          </f>
27        </fs>
28      </f>
29      <f name="twol">
30        <str>guzti</str>
31      </f>
32    </fs>
33    <fs type="morfSeg">
34      <f name="sarrera">
35        <fs type="Gako">
36          <f name="Sarrera">
37            <str>ak</str>
38          </f>
39          <f name="Homografo-Id">
40            <nbr value="2"/>
41          </f>
42        </fs>
43      </f>
44      <f name="ezaugarriak">
45        <fs type="ezaugarri-lista">
46          <f name="KAT">
47            <sym value="DEK"/>
48          </f>
49          <f name="KAS">
50            <sym value="ERG"/>
51          </f>
52          <f name="NUM">
53            <sym value="S"/>
54          </f>
55          <f name="MUG">
56            <sym value="M"/>
57          </f>
58          <f name="FSL" org="list">
59            <sym value="@SUBJ"/>
60          </f>
61        </fs>
62      </f>
63      <f name="twol">
64        <str>ak</str>
65      </f>
```



```

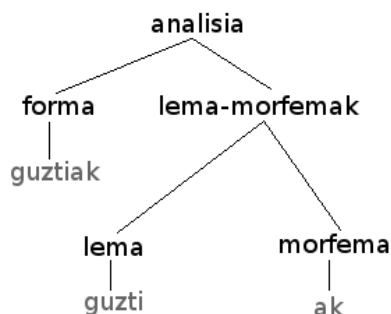
66     </fs>
67     </f>
68 </fs>

```

Gainetik ikusita XML egiturak konplexua dirudien arren, oinarrian, 9. irudiko egitura dauka. Ondoren, morfema edo lema bakoitzak informazio linguistiko zabala dauka. Informazio hori hiru ataletan banatzen da: lema edo morfemaren formari buruzko informazioa (EDBLko sarrera eta homografo-identifikadorea), ezaugarriak eta bi mailatako forma. Ezaugarriak dira lema edo morfemari buruzko informazio zabalena ematen dutenak, kategoria, azpikategoria, kasua etab. Halaber, bi mailatako formak segmentua nola deklinatuko den deskribatzen du. Hau da, hitza deklinatzerakoan abiapuntua zein den adierazten du, jasango dituen aldatetei buruzko informazioa emanez.

Adibide honetan, *guztiak* hitzari dagokionean, lema eta morfema bana daude definituta. Lema *guzti* determinatzailea izango da, eta morfema, *ak* deklinabide-atzizkia. Gainera, lemaren azpikategoria eta morfemaren kasua, numeroa eta mugatasunaren inguruko informazioa ematen da ezaugarrien artean. Bi mailatako formek, kasu hauetan, ez dute informazio gehigarririk eskaintzen.

Definizio-eskema osatzerakoan kontuan hartu behar izan dira hainbat gauza. Batetik, forma definitzea ezinbestekoa dela eta nahi adina lema edo morfema defini daitezkeela (bat gutxienez). Lema eta morfemen barne-egitura berdina da. Derrigorrezko eremuak dira sarrera, ezaugarriak eta *twol* (bi mailatako forma), eta badago adibide honetan agertu ez den hautazko eremu bat ere, aldaera. Eskema konplexu egiten duen ezaugarri bat ere badago, izan ere, definituko diren ezaugarriak desberdinak izango baitira kategoriararen arabera. Hori dela-eta, 3.2 atalean azaldu den sistemaz baliatuz, kategoria bakoitzak zein ezaugarri definitzea onartzen duen aztertu behar izan da. Horretarako, sistema egokitu da kategoria jakin bat pasatuta, kategoria horretako analisiak bakarrik kontuan har ditzan. Horrela, prozesua kategoria posible adina bider errepikatuta, kategoria bakoitzeko analisiak har dezakeen egitura zehatza erauzi ahal izan da.



9 irudia: “guztiak” hitzaren segmentazio-analisiaren egitura orokorra.

3.4.2 Morfosintaxia

Morfosintaxi mailako analisietan (Aduriz et al., 2000) hitzei informazio morfologikoa gehitzen zaie. Alde batetik, hitzak morfematan banatzen dira, horietako bakoitzaren informazio linguistikoa adieraziz. Bestalde, goi-mailako ezaugarriak gehitzen zaizkio hitzari, hau da, hitzak berak morfemetatik jasotako ezaugarriak. Azkenik, hitzari dagokion lema ere adierazten da analisi hauetan.

Adibide bezala aurreko ataleko berbera erakutsiko da. Hona hemen *guztiak* hitzari dagokion analisi morfosintaktikoa:

```

1 <!-- Adibidea: guztiak -->
2
3 <fs id="M-A-DET-ORO-13" type="analisi">
4   <f name="lema-osatua">
5     <str>guzti</str>
6   </f>
7   <f name="forma">
8     <str>guztiak</str>
9   </f>
10  <f name="goimailako-ezaugarriak">
11    <fs type="goimailako-ezaugarri-lista">
12      <f name="KAT">
13        <sym value="DET"/>
14      </f>
15      <f name="AZP">
16        <sym value="ORO"/>
17      </f>
18      <f name="KAS">
19        <sym value="ABS"/>
20      </f>
21      <f name="NUM">
22        <sym value="P"/>
23      </f>
24      <f name="MUG">
25        <sym value="M"/>
26      </f>
27      <f name="FSL" org="list">
28        <sym value="@OBJ"/>
29        <sym value="@PRED"/>
30        <sym value="@SUBJ"/>
31      </f>
32    </fs>
33  </f>
34  <f name="osagaiak" org="list">
35    <fs type="osagaia">
36      <f name="oina">
37        <fs type="lema">
38          <f name="ezaugarriak">
39            <fs type="ezaugarri-lista">
40              <f name="KAT">
41                <sym value="DET"/>
42              </f>
43              <f name="AZP">
44                <sym value="ORO"/>
45              </f>
46            </fs>
47          </f>
48        <f name="twol">

```

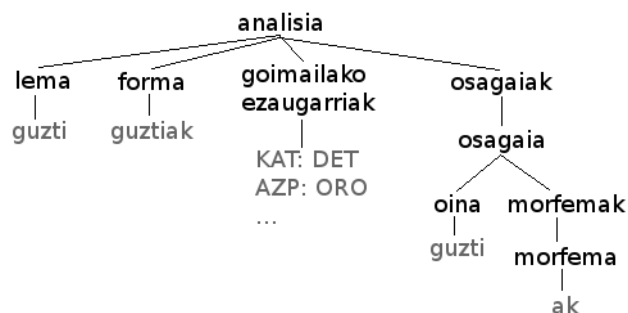
```

49     <str>guzti</str>
50     </f>
51     <f name="sarrera">
52         <fs type="Gako">
53             <f name="Sarrera">
54                 <str>guzti</str>
55             </f>
56             <f name="Homografo-Id">
57                 <nbr value="0"/>
58             </f>
59         </fs>
60     </f>
61 </fs>
62 </f>
63 <f name="morfeak" org="list">
64     <fs type="morfema">
65         <f name="ezaugarriak">
66             <fs type="ezaugarri-lista">
67                 <f name="KAT">
68                     <sym value="DEK"/>
69                 </f>
70                 <f name="MUG">
71                     <sym value="M"/>
72                 </f>
73                 <f name="NUM">
74                     <sym value="P"/>
75                 </f>
76                 <f name="KAS">
77                     <sym value="ABS"/>
78                 </f>
79                 <f name="FSL" org="list">
80                     <sym value="@OBJ"/>
81                     <sym value="@PRED"/>
82                     <sym value="@SUBJ"/>
83                 </f>
84             </fs>
85         </f>
86         <f name="twol">
87             <str>ak</str>
88         </f>
89     <f name="sarrera">
90         <fs type="Gako">
91             <f name="Sarrera">
92                 <str>ak</str>
93             </f>
94             <f name="Homografo-Id">
95                 <nbr value="1"/>
96             </f>
97         </fs>
98     </f>
99 </fs>
100 </f>
101 </fs>
102 </f>
103 </fs>

```

Analisiaren egitura orokorra 10. irudian ageri dena da. Segmentazio-analisietan segmentuei (lema-morfemei) zegozkienak orain osagaietan definituta daude. Hau da, hitza osagaietan banatuta egon daiteke, eta osagaietako bakoitza morfemetan. Adibide honetan osagai bakarra dago, eta osagaia bitan banatu da: batetik, *guzti*, osagaiaren oina izango dena, eta bestetik, *ak*, morfema. Morfema bakoitzak sarrera, ezaugarriak eta bi mailatako

forma izaten jarraitzen du. Goi-mailako ezaugarriei dagokienean, kasu honetan, oinaren kategoria da goratu dena. Hau da, *guzti* oinaren kategoria determinatzailea eta *ak* morfemarena deklinatzailea zen moduan, *guztiak* hitz osoaren kategoria determinatzailea da. Azpikategoria ere oinarena goratu da, baina goratu diren kasua, numeroa, mugatasuna eta funtzio sintaktikoak *ak* morfemarenak izan dira. *guztiak* hitzaren lema osatua *guzti* da orain.



10 irudia: “guztiak” hitzaren morfosintaxi-analisiaren egitura orokorra.

Definizio-eskemari begira, kontuan hartu beharreko kontu pare bat. Batetik, lema-osatua eremua hatuazkoa da, baita adibide honetan ageri ez den aldaera-osatua ere. Segmentazioko analisietan ageri zen lema-morfemak eremuaren orde, orain, osagaiak eremua dago. Barruan nahi adina osagai egon daitezke, eta horietako bakoitzak hautazko oin bat eta hainbat morfema izan ditzake. Gainera, morfema bakoitzak bere ezaugarriak dituen bezala, osagai bakoitzak ere bereak izan ditzake definituta. Kontuan izan behar da osagaien oin edo morfemen ezaugarriak segmentazioko berdinak izan daitezkeela, baina goi-mailako ezaugarriak, aldiz, aldatu egiten direla. Izan ere, goimailako ezaugarriak ez dira kategorien araberakoak, morfema desberdinetatik jasotzen baitira ezaugarriak. Gainera, kategorietako batzuek besterik ezin dute izan goratuak. Ondorioz, segmentazioko ezaugarrien definizioak berrerabili egin dira osagaien ezaugarrietarako, baina goi-mailako ezaugarriak birdefinitu egin behar izan dira. Amaitzeko, osagaietatik kanpo ere hautazko oina eremua dago, osagai baten oina izan orde, hitz osoarena izango litzatekeena.

Beste adibide bat erakutsiko da jarraian, aurrekoa baino konplexuagoa. Adibideko analisia *bildutakoek* hitzari dagokio. Adibide honetan, arestian aipatutako eremu gehienak daude definituta:

```

1 <!-- Adibidea: bildutakoek -->
2
3 <fs id="M-A-ADI_IZEELI-1" type="analisa">
4   <f name="lema-osatua">
5     <str>bildu</str>
6   </f>
7   <!-- ... -->
8   <f name="osagaiak" org="list">
9     <fs type="osagaia">

```

```
10 <f name="oina">
11   <fs type="lema">
12     <f name="ezaugarriak">
13       <fs type="ezaugarri-lista">
14         <f name="KAT">
15           <sym value="ADI"/>
16         </f>
17         <!-- ... -->
18       </fs>
19     </f>
20     <f name="twol">
21       <str>bil</str>
22     </f>
23     <f name="sarrera">
24       <fs type="Gako">
25         <f name="Sarrera">
26           <str>bildu</str>
27         </f>
28         <!-- ... -->
29       </fs>
30     </f>
31   </fs>
32 </f>
33 <f name="ezaugarriak">
34   <fs type="ezaugarri-lista">
35     <f name="KAT">
36       <sym value="ADI"/>
37     </f>
38     <!-- ... -->
39   </fs>
40 </f>
41 <f name="morfemak" org="list">
42   <fs type="morfema">
43     <f name="ezaugarriak">
44       <fs type="ezaugarri-lista">
45         <f name="KAT">
46           <sym value="AMM"/>
47         </f>
48         <!-- ... -->
49       </fs>
50     </f>
51     <f name="twol">
52       <str>tu</str>
53     </f>
54     <f name="sarrera">
55       <fs type="Gako">
56         <f name="Sarrera">
57           <str>tu</str>
58         </f>
59         <!-- ... -->
60       </fs>
61     </f>
62   </fs>
63   <fs type="morfema">
64     <!-- ... -->
65     <f name="sarrera">
66       <fs type="Gako">
67         <f name="Sarrera">
68           <str>ta</str>
69         </f>
70       </fs>
71     </f>
72   </fs>
73   <fs type="morfema">
74     <!-- ... -->
```

```
75         <f name="sarrera">
76             <fs type="Gako">
77                 <f name="Sarrera">
78                     <str>ko</str>
79                 </f>
80                 <!-- ... -->
81             </fs>
82         </f>
83     </fs>
84 </f>
85 </fs>
86 <fs type="osagaia">
87     <f name="oina">
88         <fs type="lema">
89             <!-- ... -->
90             <f name="sarrera">
91                 <fs type="Gako">
92                     <f name="Sarrera">
93                         <str>0</str>
94                     </f>
95                     <f name="Homografo-Id">
96                         <nbr value="14"/>
97                     </f>
98                 </fs>
99             </f>
100         </fs>
101     </f>
102     <f name="ezaugarriak">
103         <fs type="ezaugarri-lista">
104             <f name="MUG">
105                 <sym value="M"/>
106             </f>
107             <!-- ... -->
108         </fs>
109     </f>
110     <f name="morfemak" org="list">
111         <fs type="morfema">
112             <!-- ... -->
113             <f name="sarrera">
114                 <fs type="Gako">
115                     <f name="Sarrera">
116                         <str>0</str>
117                     </f>
118                     <!-- ... -->
119                 </fs>
120             </f>
121         </fs>
122         <fs type="morfema">
123             <!-- ... -->
124             <f name="sarrera">
125                 <fs type="Gako">
126                     <f name="Sarrera">
127                         <str>ek</str>
128                     </f>
129                     <!-- ... -->
130                 </fs>
131             </f>
132         </fs>
133     </f>
134 </fs>
135 </f>
136 </fs>
```

Adibide honen berezitasun aipagarrienetako bat, aurrekoarekin alderatuta, hitzaren oinaren definizioa da. Kasu honetan, *bildu* da oina, eta bere ezaugarriak eta bi mailatako forma ere definituta dauzka. Horrez gain, osagaietan orain bi osagai daude, bata *bildutako* zatiari dagokiona, eta bestea, *ek* zatiari dagokiona. Lehenak oina, *bil*, eta hiru osagai, *tu*, *ta* eta *ko*, ditu. Bigarrena, aldiz, berezia da. Izan ere, *ek* soilik da osagaiaren testu-forma, baina hain motza izanagatik ere oina eta bi morfema ditu. Gertatzen dena da oina eta morfemetako bat elidituta daudela, hau da, bi elipsi daudela.

3.4.3 Lematizazioa

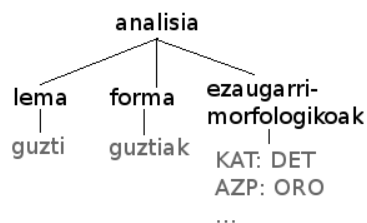
Euskarazko lematizatzaileak (Aduriz et al., 1996) morfosintaxi mailako analisiekin lan egin eta lematizazio-analisiak sortzen ditu. Hauek morfosintaxi mailakoak baino sinpleagoak dira, lema mailan lan egiten dutenez, goi-mailako informazioarekin bakarrik gelditzen baitira, gainontzekoa baztertuz.

Hona hemen aurreko bi kasuetan ikusi den adibidea, oraingoan, lematizazio mailan analizatuta:

```
1 <!-- Adibidea: guztiak -->
2
3 <fs id="L-A-DET-ORO-13" type="lematizazioa">
4   <f name="forma">
5     <str>guztiak</str>
6   </f>
7   <f name="lema-osatua">
8     <str>guzti</str>
9   </f>
10  <f name="ezaugarri-morfologikoak">
11    <fs type="goimailako-ezaugarri-lista">
12      <f name="KAT">
13        <sym value="DET"/>
14      </f>
15      <f name="AZP">
16        <sym value="ORO"/>
17      </f>
18      <f name="KAS">
19        <sym value="ABS"/>
20      </f>
21      <f name="NUM">
22        <sym value="P"/>
23      </f>
24      <f name="MUG">
25        <sym value="M"/>
26      </f>
27    </fs>
28  </f>
29 </fs>
```

Oraingoan, lema eta formaz gain, morfosintaxi mailako analisisian goi-mailako ezaugarri zirenak bakarrik jaso dira, osagaietan banatutako informazio linguistiko guztia baztertuz. Gainera, goimailako ezaugarrietatik ere ez da informazio guztia mantendu, funtzio sintakti-

koak baztertu egin baitira. Beraz, informazio garrantzitsuena besterik ez da bildu, analisisa asko sinplifikatuz. Egitura orokorra 11. irudian ikus daiteke.



11 irudia: “guztiak” hitzaren lematizazioaren egitura orokorra.

Morfosintaxian ikusi den beste adibidea ere azalduko da jarraian. Morfosintaxi mailan oso analisi luzea eta konplexua zen. Orain aldiz, goi mailako ezaugarriak bakarrik jaso direnez, erruz sinplifikatu da analisisa:

```

1 <!-- Adibidea: bildutakoek -->
2
3 <fs id="L-A-ADI-SIN-760" type="lematizazioa">
4   <f name="forma">
5     <str>bildutakoek</str>
6   </f>
7   <f name="lema-osatua">
8     <str>bildu</str>
9   </f>
10  <f name="ezaugarri-morfologikoak">
11    <fs type="goimailako-ezaugarri-lista">
12      <f name="KAT">
13        <sym value="ADI"/>
14      </f>
15      <f name="AZP">
16        <sym value="SIN"/>
17      </f>
18      <f name="ADM">
19        <sym value="PART"/>
20      </f>
21      <f name="KAS">
22        <sym value="ERG"/>
23      </f>
24      <f name="NUM">
25        <sym value="P"/>
26      </f>
27      <f name="MUG">
28        <sym value="M"/>
29      </f>
30      <f name="ADOIN">
31        <str>bil</str>
32      </f>
33    </fs>
34  </f>
35 </fs>

```

Dena dela, beti ez da osagaien inguruko informazio guztia baztertzen. Lematizazio mailan ere posible da osagai bakoitzaren informazioa agertzea. Hala ere, osagai bakoitzak

HAP masterra

daraman informazioa sinpleagoa da morfosintaxian baino, oinaren forma eta goi-mailako ezaugarriak baitira gordetzen diren bakarrak. Hurrengo adibidean ikus daiteke aipatutakoa; *zena* hitzak bi osagai ditu: *zen* adizkia, batetik, eta absolutiboan doan elipsia, bestetik.

```

1 <!-- Adibidea: zena -->
2
3 <fs id="L-A-ADL_IZEELI-1" type="lematizazioa">
4   <f name="forma">
5     <str>zena</str>
6   </f>
7   <f name="lema-osatua">
8     <str>izan</str>
9   </f>
10  <f name="ezaugarri-morfologikoak">
11    <fs type="goimailako-ezaugarri-lista">
12      <f name="KAT">
13        <sym value="ADL_IZEELI"/>
14      </f>
15      <f name="KAS">
16        <sym value="ABS"/>
17      </f>
18      <f name="NUM">
19        <sym value="S"/>
20      </f>
21      <f name="MUG">
22        <sym value="M"/>
23      </f>
24      <f name="MDN">
25        <sym value="B1"/>
26      </f>
27      <f name="NOR">
28        <sym value="HURA"/>
29      </f>
30    </fs>
31  </f>
32  <f name="osagaiak" org="list">
33    <fs type="osagaia">
34      <f name="oina">
35        <str>zen</str>
36      </f>
37      <f name="ezaugarri-morfologikoak">
38        <fs type="goimailako-ezaugarri-lista">
39          <f name="KAT">
40            <sym value="ADL"/>
41          </f>
42          <f name="ERL">
43            <sym value="ERLT"/>
44          </f>
45          <f name="MDN">
46            <sym value="B1"/>
47          </f>
48          <f name="NOR">
49            <sym value="HURA"/>
50          </f>
51        </fs>
52      </f>
53    </fs>
54    <fs type="osagaia">
55      <f name="oina">
56        <str>0</str>
57      </f>

```

```
58     <f name="ezaugarri-morfologikoak">
59       <fs type="goimailako-ezaugarri-lista">
60         <f name="MUG">
61           <sym value="M"/>
62         </f>
63         <f name="NUM">
64           <sym value="S"/>
65         </f>
66         <f name="KAT">
67           <sym value="ELI"/>
68         </f>
69         <f name="KAS">
70           <sym value="ABS"/>
71         </f>
72       </fs>
73     </f>
74   </fs>
75 </f>
76 </fs>
```

4 Armiarma: Analisi linguistikoaren sorkuntza eta kontsulta-ahalmena

Proiektu honen helburuetako bat anotazio linguistiko konplexuak kudeatzeko Armiarma aplikazioari bi funtzionalitate berri gehitzea izan da: corpusaren gainean kontsultak egiteko sistema berria inplementatzea eta analisi berrien eskuzko sorkuntza ahalbidetzea.

Bi funtzionalitateek ezaugarri komun bat dute, alegia, analisisien definizioa zehazten duten RNG eskemak erabili behar direla eginkizuna behar bezala betetzeko. Hau da, kontsulta-sistemak gai izan behar du une bakoitzean proposatu behar dituen ezaugarriak ordura arte zehaztu direnen arabera izateko, eta gauza bera egin behar du analisiak eskuz sortzen lagunduko duen funtzionalitateak ere, une bakoitzean erabiltzaileari eskatu-ko dizkion eremu/balioak aurrez definitutako arabera izan beharko baitute. Hori dela-eta, analisisien RNG eskemak irakurri eta parseatzeko liburutegi bat garatu da, funtzionalitate berrien garapena errazteko.

Atal honetan RNG eskemak kudeatzeko liburutegia azalduko da lehenik, zein eginkizun betetzen dituen argituz. Ondoren, Armiarma gehitu zaizkion bi funtzionalitateak xehe-tasun gehiagorekin azalduko dira, kontsulten sistemaren hobekuntza lehenik, eta analisisien eskuzko sorkuntzan laguntzeko sistema ondoren.

4.1 RNG eskemak kudeatzeko liburutegia

Analisi linguistikoak Relax NG (RNG) lengoaiatzat idatzitako eskema batean definitu dira. Arazoa da tresna laguntzaile edo API ⁴ eskaintza urria dagoela lengoia horretan idatzitako eskemekin lan egiteko. Gainera, bai analisi berrien sorkuntzan eta bai kontsulten definizioan, RNGz idatzitako eskemak erabili behar ditu sistemak. Hori dela-eta, API bat diseinatu eta inplementatu da zeregin horietan laguntzeko. API honek laguntza eskaini behar du analisisien definizio-eskemak erabiltzeko garaian.

APIaren funtzionamendua azalduko da atal honetan. Bere zereginetako bat fitxategi desberdinetan definitutako RNG eskemak datu-egitura batean biltzea da. Horretarako FSSchema klasea eskaintzen du, eraikitzailean bertan eskemaren fitxategi nagusiaren kopena eta analisi-maila ⁵ pasatuta fitxategi guztien edukia bildu eta objektuaren barruko eremu batean eduki guztia antolatu eta gordeko duena. Maila zehaztea beharrezkoa da, izan ere, segmentazio, morfosintaxi eta lematizazio mailako eskemak, hirurak, eskema berean bilduta baitaude. Behin hori eginda, FSSchema klaseak eskaintzen dituen metodoak erabilgarri egongo dira:

⁴Ingelesez Application Programming Interface. Programatzeko interfaze bat izan ohi da, software-modulu desberdinak elkarren artean komunikatzeko. Askotan, zeregin batean laguntzeko eta maila abstraktuago batean lan egiteko eskaintzen den liburutegi bezala erabiltzen da.

⁵Analisi-maila segmentazioa (SEG), morfosintaxia (MORF) edo lematizazioa (LEM) izan daiteke.

- `string getId()`: Analisiaren identifikadorea itzultzen du.
- `Node getRootNode()`: Analisiaren erro-elementua itzultzen du. Beharrezkoa da analisi bat goitik behera prozesatu nahi denean, erroetik abiatu eta nodo bakoitza banan-banan prozesatzeko. Nodo bakoitzak bere barruan testu soila edo beste nodo zerrenda bat izan dezake. Nodoekin lan egiteko ere `Node` klasea eskaintzen da, aurrerago azalduko dena.
- `List<string> getKategoriak()`: Katetoria posibleen zerrenda itzultzen du. Oso lagungarria da kontsulten formularioa inplementatzeko garaian, beti eskaini behar baitira katetoria posible guztiak katetoriaren balioa eskatzen denean, eta ondoren, katetoriaren balioa izango da hurrengo eremuak baldintzatuko dituen.
- `List<Node> getEzaugarriak(string Katetoria)`: Metodo honek, katetoria bat emanda, katetoria horretako analisisiek izan ditzaketan ezaugarriak itzultzen ditu. Morfosintaxi eta lematizazio mailako analisisetan goi-mailako ezaugarriak itzuliko ditu, eta segmentazio mailakoetan, aldiz, ezaugarri arruntak. Kontsulta-sistemaren inplementaziorako da interesgarriena, izan ere, erabiltzaileak katetoria aukeratzen duenean hasten baita sistema eskemak irakurtzen. Kontsulta zehazteko garaian, ezaugarrien balioak dira erabiltzaileak definituko dituenak, eta beraz, ezaugarrien atala da aplikazioak eskemetatik jaso behar izango duena.

Ikus daitekeen bezala, eragiketa arruntek `Node` motako objektuak itzultzen dituzte askotan. `Node` motako objektu bakoitzak eskemako nodo bat adierazten du, edozein motatakoa dela ere (`element`, `optional`, `oneOrMore` etab.), eta elementu horien informazioa jasotzeko eragiketak eskaintzen ditu:

- `string getMota()`: Elementua RNG lengoaiaren zein elementu motakoa den itzultzen du. Hauek dira mota posibleak: `element`, `optional`, `choice`, `zeroOrMore`, `oneOrMore`, `data` eta `value`.
- `string getTag()`: Elementu bakoitzak etiketa bat dauka esleituta, formularioak sortzean erabiltzaileari erakutsi beharreko izena izango dena. Metodo honek etiketa hori itzultzen du.
- `string getChildrenTag()`: Askotan, elementu baten edukia deskribatu behar da hitzez. Adibidez, analisisien eskuzko sorkuntzan, `optional` motako elementu batean gaudenean, barruko edukia definitu nahi den galdetuko zaio erabiltzaileari. Horretarako, ordea, erabiltzaileak jakin behar du zein den eduki hori. Metodo honek elementuaren umeen etiketekin osatutako etiketa bat itzultzen du.
- `List<Node> getChildren()`: Nodoren umeekin osatutako lista bat itzultzen du.

- `string getRefName()`: `Ref` motako elementuetan soilik erabilgarri. `Ref` elementuak erreferentzia egiten dion `define` moduluaren izena itzultzen du.
- `string getElementName()`: `Element` motako nodo batekin bakarrik erabil daiteke. Elementuaren izena itzultzen du, hau da, elementuaren `name` atributuaren edukia.
- `string getValue()`: `Value` motako nodo batekin bakarrik erabil daiteke. Elementuaren testu-edukia itzultzen du.
- `List<string> getValues()`: Elementuaren izena `sym` duten `Element` motako nodoetan bakarrik erabil daiteke. `Sym` elementu horrek har ditzakeen balio posibleak itzultzen ditu.

4.2 Analisisien eskuzko sorkuntza Armiarman

Lexikoaren Behatokia proiektuan automatikoki prozesatzen dira testuak. Prozesu automatiko horretatik lortzen dira testuko hitz bakoitzaren analisi linguistikoak. Noski, prozesua automatikoa denez, baliteke batzuetan analisiak zuzenak ez izatea. Honez gero aipatu den moduan, hitz batek analisi posible bat baino gehiago izan dezake testuinguruaren arabera, eta hortaz, testuko hitz bakoitzarentzat analisi bat baino gehiago sortuko dira testua prozesatzean. Hitz jakin batentzat zuzena den analisisirik sortu ez bada, erabiltzaileak analisi hori eskuz sortzeko aukera izan behar du.

Analisiak oso konplexuak dira haien egiturari dagokienez. Ereku asko izan ditzakete beren izaeraren arabera, eta eskuz sortzeko garaian idatzi behar litzatekeen XML kodea ugaria eta astuna izango litzateke. Hori dela-eta, Armiarmak analisiak sortzen laguntzen duen funtzionalitate bat eskaintzen du. Funtzionalitate hau erabili ahal izateko, kontsulta bat egin ondoren, hitz baten gainean klikatu behar da bere analisi posibleak lortzeko. Zerrenda horrekin batera, analisi berri bat sortzeko esteka agertuko da. Bertan klikatuta analisi berri bat sortzen lagunduko duen tresna martxan jarriko da.

Analisien sorkuntzarekin laguntzeko Armiarmak formulario dinamiko bat eskaintzen du. Formulario horrek erabiltzailea gidatuko du analisiaren sorkuntzan zehar, unean uneko eremuen balioak sartzeko eskatuz. Eskatu beharreko eremuak, ordea, dinamikoki aukeratu behar dira, erabiltzailea sartuz doan balioen arabera aldatu egingo baitira. Hori egiteko analisisien definizio-eskemak erabiltzen dira. Aplikazioak eskemak irakurriko ditu, honela, beren egitura xehetasunez ezagutzeko. Horrela, une bakoitzean sistemak erabakiko du zein den eskatu beharreko hurrengo eremua. Zenbait kasutan, baliteke hurrengo pausoa definitu beharreko eremu posibleak bat baino gehiago izatea. Adibidez, eremu jakin bat hautazko bezala definituta badago, erabiltzaileak izango du eremu hori definitu nahi duen edo zuzenean hurrengoarekin jarraitu behar den erabakitzeke ardurak. Horrelako kasuetan erabiltzaileari galdetuko zaio, formularioan bertan, hautazko eremua definitu nahi duen edo

ez. Antzeko kasu gehiago ere badaude, adibidez, hainbat eremuren artean bakarra aukeratu behar denean, edo eremu bat nahi adina aldiz errepika daitekeenean zenbatetan errepikatu nahi den adierazi behar denean. Sorkuntza-prozesua amaitzean sistemak dagokion XML kodea sortuko du, eta analisisien datu-basean analisi berria txertatu.

4.2.1 Algoritmoa

Idea sinplea da: analisisien definizio-eskema errekurtsiboki prozesatzen joatea elementu bakoitzari dagokion XML kodea sortuz. Zenbaitetan erabiltzailearen parte-hartzea beharrezkoa da, balioak sartzeko (data motako elementuetan) edota optional edo oneOrMore bezalako elementuetan zein erabaki hartu jakiteko. Jarraian ikus daiteke algoritmoa:

```

1
2  /* Programa nagusia. FSSchema motako elementu bat sortzen du eta erroari
   dagokion XML kodea sortzen du. Irekiera eta itxiera etiketen artean
   barruko elementu guztiak prozesatzeko deia egiten du, hauek,
   errekurtsiboki, gauza bera egin dezaten beren edukiarekin. */
3 prozedura nagusia(schema_path, maila)
4   /* FSSchema objektua sortu eta kargatu */
5   schema = FSSchema(schema_path, maila);
6   /* Erro elementua lortu eta dagokion XML kodea sortu. */
7   root = schema->getRootNode();
8   analysis = "";
9   openTag(analysis, root->getElementName());
10  /* Etiketa zabaldu ondoren, itxi aurretik, barruko elementuak prozesatuko
   dira, hauek sortutako kodea erro elementuaren barruan gelditu dadin. */
11  processChildren(root);
12  closeTag(analysis, root);
13 amprozedura;
14
15  /* Nodo baten ume guztiak prozesatzen ditu banan banan, beren XML kodea
   sortuz. */
16 prozedura processChildren(Node node)
17   bakoitzeko node->getChildren() as child egin
18   /* Elementu motaren arabera prozesua desberdina da. Beraz, dagokion
   funtzioari dei egin. */
19   switch child->getMota() non
20     /* Ref motako nodo batera iristen denean, dagokion define elementura
   jauzi egin eta jarraitu prozesatzen. */
21     baldin ``ref``: processRefNode(child, schema);
22     /* Element motako nodo batean, dagokion irekiera etiketa sortu,
   barruko edukia errekurtsiboki prozesatu, eta etiketa itxi. */
23     baldin ``element``: processElementNode(child, schema);
24     /* Optional motako elementu batean, erabiltzaileari galdetu behar zaio
   ea edukia definitu nahi duen. */
25     baldin ``optional``: processOptionalNode(child, schema);
26     /* Choice motako elementu batean, barruko elementuen artean bakarra

```

```

    aukeratzeko eskatu behar zaio erabiltzaileari. Aukeratutako
    elementua prozesatuko da soilik. */
27  baldin ``choice``: processChoiceNode(child, schema);
28  /* ZeroOrMore motako elementu batean, edukia nahi adina aldiz
    definitzen utziko zaio erabiltzaileari, iterazio bakoitzean edukia
    berriz definitu nahi duen galdetuz. */
29  baldin ``zeroOrMore``: processZeroOrMoreNode(child, schema);
30  /* OneOrMore motako elementu batean, edukia gutxienez behin definituko
    da, eta hortik aurrera, iterazio bakoitzean erabiltzaileari
    galdetuko zaio edukia berriz definitu nahi duen. */
31  baldin ``oneOrMore``: processOneOrMoreNode(child, schema);
32  /* Data motako elementu batean, erabiltzaileari balio bat sartzeko
    eskatuko zaio. */
33  baldin ``data``: processDataNode(child, schema);
34  /* Value motako nodo batean, zuzenean bere balioa analisiaren kodean
    gehituko da. */
35  baldin ``value``: processValueNode(child, schema);
36  amswitch;
37  ambakoitzeko;
38  amprozedura;

```

Hau dena ulertzeko adibide simple bat azalduko da jarraian. Eman dezagun analisisien definizio-eskema jarraian ageri dena bezain sinplea dela:

```

1  <start>
2  <!-- Analisiaren definizioa -->
3  <element name="fs">
4  <!-- Analisi mota -->
5  <attribute name="type">
6  <value>segmentazioa</value>
7  </attribute>
8  <!-- Analisiaren identifikadorea -->
9  <attribute name="id">
10 <data type="string"/>
11 </attribute>
12 <!-- Analisiaren kategoria -->
13 <element name="f">
14 <attribute name="name">
15 <value>KAT</value>
16 </attribute>
17 <element name="sym">
18 <attribute name="value">
19 <choice>
20 <value>ADB</value>
21 <value>ADI</value>
22 <value>IZE</value>
23 </choice>
24 </attribute>
25 <empty/>
26 </element>
27 </element>
28 </element>
29 </start>

```

Kasu horretan, honakoa izango litzateke prozesua. Lehenik, erro elementua eskuratu eta dagokion etiketa zabalduko litzateke, baina etiketa zabaltzearekin batera atributuak ere prozesatu behar dira, etiketa barruan baitoaz. Kasu honetan, lehenbiziko atributuaren kasuan ez da erabiltzailearen parte-hartzerik behar, atributu-izena `type` baita eta balioa segmentazioa. Bigarren atributuaren kasuan, aldiz, erabiltzaileari galdetu beharko zaio, `id` atributuaren edukia `data` motakoa baita. Horrek esan nahi du `string` motako balioen bat izango dela, baina erabiltzaileak adierazi behar du balio zehatza zein den. Behin hori eginda, erro-elementuaren etiketa sortuko litzateke:

```
1 <fs type="segmentazioa" id="M-A-IZE-ARR-234">
```

Jarraian, etiketa itxi aurretik, elementu umeak prozesatzeari ekingo zaio, kasu honetan kategoriari dagokiona. Nodoa `element` motakoa denez, elementuari dagokion etiketa zabalduko da, honelako zerbaite lortuz:

```
1 <fs type="segmentazioa" id="M-A-IZE-ARR-234">
2   <f name="KAT">
```

Prozesuarekin jarraituz, azken nodoaren umea prozesatzeari ekingo zaio. Nodo berri hau ere `element` motakoa da, bere balioa `sym` delarik. Beraz, elementu honi dagokion kodea sortu beharko litzateke, baina bere atributua `choice` motakoa dela ikusita, erabiltzaileari honako galdera egingo zaio: *KAT ezaugarriarentzat balio hauetako bat aukeratu: ADB, ADI edo IZE*. Demagun erabiltzaileak ADI aukeratu duela, beraz, ume-nodoen artean bigarrena aukeratuko da:

```
1 <fs type="segmentazioa" id="M-A-IZE-ARR-234">
2   <f name="KAT">
3     <sym value="ADI"/>
```

Elementu guztiak prozesatu direnez, etiketak ixtea bakarrik gelditzen da. Honakoa litzateke sortutako analisia:

```
1 <fs type="segmentazioa" id="M-A-IZE-ARR-234">
2   <f name="KAT">
3     <sym value="ADI"/>
4   </f>
5 </fs>
```


4.3 Analisi-kontsulten hobekuntza Armiarman

Armiarmaren oinarrizko funtzionalitatea corpusen gainean kontsultak egitearena da. Erabiltzaileak kontsulta bat definituko du zenbait baldintza zehaztuz eta sistemak corpus osoan zehar baldintza horiek betetzen dituzten agerpenak erakutsi pantailan.

Kontsultak egiteko hiru terminoko leiho bat definitzeko aukera eskaintzen da. Orain arte, termino bakoitza definitzeko, forma edo lemaz gain kategoria eta azpikategoria baka-rik zehaztu zitezkeen. Gainera, inkonsistentziak sor zitezkeen, azpikategoriaren aukeren artean azpikategoria guztiak agertzen baitziren aukeratutako kategoria edozein zela ere. Hitz batek, ordea, ezaugarri asko izan ditzake kategoria eta azpikategoriaz gain.

Ezaugarriak elkarren menpe daude, askotan. Hau da, hitz batek ezaugarri jakin bat izan dezake, beste ezaugarri jakin baten balioa dagokiona bada. Adibidez, hitz batek numeroa-mugatasuna (NMG) ezaugarria bere kategoria *determinantea* bada soilik izango du. Baliteke ere ezaugarri bat beste ezaugarri jakin bat agertzen ez bada bakarrik agertu ahal izatea. Adibidez, hobetsiak ezaugarria estandarreak ezaugarria ez badu soilik agertuko da. Menpekotasun gehienak kategoriaren arabera dira. Kategoriak baldin-tzatzen du, beraz, analisi baten egitura neurri handi batean.

Funtzionalitate hau implementatzeko formulario dinamiko bat erabiltzea hautatu da. Formularioak hasieran forma edo lema, aldaera, modua eta kategoria hautatzeko aukera eskainiko du, besterik gabe. Erabiltzaileak eremu hauek betetzen dituztenean, zehaztutako kategoriaren arabera, ezaugarri berriak proposatuko zaizkio. Kategoria eta gero azpikate-goria da normalean eskatuko dena, baina ez beti, zenbait kategoriatako hitzek ez baitute azpikategoriarik. Proposatuko diren eremuak, esan bezala, analisisien definizio-eskemetan daudenak dira. Horiek 3.4. atalean kontsultatu daitezke xehetasun guztiekin.

4.3.1 Algoritmoa

Forma, lema eta moduaren formulario-atalak estatikoki definituta daude, hauek ez baitira analisisien definizio-eskemaren arabera. Kategoria aukeratzeko unetik hasten da sistema analisisien definizio-eskemak irakurtzen. Lehenik, kategoriaren balio posible guztiak lortu behar dira. Erabiltzaileak kategoria aukeratzeko duenean sistemak definizio-eskemak ku-deatzeko APIa erabiltzen du aukeratutako kategoriari dagozkion ezaugarriak eskuratzeko. Behin hori eginda, ezaugarriak banan-banan prozesatuko dira erabiltzaileari hauen balioak eskatuz. Noski, erabiltzaileak normalean, kategoriaz gain, oso ezaugarri gutxi definitu nahi izango ditu, eta beraz, ez da beharrezkoa ezaugarri guztiak definitzea. Algoritmoa jarraian ikus daiteke:

```
1 funtzioa kontsulta_definitu(schema_path, maila)
2   schema = FSSchema(schema_path, maila);
3   /* Eskatu erabiltzaileari lema eta forma. */
```

```
4  lema = sortu_lema_form();
5  forma = sortu_forma_form();
6  kontsulta_balioak.add("lema", lema);
7  kontsulta_balioak.add("forma", forma);
8  /* Lortu kategoria balio posibleak eta eskatu erabiltzaileari bat
   aukeratzeko. */
9  kategoria = sortu_kategoria_form(schema->getKategoriak());
10 kontsulta_balioak.add("kategoria", kategoria);
11 /* Kategoriaren arabera, eskematik dagozkion ezaugarriak lortu eta banan-
   banan erabiltzaileari balioak eskatu. */
12 ezaugarriak = schema->getEzaugarriak(kategoria);
13 bakoitzeko ezaugarriak as ezaugarri egin
14     balio = sortu_form(ezaugarri);
15     kontsulta_balioak.add(ezaugarri->getElementName(), balio);
16 ambakoitzeko;
17 /* Itzuli kontsultaren definizioa osatzen duten eremu eta balioak. */
18 itzuli kontsulta_balioak;
19 amfuntzioa;
```

5 Ondorioak

Oro har, proiektuaren hasieran ezarritako helburuak bete dira.

Batetik, urteetan zehar aldatzen joan den analisi linguistikoaren egitura xehetasunez eta formalki zehaztu eta definitu da. Zenbaitetan, beharra sortu den heinean, behar hori asetuko duten konponketak egin dira analisisien egituran, hasieran definitutako bertsioa aldatzen joan delarik. Horrek, zenbait analisisitan, espero ez ziren salbuespenak agertzea eragin du. Orain, salbuespen horiek behar bezala definituta geratu dira.

Gainera, XML dokumentuen definiziorako lengoia den RelaxNG lengoiaz analisisien definizio-eskema zehatza idatzi da. Horrek ematen dituen abantailak askotarikoak dira. Esaterako, analisisien egitura ezagutu nahi duenak eskema hori kontsulta dezake, xehetasun guztiak kontsultatzeko aukera izango baitu bertan. Bestalde, programek ere interpreta dezakete eskema, RNG lengoia horretarako tresnak eskaintzen baititu.

Azken horrekin lotuta, anotazio linguistiko konplexuak kudeatzeko Armiarma aplikazioari bi hobekuntza esanguratsu egin zaizkio. Bi hobekuntzak analisisien egitura ezagutzearekin lotuta daude, sistemak analisisien definizio-eskemak erabiliko baititu erabiltzaileari laguntza eskaintzeko. Batetik, testu-corpusen gaineko kontsultak egiteko interfazea hobetu da, hitzen ia edozein ezaugarri definitzeko aukera eskaintzen baita orain. Bestetik, analisi berrien eskuzko sorkuntza egiteko tresna diseinatu da, prozesu horretan erabiltzailea zehaztasunez gidatuko duen sistema, alegia.

Egindako lana garrantzitsua iruditzen zaigu Ixa taldearen azpiegitura hobetze aldera, egindako hobekuntzak sistema osoaren oinarrian egin baitira, eta horrek, neurri batean, sistemari berari sendotasuna eman diola baiteritzogu.

6 Etorkizuneko lanak

Etorkizun hurbilari begira, Armiarma aplikazioaren analisisien eskuzko sorkuntzarekin laguntzeko funtzionalitatearen implementazioa osotara bukatzea falta da. Horretarako, analisisien definizio-eskeman eremu bakoitzaren inguruko iruzkin eta etiketak idatzi behar dira, ondoren, interfazean eremu bakoitzaren balioak eskatzean, erabiltzaileak eremua zein den jakin dezan.

Horrez gain, Armiarma aplikazioa bukatzeko zenbait ataza gelditzen dira egiteko. Interfazean hobekuntzak egin behar dira, eta bai funtzionalitatean ere. Dena dela, Armiarmaren inguruan aurrerantzean egin beharreko lanak definitzeko daude oraindik.

7 Eranskinak

7.1 Analisi linguistikoaren ezaugarrien antolaketa eta multzokatzeta

Eranskin honetan goi-mailako ezaugarriak nola multzokatu diren ikus daiteke. Ezaugarriak zerrendatuta adierazten dira analisisetan, bakoitza nondik datorren adierazi gabe. Hone-lan, adibidez, kategoriaren araberrako ezaugarriak eta hiztegi-sarrera izateagatik dagozkion ezaugarriak nahastuta agertzen dira analisisetan. Ezaugarri guztiak multzo logikotan sailkatu dira, eta definizio-eskeman multzo logikotan elkartuta ordenatu dira. Jarraian, kategoriak, zein ezaugarri onartzen dituen eta hauek nola multzokatzen diren ikus daiteke.

Bost multzo desberdin sailkatu dira, eta koloreen bidez bereizi dira jarraian ageri den eskeman:

- **Horia:** Kategoria eta azpikategoriak.
- **Gorria:** Kategoriari dagozkion ezaugarriak.
- **Berdea:** Flexio-ezaugarriak.
- **Urdina:** Hiztegi-sarrera konplexuen (laburdura-siglak, eratorriak eta hitz-elkartuak) ezaugarriak.
- **Grisa:** Morfemetatik jaso ditzaketan ezaugarriak.

ADB

* KAT
* AZP
* ADBM
* KAS
* NUM
* MUG
* FSL
* MTKAT
* OIN
* AURL
* ATZL
* OSA1

* OSA2
* ADOIN [adi]
* ADM [amm]
* MAI [gra]
* BIZ [ize]
* KAS-PLUS

ADI

* KAT
* AZP
* ADOIN
* ASP
* LAGM

* KAS
* NUM
* MUG
* FSL
* MTKAT
* OIN
* AURL
* ATZL
* MDN [adl/adt]
* NOR [adl/adt]
* NORI [adl/adt]
* NOROK [adl/adt]
* HIT [adl/adt]
* ADM [amm]

- * ERL [erl]
- * MAI [gra]
- * MOD [prt]
- * KAS-PLUS

ADJ

- * KAT
- * AZP
- * IZAUR
- * KAS
- * NUM
- * MUG
- * FSL
- * MTKAT
- * OIN
- * AURL
- * ATZL
- * OSA1
- * OSA2
- * ADOIN [adi]
- * LAGM [adi]
- * ADM [amm]
- * BIZ [ize]
- * MAI [gra]
- * KAS-PLUS

ADL

- * KAT
- * MDN
- * NOR
- * NORI
- * NORK
- * HIT

- * KAS
- * NUM
- * MUG
- * FSL
- * ERL [erl]
- * MOD [prt]

ADT

- * KAT
- * MDN
- * NOR
- * NORI
- * NOR
- * HIT
- * ASP
- * KAS
- * NUM
- * MUG
- * FSL
- * ERL [erl]
- * MOD [prt]

BST

- * KAT
- * AZP
- * KAS
- * NUM
- * MUG
- * FSL
- * MTKAT
- * KAS-PLUS

DET

- * KAT
- * AZP
- * NMG
- * KAS
- * NUM
- * MUG
- * FSL
- * OIN
- * ATZL
- * OSA1
- * OSA2
- * MAI [gra]
- * KAS-PLUS

HAOS

- * KAT
- * KAS
- * MUG
- * FSL

IOR

- * KAT
- * AZP
- * PER
- * KAS
- * NUM
- * MUG
- * FSL
- * NMG [det]
- * KAS-PLUS

ITJ

- * KAT
- * AZP
- * KAS

* NUM
 * MUG
 * FSL
 * BIZ [ize]

IZE

* KAT
 * AZP
 * BIZ
 * PLU
 * KAS
 * NUM
 * MUG
 * FSL

* MTKAT
 * OIN
 * AURL
 * ATZL
 * OSA1
 * OSA2
 * ELK
 * ADOIN [adi]
 * LAGM [adi]
 * IZAUR [adj]
 * ADM [amm]
 * NMG [det]
 * MAI [gra]
 * KAS-PLUS

LOT

* KAT
 * AZP
 * ERL
 * KLM
 * FSL
 * MTKAT

PRT

* KAT
 * MOD
 * FSL
 * ERL [erl]

7.2 Analisisien Relax NG definizio-eskemak

Eranskin honetan analisisien definizio-eskema dago ikusgai, RelaxNG lengoiaz idatzita. Eskema bakarra izan arren, sei fitxategitan banatu da. Batetik, burukoa eta gorputza-ren lehenbiziko elementu orokorrak definitzen dituen *main.rng* fitxategia dago. Bertan, analisi bakoitza segmentazio, morfosintaxi edo lematizazio mailakoa izan daitekeela definitu da. Analisi mota horietako bakoitza fitxategi desberdin batean definitu da, beraz, *seg.rng*, *morf.rng* eta *lem.rng* fitxategiak ere baditugu. Morfosintaxi mailako analisisiek segmentazio analisisien atalak berrerabiltzen dituztenez, *include* bat egiten da *seg.rng* fitxategia *morf.rng*-n inportatzeko. Era berean, bi horiek lematizazioan berrerabiltzen direnez, *morf.rng* fitxategia *lem.rng*-n inportatzen da. Ezaugarri guztiak, kategoria eta azpikategoriak barne, *ezaugarriak.rng* fitxategian definitu dira. Azkenik, motak, *motak.rng* fitxategian definitu dira.

```

1 <!-- main.rng -->
2
3 <?xml version="1.0" encoding="UTF-8"?>
4 <grammar xmlns="http://relaxng.org/ns/structure/1.0"
5   xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
6   datatypeLibrary="http://www.w3.org/2001/XMLSchema-2
7   datatypes">
8
9   <include href="len.rng"/>
10
11   <start>
12     <element name="TEI.2">
13       <element name="teiHeader">
14         <element name="fileDesc">
15           <data type="string"/>
16         </element>
17         <element name="publicationStm">
18           <element name="publisher">
19             <data type="string"/>
20           </element>
21         </element>
22         <element name="sourceDesc">
23           <data type="string"/>
24         </element>
25       </element>
26     </element>
27     <zeroOrMore>
28       <element name="fsdDecl">
29         <attribute name="type">
30           <data type="string"/>
31         </attribute>
32         <attribute name="fsd">
33           <data type="string"/>
34         </attribute>
35       </empty/>
36     </element>
37   </zeroOrMore>
38 </element>
39 <element name="profileDesc">
40 <element name="language">
41 <attribute name="id">
42 <value>eu</value>
43 </attribute>
44 </empty/>
45 </element>

```

```

46 </element>
47 <element name="revisionDesc">
48 <data type="string"/>
49 </element>
50 </element>
51 <element name="text">
52 <attribute name="id">
53 <data type="string"/>
54 </attribute>
55 <element name="body">
56 <choice>
57 <!-- Segmentazioa -->
58 <oneOrMore>
59 <element name="p">
60 <ref name="fs.segmentazioa"/>
61 </element>
62 </oneOrMore>
63 <!-- Morfosintaxia -->
64 <oneOrMore>
65 <element name="p">
66 <ref name="fs.morfosintaxia"/>
67 </element>
68 </oneOrMore>
69 <!-- Lematizazioa -->
70 <oneOrMore>
71 <element name="p">
72 <ref name="fs.lematizazioa"/>
73 </element>
74 </oneOrMore>
75 </choice>
76 </element>
77 </element>
78 </start>
79 </grammar>
80
81 <!-- seg.rng -->
82
83 <?xml version="1.0" encoding="UTF-8"?>
84 <grammar xmlns="http://relaxng.org/ns/structure/1.0"

```

```

5  xmlns:as="http://relaxng.org/ns/compatibility/annotations"
6  /1.0"
7  datatypeLibrary="http://www.w3.org/2001/XMLSchema-
8  datatypes"
9
10 <include href="ezaugarriak.rng" />
11
12 <!-- Segmentazio-analisiaren definizioa -->
13
14 <define name="fs.segmentazio"
15 <element name="fs"
16 <attribute name="id"
17 <data type="string" />
18 </attribute>
19 <attribute name="type"
20 <value>segmentazio</value>
21 </attribute>
22 <ref name="f.forma" />
23 <ref name="f.lemma-morfemak" />
24 </element>
25 </define>
26
27 <define name="f.forma"
28 <element name="f"
29 <attribute name="name"
30 <value>forma</value>
31 </attribute>
32 <ref name="mota.str" />
33 </element>
34 </define>
35
36 <define name="f.lemma-morfemak"
37 <element name="f"
38 <attribute name="name"
39 <value>lemma-morfemak</value>
40 </attribute>
41 <attribute name="org"
42 <value>list</value>
43 </attribute>
44 <oneOrMore
45 <choice
46 <ref name="fs.lemmaSeg" />
47 <ref name="fs.morfSeg" />
48 </choice>
49 </oneOrMore>
50 </element>
</define>

```

```

51 <define name="fs.lemmaSeg"
52 <element name="fs"
53 <attribute name="type"
54 <value>lemmaSeg</value>
55 </attribute>
56 <ref name="f.sarrera" />
57 <ref name="f.ezaugarriak" />
58 </element>
59 <ref name="f.aldaera" />
60 </optional>
61 <ref name="f.twol" />
62 </element>
63 </define>
64
65 <define name="fs.morfSeg"
66 <element name="fs"
67 <attribute name="type"
68 <value>morfSeg</value>
69 </attribute>
70 <ref name="f.sarrera" />
71 <ref name="f.ezaugarriak" />
72 </optional>
73 <ref name="f.aldaera" />
74 </optional>
75 <ref name="f.twol" />
76 </element>
77 </define>
78
79 <define name="f.sarrera"
80 <element name="f"
81 <attribute name="name"
82 <value>sarrera</value>
83 </attribute>
84 <ref name="mota.fs.gako" />
85 </element>
86 </define>
87
88 <define name="f.ezaugarriak"
89 <element name="f"
90 <attribute name="name"
91 <value>ezaugarriak</value>
92 </attribute>
93 <ref name="fs.ezaugarri-lista" />
94 </element>
95 </define>
96
97 <define name="f.aldaera"
98 <element name="f"

```

99			
100	<attribute name="name">	<ref name="MAR-kategoria-eta-ezaugarriak"/>	
101	<value>aldaera</value>	<ref name="PRT-kategoria-eta-ezaugarriak"/>	
102	</attribute>	</choice>	
103	<ref name="mota.str"/>	</define>	
104	</element>		
105	</define>		
106	<define name="f.twol">	<define name="ADB-kategoria-eta-ezaugarriak">	
107	<element name="f">	<!-- Kategoria -->	
108	<attribute name="name">	<ref name="f.kat.adb"/>	
109	<value>twol</value>	<!-- Azpikategoria -->	
110	</attribute>	<ref name="f.azpkat.adb"/>	
111	<ref name="mota.str"/>	<!-- Ezaugarriak -->	
112	</element>	<ref name="ADB-ezaugarriak"/>	
113	</define>	<ref name="flexio-ezaugarriak"/>	
114		<ref name="funtzio-sintaktikoak"/>	
115	<define name="fs.ezaugarri-lista">	<ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>	
116	<element name="fs">	</define>	
117	<attribute name="type">	<define name="ADI-kategoria-eta-ezaugarriak">	
118	<value>ezaugarri-lista</value>	<!-- Kategoria -->	
119	</attribute>	<ref name="f.kat.adi"/>	
120	<ref name="kategoria-ezaugarriak"/>	<!-- Azpikategoria -->	
121	<ref name="EDBL-ezaugarri-hautazkoak"/>	<ref name="f.azpkat.adi"/>	
122	</element>	<!-- Ezaugarriak -->	
123	</define>	<ref name="ADI-ezaugarriak"/>	
124		<ref name="flexio-ezaugarriak"/>	
125	<define name="kategoria-ezaugarriak">	<ref name="funtzio-sintaktikoak"/>	
126	<choice>	<ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>	
127	<ref name="ADB-kategoria-eta-ezaugarriak"/>	</define>	
128	<ref name="ADI-kategoria-eta-ezaugarriak"/>	<define name="ADJ-kategoria-eta-ezaugarriak">	
129	<ref name="ADJ-kategoria-eta-ezaugarriak"/>	<!-- Kategoria -->	
130	<ref name="ADL-kategoria-eta-ezaugarriak"/>	<ref name="f.kat.adj"/>	
131	<ref name="ADT-kategoria-eta-ezaugarriak"/>	<!-- Azpikategoria -->	
132	<ref name="AMM-kategoria-eta-ezaugarriak"/>	<ref name="f.azpkat.adj"/>	
133	<ref name="ASP-kategoria-eta-ezaugarriak"/>	<!-- Ezaugarriak -->	
134	<ref name="ATZ-kategoria-eta-ezaugarriak"/>	<ref name="ADJ-ezaugarriak"/>	
135	<ref name="AUR-kategoria-eta-ezaugarriak"/>	<ref name="flexio-ezaugarriak"/>	
136	<ref name="BST-kategoria-eta-ezaugarriak"/>	<ref name="funtzio-sintaktikoak"/>	
137	<ref name="DEK-kategoria-eta-ezaugarriak"/>	<ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>	
138	<ref name="DEL-kategoria-eta-ezaugarriak"/>	</define>	
139	<ref name="ELI-kategoria-eta-ezaugarriak"/>	<define name="ADL-kategoria-eta-ezaugarriak">	
140	<ref name="ERL-kategoria-eta-ezaugarriak"/>	<!-- Kategoria -->	
141	<ref name="GRA-kategoria-eta-ezaugarriak"/>	<ref name="f.kat.adl"/>	
142	<ref name="HAOS-kategoria-eta-ezaugarriak"/>	<!-- Ezaugarriak -->	
143	<ref name="IOR-kategoria-eta-ezaugarriak"/>	<ref name="ADL-ezaugarriak"/>	
144	<ref name="ITU-kategoria-eta-ezaugarriak"/>	<ref name="ADL-ezaugarriak"/>	
145	<ref name="IZE-kategoria-eta-ezaugarriak"/>	<ref name="funtzio-sintaktikoak"/>	
146	<ref name="LOT-kategoria-eta-ezaugarriak"/>	</define>	

195		
196		
197		
198		
199		
200		
201		
202		
203		
204		
205		
206		
207		
208		
209		
210		
211		
212		
213		
214		
215		
216		
217		
218		
219		
220		
221		
222		
223		
224		
225		
226		
227		
228		
229		
230		
231		
232		
233		
234		
235		
236		
237		
238		
239		
240		
241		
242		
243		
244		
245		
246		
247		
248		
249		
250		
251		
252		
253		
254		
255		
256		
257		
258		
259		
260		
261		
262		
263		
264		
265		
266		
267		
268		
269		
270		
271		
272		
273		
274		
275		
276		
277		
278		
279		
280		
281		
282		
283		
284		
285		
286		
287		
288		
289		
290		
195		
196		
197		
198		
199		
200		
201		
202		
203		
204		
205		
206		
207		
208		
209		
210		
211		
212		
213		
214		
215		
216		
217		
218		
219		
220		
221		
222		
223		
224		
225		
226		
227		
228		
229		
230		
231		
232		
233		
234		
235		
236		
237		
238		
239		
240		
241		
242		
243		
244		
245		
246		
247		
248		
249		
250		
251		
252		
253		
254		
255		
256		
257		
258		
259		
260		
261		
262		
263		
264		
265		
266		
267		
268		
269		
270		
271		
272		
273		
274		
275		
276		
277		
278		
279		
280		
281		
282		
283		
284		
285		
286		
287		
288		
289		
290		

HAP masterra

```

291 <define name="IOR-kategoria-eta-ezaugarriak">
292 <!-- Kategoria -->
293 <ref name="f.kat.ior"/>
294 <!-- Azpikategoria -->
295 <ref name="f.azpkat.ior"/>
296 <!-- Ezaugarriak -->
297 <ref name="IOR-ezaugarriak"/>
298 <ref name="flexio-ezaugarriak"/>
299 <ref name="funtzio-sintaktikoak"/>
300 <ref name="funtzio-sintaktikoak"/>
301 <ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>
302 </define>
303
304 <define name="ITJ-kategoria-eta-ezaugarriak">
305 <!-- Kategoria -->
306 <ref name="f.kat.itj"/>
307 <!-- Ezaugarriak -->
308 <ref name="ITJ-ezaugarriak"/>
309 <ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>
310 </define>
311
312 <define name="IZE-kategoria-eta-ezaugarriak">
313 <!-- Kategoria: IZE -->
314 <ref name="f.kat.ize"/>
315 <!-- Azpikategoria -->
316 <ref name="f.azpkat.ize"/>
317 <!-- Ezaugarriak -->
318 <ref name="IZE-ezaugarriak"/>
319 <ref name="flexio-ezaugarriak"/>
320 <ref name="funtzio-sintaktikoak"/>
321 <ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>
322 </define>
323
324 <define name="LOT-kategoria-eta-ezaugarriak">
325 <!-- Kategoria -->
326 <ref name="f.kat.lot"/>
327 <!-- Azpikategoria -->
328 <ref name="f.azpkat.lot"/>
329 <!-- Ezaugarriak -->
330 <ref name="LOT-ezaugarriak"/>
331 <ref name="f.funtzio-sintaktikoak"/>
332 <ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>
333 </define>
334
335 <define name="MAR-kategoria-eta-ezaugarriak">
336 <!-- Kategoria -->
337 <ref name="f.kat.mar"/>
338 <!-- Ezaugarriak -->
339 <ref name="MAR-ezaugarriak"/>
340 </define>
341
342 <define name="PRT-kategoria-eta-ezaugarriak">
343 <!-- Kategoria -->
344 <ref name="f.kat.prt"/>
345 <!-- Ezaugarriak -->
346 <ref name="PRT-ezaugarriak"/>
347 <ref name="hiztegi-sarrera-konplexuen-ezaugarriak"/>
348 </define>
349
350
351 <!-- Kategorien arabera ezaugarriak -->
352
353 <define name="ADB-ezaugarriak">
354 <optional>
355 <ref name="f.adberbio-mota"/>
356 </optional>
357 </define>
358
359 <define name="ADI-ezaugarriak">
360 <ref name="f.aditzoina"/>
361 <optional>
362 <ref name="f.aspektua"/>
363 </optional>
364 <optional>
365 <ref name="f.laguntzaile-mota"/>
366 </optional>
367 </define>
368
369 <define name="ADJ-ezaugarriak">
370 <optional>
371 <ref name="f.izenaren-aurretik"/>
372 </optional>
373 </define>
374
375 <define name="ADL-ezaugarriak">
376 <ref name="adizki-ezaugarriak"/>
377 </define>
378
379 <define name="ADT-ezaugarriak">
380 <ref name="adizki-ezaugarriak"/>
381 <ref name="f.aspektua"/>
382 </define>
383
384 <define name="AMM-ezaugarriak">
385 <ref name="f.aditz-mota"/>
386 </define>

```

HAP masterra

```

387 <define name="ASP-ezaugarriak">
388 <ref name="f.aspektua"/>
389 </define>
390
391 <define name="ATZ-ezaugarriak">
392 <ref name="f.eratorriaren-kategoria"/>
393 <ref name="f.eratorriaren-azpikategoria"/>
394 </define>
395
396
397 <define name="AUR-ezaugarriak">
398 <empty/>
399 </define>
400
401 <define name="BST-ezaugarriak">
402 <optional>
403 <ref name="f.metakategoria"/>
404 </optional>
405 <optional>
406 <ref name="f.adierazia"/>
407 </optional>
408 </define>
409
410 <define name="DEK-ezaugarriak">
411 <empty/>
412 </define>
413
414 <define name="DET-ezaugarriak">
415 <optional>
416 <ref name="f.numeroa-mugatasuna"/>
417 </optional>
418 </define>
419
420 <define name="ELI-ezaugarriak">
421 <empty/>
422 </define>
423
424 <define name="ERL-ezaugarriak">
425 <ref name="f.erlazioa"/>
426 </define>
427
428 <define name="GRA-ezaugarriak">
429 <ref name="f.gradu-maila"/>
430 </define>
431
432 <define name="HAOS-ezaugarriak">
433 <empty/>
434 </define>
435
436 <define name="IOR-ezaugarriak">
437 <optional>
438 <ref name="f.numeroa"/>
439 </optional>
440 <optional>
441 <ref name="f.pertsona"/>
442 </optional>
443 </define>
444
445 <define name="ITJ-ezaugarriak">
446 <empty/>
447 </define>
448
449 <define name="IZE-ezaugarriak">
450 <optional>
451 <ref name="f.plurala"/>
452 </optional>
453 <optional>
454 <ref name="f.biziduna"/>
455 </optional>
456 <optional>
457 <ref name="f.neurgarria"/>
458 </optional>
459 <optional>
460 <ref name="f.zenbakarria"/>
461 </optional>
462 </define>
463
464 <define name="LOT-ezaugarriak">
465 <ref name="f.erlazioa"/>
466 <optional>
467 <ref name="f.klausula-muga"/>
468 </optional>
469 </define>
470
471 <define name="MAR-ezaugarriak">
472 <empty/>
473 </define>
474
475 <define name="PRT-ezaugarriak">
476 <ref name="f.modalitatea"/>
477 </define>
478
479 <define name="adizki-ezaugarriak">
480 <optional>
481 <ref name="f.erroa"/>
482 </optional>

```

HAP masterra

```

483 <ref name="f.modua-denbora" />
484 <ref name="f.abolutiboa" />
485 <optional>
486 <ref name="f.datiboa" />
487 </optional>
488 <optional>
489 <ref name="f.ergatiboa" />
490 </optional>
491 <optional>
492 <ref name="f.hitanoa" />
493 </optional>
494 </define>
495 </grammar>
496
    
```

```

29 <ref name="f.oina" />
30 </optional>
31 <optional>
32 <ref name="f.osagaiak" />
33 </optional>
34 </element>
35 </define>
36
37 <define name="f. lema-osatua">
38 <element name="f">
39 <attribute name="name">
40 <value>lema-osatua</value>
41 </attribute>
42 <ref name="mota.str" />
43 </element>
44 </define>
45
46 <define name="f.goimailako-ezaugarriak">
47 <element name="f">
48 <attribute name="name">
49 <value>goimailako-ezaugarriak</value>
50 </attribute>
51 <ref name="fs.goimailako-ezaugarri-lista" />
52 </element>
53 </define>
54
55 <define name="f.osagaiak">
56 <element name="f">
57 <attribute name="name">
58 <value>osagaiak</value>
59 </attribute>
60 <attribute name="org">
61 <value>list</value>
62 </attribute>
63 <oneOrMore>
64 <ref name="fs.osagaia" />
65 </oneOrMore>
66 </element>
67 </define>
68
69 <define name="fs.osagaia">
70 <element name="fs">
71 <attribute name="type">
72 <value>osagaia</value>
73 </attribute>
74 <optional>
75 <ref name="f.oina" />
76 </optional>
    
```

```

1 <!-- morf.rng -->
2
3 <?xml version="1.0" encoding="UTF-8"?>
4 <grammar xmlns="http://relaxng.org/ns/structure/1.0"
5   xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
6   datatypeLibrary="http://www.w3.org/2001/XMLSchema-1
7   datatypes">
8
9   <include href="seg.rng" />
10
11   <!-- Morfosintaxi-analisiaren definizioa -->
12
13   <define name="fs.morfosintaxia">
14     <element name="fs">
15       <attribute name="id">
16         <data type="string" />
17       </attribute>
18       <attribute name="type">
19         <value>analisi</value>
20       </attribute>
21       <optional>
22         <ref name="f.lema-osatua" />
23       </optional>
24       <optional>
25         <ref name="f.aidaera-osatua" />
26       </optional>
27       <ref name="f.forma" />
28       <ref name="f.goimailako-ezaugarriak" />
29     </element>
    
```


77	</optional>	125	<define name="fs. lema">
78	<ref name="f.morfemak"/>	126	<element name="fs">
79	</optional>	127	<attribute name="type">
80	</optional>	128	<value>lema</value>
81	<ref name="f.ezaugarriak-morf"/>	129	</attribute>
82	</optional>	130	<ref name="f.ezaugarriak-morf"/>
83	</element>	131	</optional>
84	</define>	132	<ref name="f.twol"/>
85		133	<ref name="f.sarrera"/>
86	<define name="f.morfemak">	134	</optional>
87	<element name="f">	135	</optional>
88	<attribute name="name">	136	<ref name="f.elkarketa"/>
89	<value>morfemak</value>	137	</optional>
90	</attribute>	138	</element>
91	<attribute name="org">	139	
92	<value>list</value>	140	
93	</attribute>	141	<define name="f.elkarketa">
94	<oneOrMore>	142	<element name="f">
95	<element name="fs">	143	<attribute name="name">
96	<attribute name="type">	144	<value>elkarketa</value>
97	<value>morfema</value>	145	</attribute>
98	</attribute>	146	<element name="fs">
99	<ref name="f.ezaugarriak-morf"/>	147	<attribute name="type">
100	<ref name="f.twol"/>	148	<value>elkarketa</value>
101	<ref name="f.sarrera"/>	149	</attribute>
102	</element>	150	<element name="f">
103	</oneOrMore>	151	<attribute name="name">
104	</element>	152	<value>mugakizuna</value>
105	</define>	153	</attribute>
106		154	<ref name="fs. lema"/>
107	<define name="f.aldaera-osatua">	155	</element>
108	<element name="f">	156	<element name="f">
109	<attribute name="name">	157	<attribute name="name">
110	<value>aldaera-osatua</value>	158	<value>mugatzailea</value>
111	</attribute>	159	</attribute>
112	<ref name="mota.str"/>	160	<ref name="fs. lema"/>
113	</element>	161	</element>
114	</define>	162	</element>
115		163	</define>
116	<define name="f.oina">	164	
117	<element name="f">	165	
118	<attribute name="name">	166	<define name="fs.goimailako-ezaugarri-lista">
119	<value>oina</value>	167	<element name="fs">
120	</attribute>	168	<attribute name="type">
121	<ref name="fs. lema"/>	169	<value>goimailako-ezaugarri-lista</value>
122	</element>	170	</attribute>
123	</define>	171	<ref name="goimailako-kategoria-ezaugarriak"/>
124		172	</optional>

173			
174			
175			
176			
177			
178			
179			
180			
181			
182			
183			
184			
185			
186			
187			
188			
189			
190			
191			
192			
193			
194			
195			
196			
197			
198			
199			
200			
201			
202			
203			
204			
205			
206			
207			
208			
209			
210			
211			
212			
213			
214			
215			
216			
217			
218			
219			
220			
221			
222			
223			
224			
225			
226			
227			
228			
229			
230			
231			
232			
233			
234			
235			
236			
237			
238			
239			
240			
241			
242			
243			
244			
245			
246			
247			
248			
249			
250			
251			
252			

253	</define>	301	<!-- Ezaugarriak -->
254	<define name="ADT-kategoria-eta-goimailako-ezaugarriak" >	302	<ref name="DET-ezaugarriak" />
255	<!-- Kategoria -->	303	<ref name="flexio-ezaugarriak" />
256	<ref name="f.kat-osatua.adt" />	304	<ref name="funtzio-sintaktikoak" />
257	<!-- Ezaugarriak -->	305	<ref name="hiztegi-sarrerak-konplexuen-ezaugarriak" />
258	<ref name="ADT-ezaugarriak" />	306	<ref name="morfemetatik-datozen-ezaugarriak-DET" />
259	<ref name="flexio-ezaugarriak" />	307	</define>
260	<ref name="funtzio-sintaktikoak" />	308	<define name="ELI-kategoria-eta-goimailako-ezaugarriak" >
261	<ref name="morfemetatik-datozen-ezaugarriak-ADT" />	309	<!-- Kategoria -->
262	</define>	310	<ref name="f.kat-osatua.eli" />
263	<define name="adizki-goimailako-ezaugarriak" >	311	<!-- Ezaugarriak -->
264	<ref name="f.modua-denbora" />	312	<ref name="flexio-ezaugarriak" />
265	<ref name="f.absolutiboa" />	313	<ref name="funtzio-sintaktikoak" />
266	<optional>	314	</define>
267	</optional>	315	<define name="ERL-kategoria-eta-goimailako-ezaugarriak" >
268	<ref name="f.datiboa" />	316	<!-- Kategoria -->
269	</optional>	317	<ref name="f.kat-osatua.erl" />
270	</optional>	318	<!-- Ezaugarriak -->
271	<ref name="f.ergatiboa" />	319	<ref name="flexio-ezaugarriak" />
272	</optional>	320	<ref name="funtzio-sintaktikoak" />
273	</optional>	321	</define>
274	<ref name="f.hitanoa" />	322	<define name="HAOS-kategoria-eta-goimailako-ezaugarriak" >
275	</optional>	323	<!-- Kategoria -->
276	</define>	324	<ref name="f.kat-osatua.haos" />
277	<define name="BST-kategoria-eta-goimailako-ezaugarriak" >	325	<!-- Ezaugarriak -->
278	<!-- Kategoria -->	326	<ref name="flexio-ezaugarriak" />
279	<ref name="f.kat-osatua.bst" />	327	<ref name="HAOS-ezaugarriak" />
280	</optional>	328	<ref name="flexio-ezaugarriak" />
281	</optional>	329	<ref name="funtzio-sintaktikoak" />
282	<ref name="hiztegi-sarrerak-konplexuen-ezaugarriak" />	330	<ref name="morfemetatik-datozen-ezaugarriak-BST" />
283	<ref name="morfemetatik-datozen-ezaugarriak-BST" />	331	</define>
284	</define>	332	<define name="IOR-kategoria-eta-goimailako-ezaugarriak" >
285	<!-- Ezaugarriak -->	333	<!-- Kategoria -->
286	<ref name="BST-ezaugarriak" />	334	<ref name="f.kat-osatua.ior" />
287	<ref name="flexio-ezaugarriak" />	335	<!-- Kategoria -->
288	<ref name="funtzio-sintaktikoak" />	336	<ref name="f.kat-osatua.ior" />
289	<ref name="hiztegi-sarrerak-konplexuen-ezaugarriak" />	337	<!-- Azpikategoria -->
290	<ref name="morfemetatik-datozen-ezaugarriak-BST" />	338	<optional>
291	</define>	339	<ref name="f.azpkat.ior" />
292	<define name="DET-kategoria-eta-goimailako-ezaugarriak" >	340	</optional>
293	<!-- Kategoria -->	341	<!-- Ezaugarriak -->
294	<ref name="f.kat-osatua.det" />	342	<ref name="IOR-ezaugarriak" />
295	<!-- Kategoria -->	343	<ref name="flexio-ezaugarriak" />
296	<ref name="f.kat-osatua.det" />	344	<ref name="funtzio-sintaktikoak" />
297	<!-- Azpikategoria -->	345	<ref name="morfemetatik-datozen-ezaugarriak-IOR" />
298	<optional>	346	</define>
299	<ref name="f.azpkat.det" />	347	
300	</optional>		

348	<define name="ITUJ-kategoria-eta-goimailako-ezaugarriak" >	396	<define name="morfemetatik-datzen-ezaugarriak-ADB" >
349	<!-- Kategoria -->	397	<optional>
350	<ref name="f.kat-osatua.itj"/>	398	<ref name="f.aditzoia"/>
351	<!-- Ezaugarriak -->	399	</optional>
352	<ref name="ITUJ-ezaugarriak"/>	400	<optional>
353	<ref name="flexio-ezaugarriak"/>	401	<ref name="f.aditz-mota"/>
354	<ref name="funtzio-sintaktikoak"/>	402	</optional>
355	<ref name="morfemetatik-datzen-ezaugarriak-ITUJ"/>	403	<optional>
356	</define>	404	<ref name="f.gradu-maila"/>
357		405	</optional>
358	<define name="IZE-kategoria-eta-goimailako-ezaugarriak" >	406	<optional>
359	<!-- Kategoria: IZE -->	407	<ref name="f.biziduna"/>
360	<ref name="f.kat-osatua.ize"/>	408	</optional>
361	<!-- Azpikategoria -->	409	<optional>
362	<optional>	410	<ref name="f.kasua-plus"/>
363	<ref name="f.azpkat.ize"/>	411	</optional>
364	</optional>	412	</define>
365	<!-- Ezaugarriak -->	413	
366	<ref name="IZE-ezaugarriak"/>	414	<define name="morfemetatik-datzen-ezaugarriak-ADI" >
367	<ref name="flexio-ezaugarriak"/>	415	<optional>
368	<ref name="funtzio-sintaktikoak"/>	416	<ref name="f.modua-denbora"/>
369	<ref name="hiztegi-sarrera-komplexuen-ezaugarriak"/>	417	</optional>
370	<ref name="morfemetatik-datzen-ezaugarriak-IZE"/>	418	<optional>
371	</define>	419	<ref name="f.absoiutiboa"/>
372		420	</optional>
373	<define name="LOT-kategoria-eta-goimailako-ezaugarriak" >	421	<optional>
374	<!-- Kategoria -->	422	<ref name="f.datiboa"/>
375	<ref name="f.kat-osatua.lot"/>	423	</optional>
376	<!-- Azpikategoria -->	424	<optional>
377	<ref name="f.azpkat.lot"/>	425	<ref name="f.ergatiboa"/>
378	<!-- Ezaugarriak -->	426	</optional>
379	<ref name="LOT-ezaugarriak"/>	427	<optional>
380	<ref name="funtzio-sintaktikoak"/>	428	<ref name="f.hitanoa"/>
381	</define>	429	</optional>
382		430	<optional>
383	<define name="PRT-kategoria-eta-goimailako-ezaugarriak" >	431	<ref name="f.aurrizkia"/>
384	<!-- Kategoria -->	432	</optional>
385	<ref name="f.kat-osatua.prt"/>	433	<optional>
386	<!-- Ezaugarriak -->	434	<ref name="f.aditz-mota"/>
387	<ref name="PRT-ezaugarriak"/>	435	</optional>
388	<ref name="flexio-ezaugarriak"/>	436	<optional>
389	<ref name="funtzio-sintaktikoak"/>	437	<ref name="f.erlazioa"/>
390	<ref name="morfemetatik-datzen-ezaugarriak-PRT"/>	438	</optional>
391	</define>	439	<optional>
392		440	<ref name="f.gradu-maila"/>
393		441	</optional>
394	<!-- Morfemetatik datzen ezaugarriak -->	442	<optional>
395		443	<ref name="f.modalitatea"/>

444	</optional>	492	</optional>
445	<optional>	493	</define>
446	<ref name="f.kasua-plus" />	494	<define name="morfemetatik-datozen-ezaugarriak-DET">
447	</optional>	495	<optional>
448	</define>	496	<ref name="f.gradu-maila" />
449		497	</optional>
450	<define name="morfemetatik-datozen-ezaugarriak-ADJ">	498	</optional>
451	<optional>	499	<optional>
452	<ref name="f.aditzoina" />	500	<ref name="f.kasua-plus" />
453	</optional>	501	</optional>
454	<optional>	502	</define>
455	<ref name="f.laguntzaile-mota" />	503	<define name="morfemetatik-datozen-ezaugarriak-IOR">
456	</optional>	504	<optional>
457	<optional>	505	<ref name="f.numeroa-mugatasuna" />
458	<ref name="f.aditz-mota" />	506	</optional>
459	</optional>	507	</optional>
460	<optional>	508	<optional>
461	<ref name="f.biziduna" />	509	<ref name="f.kasua-plus" />
462	</optional>	510	</optional>
463	<optional>	511	</define>
464	<ref name="f.gradu-maila" />	512	<define name="morfemetatik-datozen-ezaugarriak-ITJ">
465	</optional>	513	<optional>
466	<optional>	514	<optional>
467	<ref name="f.kasua-plus" />	515	<ref name="f.biziduna" />
468	</optional>	516	</optional>
469	</define>	517	</define>
470		518	
471	<define name="morfemetatik-datozen-ezaugarriak-ADL">	519	<define name="morfemetatik-datozen-ezaugarriak-IZE">
472	<optional>	520	<optional>
473	<ref name="f.erlazioa" />	521	<ref name="f.aditzoina" />
474	</optional>	522	</optional>
475	<optional>	523	<optional>
476	<ref name="f.modalitatea" />	524	<ref name="f.laguntzaile-mota" />
477	</optional>	525	</optional>
478	</define>	526	<optional>
479		527	<ref name="f.izenaren-aurretik" />
480	<define name="morfemetatik-datozen-ezaugarriak-ADT">	528	</optional>
481	<optional>	529	<optional>
482	<ref name="f.erlazioa" />	530	<ref name="f.aditz-mota" />
483	</optional>	531	</optional>
484	<optional>	532	<optional>
485	<ref name="f.modalitatea" />	533	<ref name="f.numeroa-mugatasuna" />
486	</optional>	534	</optional>
487	</define>	535	<optional>
488		536	<ref name="f.gradu-maila" />
489	<define name="morfemetatik-datozen-ezaugarriak-BST">	537	</optional>
490	<optional>	538	<optional>
491	<ref name="f.kasua-plus" />	539	<ref name="f.kasua-plus" />

540	</optional>	587	<ref name="HAOS-kategoria-eta-ezaugarriak"/>
541	</define>	588	<ref name="IOR-kategoria-eta-ezaugarriak"/>
542	<define name="morfemetatik-datozen-ezaugarriak-PRT">	589	<ref name="ITJ-kategoria-eta-ezaugarriak"/>
543	<optional>	590	<ref name="IZE-kategoria-eta-ezaugarriak-morf"/>
544	<ref name="f.erlazioa"/>	591	<ref name="LOT-kategoria-eta-ezaugarriak"/>
545	</optional>	592	<ref name="MAR-kategoria-eta-ezaugarriak"/>
546	</define>	593	<ref name="PRT-kategoria-eta-ezaugarriak"/>
547		594	</choice>
548		595	</define>
549		596	
550	<!-- Ezaugarriak, morf mailan sortzen diren aldatetekin -->	597	<define name="ADI-kategoria-eta-ezaugarriak-morf">
551		598	<ref name="ADI-kategoria-eta-ezaugarriak"/>
552	<define name="f.ezaugarriak-morf">	599	<optional>
553	<element name="f">	600	<ref name="f.aditz-mota"/>
554	<attribute name="name">	601	</optional>
555	<value>ezaugarriak</value>	602	</define>
556	</attribute>	603	
557	<element name="fs">	604	<define name="ADJ-kategoria-eta-ezaugarriak-morf">
558	<attribute name="type">	605	<ref name="ADJ-kategoria-eta-ezaugarriak"/>
559	<value>ezaugarri-lista</value>	606	<optional>
560	</attribute>	607	<ref name="f.gradu-maila"/>
561	<ref name="kategoria-ezaugarriak-morf"/>	608	</optional>
562	<ref name="EDBL-ezaugarri-hautazkoak"/>	609	</define>
563	<optional>	610	<define name="ADL-kategoria-eta-ezaugarriak-morf">
564	<ref name="f.kasua-plus"/>	611	<ref name="ADL-kategoria-eta-ezaugarriak"/>
565	</optional>	612	<optional>
566	</element>	613	<ref name="f.erlazioa"/>
567	</define>	614	</optional>
568		615	</define>
569		616	
570	<define name="kategoria-ezaugarriak-morf">	617	
571	<choice>	618	<define name="ADT-kategoria-eta-ezaugarriak-morf">
572	<ref name="ADB-kategoria-eta-ezaugarriak"/>	619	<ref name="ADT-kategoria-eta-ezaugarriak"/>
573	<ref name="ADI-kategoria-eta-ezaugarriak-morf"/>	620	<optional>
574	<ref name="ADJ-kategoria-eta-ezaugarriak-morf"/>	621	<ref name="f.erlazioa"/>
575	<ref name="ADL-kategoria-eta-ezaugarriak-morf"/>	622	</optional>
576	<ref name="ADT-kategoria-eta-ezaugarriak-morf"/>	623	</define>
577	<ref name="AMM-kategoria-eta-ezaugarriak"/>	624	
578	<ref name="ASP-kategoria-eta-ezaugarriak"/>	625	<define name="IZE-kategoria-eta-ezaugarriak-morf">
579	<ref name="ATZ-kategoria-eta-ezaugarriak"/>	626	<ref name="IZE-kategoria-eta-ezaugarriak"/>
580	<ref name="AUR-kategoria-eta-ezaugarriak"/>	627	<optional>
581	<ref name="BST-kategoria-eta-ezaugarriak"/>	628	<ref name="f.aditzoina"/>
582	<ref name="DEK-kategoria-eta-ezaugarriak"/>	629	</optional>
583	<ref name="DEP-kategoria-eta-ezaugarriak"/>	630	</define>
584	<ref name="ELI-kategoria-eta-ezaugarriak"/>	631	
585	<ref name="ERL-kategoria-eta-ezaugarriak"/>	632	</grammar>
586	<ref name="GRA-kategoria-eta-ezaugarriak"/>		

```

1 <!-- lem.rng -->
2
3 <?xml version="1.0" encoding="UTF-8"?>
4 <grammar
5   xmlns="http://relaxng.org/ns/structure/1.0"
6   xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
7   datatypeLibrary="http://www.w3.org/2001/XMLSchema-
8     ↳ datatypes">
9
10  <include href="morf.rng"/>
11
12  <!-- Lematizazio-analisiaren definizioa -->
13
14  <define name="fs.lematizazioa">
15    <element name="fs">
16      <data type="string"/>
17    </attribute>
18    <attribute name="type">
19      <value>lematizazioa</value>
20    </attribute>
21    <optional>
22      <ref name="f. lema-osatua"/>
23    </optional>
24    <optional>
25      <ref name="f. aldaera-osatua"/>
26    </optional>
27    <ref name="f. forma"/>
28    <ref name="f. ezaugarri-morfologikoak"/>
29    <optional>
30      <ref name="f. osagaiak-lem"/>
31    </optional>
32    </element>
33  </define>
34
35  <define name="f. ezaugarri-morfologikoak">
36    <element name="f">
37      <attribute name="name">
38        <value>ezaugarri-morfologikoak</value>
39      </attribute>
40      <ref name="fs.goimailako-ezaugarri-lista"/>
41    </element>
42  </define>
43
44  <define name="f. osagaiak-lem">
45    <element name="f">

```

```

46    <attribute name="name">
47      <value>osagaiak</value>
48    </attribute>
49    <attribute name="org">
50      <value>list</value>
51    </attribute>
52    <oneOrMore>
53      <ref name="fs.osagaia-lem"/>
54    </oneOrMore>
55  </element>
56 </define>
57
58 <define name="fs.osagaia-lem">
59   <element name="fs">
60     <attribute name="type">
61       <value>osagaia</value>
62     </attribute>
63     <optional>
64       <ref name="f. oina-lem"/>
65     </optional>
66     <ref name="f. ezaugarri-morfologikoak"/>
67   </element>
68 </define>
69
70 <define name="f. oina-lem">
71   <element name="f">
72     <attribute name="name">
73       <value>oina</value>
74     </attribute>
75     <ref name="mota.str"/>
76   </element>
77 </define>
78
79 </grammar>

```

```

1 <!-- ezaugarriak.rng -->
2
3 <?xml version="1.0" encoding="UTF-8"?>
4 <grammar xmlns="http://relaxng.org/ns/structure/1.0"
5   xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
6   datatypeLibrary="http://www.w3.org/2001/XMLSchema-
7     ↳ datatypes">
8
9   <include href="motak.rng"/>

```

9	<!-- EDBLko zenbait hautazko ezaugarri: estandarreak, ↵	53	<!-- hiztegi-sarreraren konplexuen ezaugarriak (ADB, ADI, ↵
10	↵ hobetsiak, error-kode eta arrarotasuna -->	54	↵ ADJ, BST, DET, IOR, ITJ, IZE, IOT eta PRT) -->
11		55	
12	<define name="EDBL-ezaugarri-hautazkoak">	56	<define name="hiztegi-sarreraren konplexuen ezaugarriak">
13	<optional>	57	<a:documentation>
14	<choice>	58	Hiztegi sarrera konplexuak dira laburdurak, siglak ↵
15	<ref name="f.estandarrak"/>	59	↵ , hitz
16	<ref name="f.hobetsiak"/>	60	eratorriak eta hitz elkartuak.
17	</choice>	61	</a:documentation>
18	</optional>	62	<ref name="lab-siglen-ezaugarriak"/>
19	<optional>	63	</optional>
20	<ref name="f.error-kode"/>	64	<optional>
21	</optional>	65	<ref name="eratorrien-ezaugarriak"/>
22	<optional>	66	</optional>
23	<ref name="f.arrarotasuna"/>	67	<ref name="hitz-elkartuen-ezaugarriak"/>
24	</optional>	68	</optional>
25	</define>	69	</define>
26		70	
27		71	
28		72	<define name="lab-siglen-ezaugarriak">
29	<!-- Flexio ezaugarriak (DEK, DET, IOR) -->	73	<optional>
30		74	<ref name="f.metakategoria"/>
31	<define name="flexio-ezaugarriak">	75	<optional>
32	<optional>	76	<ref name="f.adierazia"/> <!-- ez da ↵
33	<ref name="f.kasua"/>	77	↵ agertzen MORF analisietan -->
34	</optional>	78	</optional>
35	<optional>	79	</define>
36	<ref name="f.numeroa"/>	80	
37	</optional>	81	<define name="eratorrien-ezaugarriak">
38	<optional>	82	<ref name="f.oinarria"/>
39	<ref name="f.mugatasuna"/>	83	<optional>
40	</optional>	84	<choice>
41	</define>	85	<ref name="f.aurrizkia"/>
42		86	<ref name="f.atzizkiak"/>
43		87	</group>
44	<!-- Funtzio sintaktikoak -->	88	<ref name="f.aurrizkia"/>
45		89	<ref name="f.atzizkiak"/>
46	<define name="funtzio-sintaktikoak">	90	</group>
47	<!-- MORFen, askotan, derrigorrezkoa dela dirudi. ↵	91	</choice>
	↵ Hala ere, salbuespenak daude askotan (akatsak ↵	92	</optional>
	↵ , zihurrenik) eta beraz, optional bezala ↵	93	</define>
	↵ definitu dira balidatu ahal izateko. -->	94	
48	<optional>	95	<define name="hitz-elkartuen-ezaugarriak">
49	<ref name="f.funtzio-sintaktikoak"/>	96	<optional>
50	</optional>	97	<!-- 1. osagaia -->
51	</define>		
52			

HAP masterra

```

98 <element name="f">
99   <attribute name="name">
100     <value>OSA1</value>
101   </attribute>
102   <ref name="mota.fs.gako"/>
103 </element>
104 <!-- 2. osagaia -->
105 <element name="f">
106   <attribute name="name">
107     <value>OSA2</value>
108   </attribute>
109   <ref name="mota.fs.gako"/>
110 </element>
111 </optional>
112 <!-- Elkarketa mota -->
113 <optional>
114   <element name="f">
115     <attribute name="name">
116       <value>ELK</value>
117     </attribute>
118     <element name="sym">
119       <attribute name="value">
120         <choice>
121           <value>I+I</value>
122           <value>IZE+IZE</value>
123           <value>I+ADJ</value>
124           <value>ADI+ADI</value>
125           <value>ADJ+ADJ</value>
126           <value>ADB+ADB</value>
127         </choice>
128       </attribute>
129     </element>
130   </optional>
131 </define>
132
133 <!-- Ezaugarrien definizioak -->
134
135 <define name="f.absolutiboa">
136   <element name="f">
137     <attribute name="name">
138       <value>NOR</value>
139     </attribute>
140     <element name="sym">
141       <attribute name="value">
142         <choice>
143           <value>NI</value>
144         </choice>
145     </element>
146     <value>HI</value>
147     <value>HURA</value>
148     <value>GU</value>
149     <value>ZU</value>
150     <value>ZUEK</value>
151     <value>HAIEK</value>
152   </choice>
153 </attribute>
154 <empty/>
155 </element>
156 </define>
157
158 <define name="f.adberbio-mota">
159   <element name="f">
160     <attribute name="name">
161       <value>ADBM</value>
162     </attribute>
163     <element name="sym">
164       <attribute name="value">
165         <choice>
166           <value>DENB</value>
167           <value>MOD</value>
168           <value>GRAD</value>
169           <value>LEK</value>
170         </choice>
171       </attribute>
172     <empty/>
173   </element>
174 </define>
175
176 <define name="f.adierazia">
177   <element name="f">
178     <attribute name="name">
179       <value>ADZ</value>
180     </attribute>
181     <ref name="mota.fs.gako"/>
182   </element>
183 </define>
184
185 <define name="f.aditzoia">
186   <element name="f">
187     <attribute name="name">
188       <value>ADOIN</value>
189     </attribute>
190     <ref name="mota.str"/>
191   </element>
192 </define>

```

194	</define>	242	</choice>
195		243	</attribute>
196	<define name="f.aditz-mota">	244	<empty/>
197	<element name="f">	245	</element>
198	<attribute name="name">	246	</define>
199	<value>ADM</value>	247	
200	</attribute>	248	<define name="f.atzizkiak">
201	<element name="sym">	249	<element name="f">
202	<attribute name="value">	250	<attribute name="name">
203	<choice>	251	<value>ATZL</value>
204	<value>ADIZE</value>	252	</attribute>
205	<value>ADOIN</value>	253	<attribute name="org">
206	<value>PART</value>	254	<value>list</value>
207	</choice>	255	<oneOrMore>
208	</attribute>	256	</choice>
209	</element>	257	<ref name="mota.fs.gako"/>
210	</define>	258	<ref name="mota.str"/>
211		259	</choice>
212		260	</oneOrMore>
213	<define name="f.arrarotasuna">	261	</element>
214	<element name="f">	262	</define>
215	<attribute name="name">	263	
216	<value>RARE</value>	264	<define name="f.aurrizkia">
217	</attribute>	265	<element name="f">
218	<element name="sym">	266	<attribute name="name">
219	<attribute name="value">	267	<value>AURL</value>
220	<choice>	268	</attribute>
221	<value>ANB</value>	269	</choice>
222	<value>LEX</value>	270	<ref name="mota.fs.gako"/>
223	<value>ABT</value>	271	<ref name="mota.str"/>
224	</choice>	272	</choice>
225	</attribute>	273	</element>
226	</element>	274	</define>
227	</define>	275	
228		276	<define name="f.biziduna">
229		277	<element name="f">
230	<define name="f.aspektua">	278	<attribute name="name">
231	<element name="f">	279	<value>BIZ</value>
232	<attribute name="name">	280	</attribute>
233	<value>ASP</value>	281	<ref name="mota.plusminus"/>
234	</attribute>	282	</element>
235	<element name="sym">	283	
236	<attribute name="value">	284	
237	<choice>	285	
238	<value>GERO</value>	286	
239	<value>BURU</value>	287	<define name="f.datiboa">
240	<value>PNT</value>	288	<element name="f">
241	<value>EZBU</value>	289	<attribute name="name">

HAP masterra

```

338     <value>NORI</value>
339     </attribute>
340     <element name="sym">
341     <attribute name="value">
342     <choice>
343     <value>NIRI</value>
344     <value>HIRI-TO</value>
345     <value>HIRI-NO</value>
346     <value>HARI</value>
347     <value>GURI</value>
348     <value>ZURI</value>
349     <value>ZUEI</value>
350     <value>HAIEI</value>
351     </choice>
352     </attribute>
353     <empty/>
354     </element>
355     </define>
356     </define>
357     <define name="f.eratorriaren-azpikategoria">
358     <element name="f">
359     <attribute name="name">
360     <value>ERAAZP</value>
361     </attribute>
362     <element name="sym">
363     <attribute name="value">
364     <choice>
365     <value>ADK</value>
366     <value>ADP</value>
367     <value>ARR</value>
368     <value>BAN</value>
369     <value>BIH</value>
370     <value>DZG</value>
371     <value>DZH</value>
372     <value>ELK</value>
373     <value>ERKARR</value>
374     <value>ERKIND</value>
375     <value>FAK</value>
376     <value>GAL</value>
377     <value>IZB</value>
378     <value>IZGAI</value>
379     <value>IZMGB</value>
380     <value>JNT</value>
381     <value>LIB</value>
382     <value>LOK</value>
383     <value>MEN</value>
384     <value>NOLARR</value>
385     <value>NOLGAL</value>
386     <value>ORD</value>
387     <value>ORO</value>
388     <value>PERARR</value>
389     <value>PERIND</value>
390     <value>SIN</value>
391     <value>ZKI</value>
392     </choice>
393     </attribute>
394     <empty/>
395     </element>
396     </define>
397     <define name="f.eratorriaren-kategoria">
398     <element name="f">
399     <attribute name="name">
400     <value>ERAKAT</value>
401     </attribute>
402     <element name="sym">
403     <attribute name="value">
404     <choice>
405     <value>ADB</value>
406     <value>ADI</value>
407     <value>ADJ</value>
408     <value>DET</value>
409     <value>IOR</value>
410     <value>IZE</value>
411     <value>LOT</value>
412     </choice>
413     </attribute>
414     <empty/>
415     </element>
416     </define>
417     <define name="f.ergatiboa">
418     <element name="f">
419     <attribute name="name">
420     <value>NORK</value>
421     </attribute>
422     <element name="sym">
423     <attribute name="value">
424     <choice>
425     <value>NIK</value>
426     <value>HIK</value>
427     <value>HIK-TO</value>
428     <value>HIK-NO</value>

```

HAP masterra

```

386 <value>HARK</value>
387 <value>GUK</value>
388 <value>ZUK</value>
389 <value>ZUEK-K</value>
390 <value>HAIEK-K</value>
391 </choice>
392 </attribute>
393 <empty/>
394 </element>
395 </element>
396 </define>
397 <define name="f.erlazioa">
398 <element name="f">
399 <attribute name="name">
400 <value>ERL</value>
401 </attribute>
402 </element name="sym">
403 <attribute name="value">
404 <choice>
405 <value>AURK</value>
406 <value>BALD</value>
407 <value>DENB</value>
408 <value>EMEN</value>
409 <value>ERLT</value>
410 <value>ESPL</value>
411 <value>HAUT</value>
412 <value>HELB</value>
413 <value>KAUS</value>
414 <value>KONPL</value>
415 <value>KONT</value>
416 <value>MOD</value>
417 <value>MOD/DENB</value>
418 <value>MOS</value>
419 <value>MOTIB</value>
420 <value>ONDO</value>
421 <value>ZHG</value>
422 </choice>
423 </attribute>
424 <empty/>
425 </element>
426 </define>
427 <define name="f.erroa">
428 <element name="f">
429 <attribute name="name">
430 <value>ERR</value>
431 </attribute name="f">
432 </define>
433 </define>
434 </attribute>
435 <ref name="mota.fs.gako"/>
436 </element>
437 </define>
438 <define name="f.error-kode">
439 <element name="f">
440 <attribute name="name">
441 <value>ERROR-KODE</value>
442 </attribute>
443 </element name="sym">
444 <attribute name="value">
445 <choice>
446 <value>LD_MA</value>
447 <value>DIAL</value>
448 <value>LD_FO</value>
449 <value>KONFOS</value>
450 <value>FO_OK</value>
451 <value>ERAT</value>
452 <value>ATZKI</value>
453 <value>A_FAK</value>
454 <value>DE_DI</value>
455 <value>AZTERT</value>
456 <value>DE_ER</value>
457 <value>NEOL</value>
458 </choice>
459 </attribute>
460 </element>
461 </define>
462 <define name="f.estandarrak">
463 <element name="f">
464 <attribute name="name">
465 <value>Estandarrak</value>
466 </attribute>
467 </attribute name="org">
468 <value>set</value>
469 </attribute>
470 </oneOrMore>
471 <ref name="mota.fs.gako"/>
472 </oneOrMore>
473 </element>
474 </define>
475 <define name="f.funtzio-sintaktikoak">
476 <element name="f">
477 <attribute name="name">
478 </define>
479 </define>
480 </define>
481 </define>

```

482	<value>FSL</value>	526	<value>@-JADNAG_MP_IZLG></>
483	</attribute>	527	<value>@-JADNAG_MP_IZLG></>
484	<attribute name="org">	528	<value>@+JADNAG_MP_PRED</value>
485	<value>list</value>	529	<value>@+JADNAG_MP_SUBJ</value>
486	</attribute>	530	<value>@+JADNAG_MP_PRED</value>
487	<oneOrMore>	531	<value>@-JADNAG_MP_KM></>
488	<element name="sym">	532	<value>@-JADNAG_MP_OBJ</value>
489	<attribute name="value">	533	<value>@-JADNAG_MP_SUBJ</value>
490	<choice>	534	<value>@-JADNAG_MP_PRED</value>
491	<value>@SUBJ</value>	535	<value>@-JADNAG_MP_< IZLG</>
492	<value>@OBJ</value>	536	<value>@+JADNAG_MP_< IZLG</>
493	<value>@ZOBJ</value>	537	<value>@+JADNAG_MP_< IZLG</>
494	<value>@ADLG</value>	538	<value>@EZ_PART</value>
495	<value>@TRIB</value>	539	<value>@APOS</value>
496	<value>@+JADNAG</value>	540	<value>@GRAD></value>
497	<value>@+JADLAG</value>	541	<value>@< GRAD</value>
498	<value>@-JADNAG</value>	542	<value>@PJ</value>
499	<value>@-JADLAG</value>	543	<value>@MP</value>
500	<value>@KM></value>	544	<value>@LOK</value>
501	<value>@IZLG></value>	545	<value>@PRED</value>
502	<value>@< IZLG</value>	546	<value>@PRT</value>
503	<value>@IA></value>	547	<value>@BST</value>
504	<value>@ID< IA</value>	548	<value>@ITJ</value>
505	<value>@< ID</value>	549	<value>@HAOS</value>
506	<value>@IZ></value>	550	</choice>
507	<value>@IZ</value>	551	</attribute>
508	<value>@+JADNAG_IZLG></value>	552	<empty/>
509	<value>@+JADLAG_IZLG></value>	553	</element>
510	<value>@-JADNAG_MP_IZLG></value>	554	</oneOrMore>
511	<value>@-JADNAG_MP</value>	555	</element>
512	<value>@+JADNAG_MP</value>	556	</define>
513	<value>@+JADLAG_MP</value>	557	<define name="f.gradu-maila">
514	<value>@+JADLAG_MP_OBJ</value>	558	<element name="f">
515	<value>@+JADLAG_MP_SUBJ</value>	559	<attribute name="name">
516	<value>@+JADLAG_MP_ADLG</value>	560	<value>MAI</value>
517	<value>@+JADLAG_MP_IZLG></>	561	</attribute>
518	<value>@+JADNAG_MP_IZLG></>	562	<element name="sym">
519	<value>@+JADNAG_MP_OBJ</value>	563	<attribute name="value">
520	<value>@+JADNAG_MP_ADLG</value>	564	<choice>
521	<value>@-JADLAG_MP</value>	565	<value>KONP</value>
522	<value>@-JADNAG_MP_OBJ</value>	566	<value>GEHI</value>
523	<value>@-JADNAG_MP_SUBJ</value>	567	<value>SUP</value>
524	<value>@-JADNAG_MP_PRED</value>	568	
525	<value>@-JADNAG_MP_ADLG</value>		

569		
570		
571		
572		
573		
574		
575		
576		
577		
578		
579		
580		
581		
582		
583		
584		
585		
586		
587		
588		
589		
590		
591		
592		
593		
594		
595		
596		
597		
598		
599		
600		
601		
602		
603		
604		
605		
606		
607		
608		
609		
610		
611		
612		
613		
614		
615		
616		
617		
618		
619		
620		
621		
622		
623		
624		
625		
626		
627		
628		
629		
630		
631		
632		
633		
634		
635		
636		
637		
638		
639		
640		
641		
642		
643		
644		
645		
646		
647		
648		
649		
650		
651		
652		
653		
654		
655		
656		
657		
658		
659		
660		
661		
662		
663		
664		

665			
666			
667			
668			
669			
670			
671			
672			
673			
674			
675			
676			
677			
678			
679			
680			
681			
682			
683			
684			
685			
686			
687			
688			
689			
690			
691			
692			
693			
694			
695			
696			
697			
698			
699			
700			
701			
702			
703			
704			
705			
706			
707			
708			
709			
710			
711			
712			
713			
714			
715			
716			
717			
718			
719			
720			
721			
722			
723			
724			
725			
726			
727			
728			
729			
730			
731			
732			
733			
734			
735			
736			
737			
738			
739			
740			
741			
742			
743			
744			
745			
746			
747			
748			
749			
750			
751			
752			
753			
754			
755			
756			
757			
758			
759			
760			

HAP masterra

761	<value>B3</value>	809	<choice>
762	<value>B4</value>	810	<value>S</value>
763	<value>B5A</value>	811	<value>P</value>
764	<value>B5B</value>	812	<value>PH</value>
765	<value>B6</value>	813	</choice>
766	<value>B7</value>	814	</attribute>
767	<value>B8</value>	815	<empty/>
768	<value>C</value>	816	</element>
769	</choice>	817	</define>
770	</choice>	818	
771	</attribute>	819	
772	</element>	820	<define name="f.numeroa-mugatasuna">
773	</define>	821	<element name="f">
774		822	<attribute name="name">
775	<define name="f.mugatasuna">	823	<value>NMG</value>
776	<element name="f">	824	</attribute>
777	<attribute name="name">	825	<element name="sym">
778	<value>MUG</value>	826	<attribute name="value">
779	</attribute>	827	<choice>
780	<element name="sym">	828	<value>S</value>
781	<attribute name="value">	829	<value>P</value>
782	<choice>	830	<value>MG</value>
783	<value>M</value>	831	</choice>
784	<value>MG</value>	832	</attribute>
785	</choice>	833	<empty/>
786	</attribute>	834	</element>
787	<empty/>	835	</define>
788	</element>	836	
789	</define>	837	
790		838	<define name="f.oinarria">
791	<define name="f.neurgarria">	839	<element name="f">
792	<element name="f">	840	<attribute name="name">
793	<attribute name="name">	841	<value>OIN</value>
794	<value>NEUR</value>	842	</attribute>
795	</attribute>	843	<choice>
796	</attribute>	844	<ref name="mota.fs.gako"/>
797	<ref name="mota.plusminusus"/>	845	<ref name="mota.str"/>
798	</element>	846	</choice>
799	</define>	847	</element>
800		848	</define>
801		849	
802	<define name="f.numeroa">	850	<define name="f.pertsona">
803	<element name="f">	851	<element name="f">
804	<attribute name="name">	852	<attribute name="name">
805	<value>NUM</value>	853	<value>PER</value>
806	</attribute>	854	</attribute>
807	<element name="sym">	855	<element name="sym">
808	<attribute name="value">	856	<attribute name="value">

HAP masterra

```

857 <choice>
858 <value>NI</value>
859 <value>HI</value>
860 <value>HURA</value>
861 <value>GU</value>
862 <value>ZU</value>
863 <value>ZUEK</value>
864 <value>HAIEK</value>
865 </choice>
866 </attribute>
867 <empty/>
868 </element>
869 </define>
870
871
872 <define name="f.plurajala">
873 <element name="f">
874 <attribute name="name">
875 <value>PLU</value>
876 </attribute>
877 <ref name="mota.plusminus"/>
878 </element>
879 </define>
880
881 <define name="f.zenbakarria">
882 <element name="f">
883 <attribute name="name">
884 <value>ZENB</value>
885 </attribute>
886 <ref name="mota.plusminus"/>
887 </element>
888 </define>
889
890 <!-- Kategoriak -->
891
892 <define name="f.kat.adb">
893 <element name="f">
894 <attribute name="name">
895 <value>KAT</value>
896 </attribute>
897 </element>
898 <attribute name="sym">
899 <value>ADB</value>
900 </attribute>
901 </element>
902 <empty/>
903 </element>
904
905 </define>
906 <define name="f.kat.adj">
907 <element name="f">
908 <attribute name="name">
909 <value>KAT</value>
910 </attribute>
911 <element name="sym">
912 <attribute name="value">
913 <value>ADI</value>
914 </attribute>
915 </empty/>
916 </element>
917 </define>
918
919 <define name="f.kat.adj">
920 <element name="f">
921 <attribute name="name">
922 <value>KAT</value>
923 </attribute>
924 <element name="sym">
925 <attribute name="value">
926 <value>ADJ</value>
927 </attribute>
928 </empty/>
929 </element>
930 </define>
931
932 <define name="f.kat.adl">
933 <element name="f">
934 <attribute name="name">
935 <value>KAT</value>
936 </attribute>
937 <element name="sym">
938 <attribute name="value">
939 <value>ADI</value>
940 </attribute>
941 </empty/>
942 </element>
943 </define>
944
945 <define name="f.kat.adt">
946 <element name="f">
947 <attribute name="name">
948 <value>KAT</value>
949 </attribute>
950 <empty/>
951 </element>
952 </define>

```

HAP masterra

953	</attribute>	1001	</element>
954	<element name="sym">	1002	</element>
955	<attribute name="value">	1003	</define>
956	<value>ADT</value>	1004	<define name="f.kat.aur">
957	</attribute>	1005	<element name="f">
958	<empty/>	1006	<attribute name="name">
959	</element>	1007	<value>KAT</value>
960	</define>	1008	</attribute>
961		1009	<element name="sym">
962	<define name="f.kat.amm">	1010	<attribute name="value">
963	<element name="f">	1011	<value>AUR</value>
964	<attribute name="name">	1012	</attribute>
965	<value>KAT</value>	1013	<empty/>
966	</attribute>	1014	</element>
967	<element name="sym">	1015	</define>
968	<attribute name="value">	1016	
969	<value>AMM</value>	1017	<define name="f.kat.bst">
970	</attribute>	1018	<element name="f">
971	</empty/>	1019	<attribute name="name">
972	</element>	1020	<value>KAT</value>
973	</define>	1021	</attribute>
974		1022	<element name="sym">
975	<define name="f.kat.asp">	1023	<attribute name="value">
976	<element name="f">	1024	<value>BST</value>
977	<attribute name="name">	1025	</attribute>
978	<value>KAT</value>	1026	<empty/>
979	</attribute>	1027	</element>
980	<element name="sym">	1028	</define>
981	<attribute name="value">	1029	
982	<value>ASP</value>	1030	<define name="f.kat.dek">
983	</attribute>	1031	<element name="f">
984	</empty/>	1032	<attribute name="name">
985	</element>	1033	<value>KAT</value>
986	</define>	1034	</attribute>
987		1035	<element name="sym">
988	<define name="f.kat.atz">	1036	<attribute name="value">
989	<element name="f">	1037	<value>DEK</value>
990	<attribute name="name">	1038	</attribute>
991	<value>KAT</value>	1039	<empty/>
992	</attribute>	1040	</element>
993	<element name="sym">	1041	</define>
994	<attribute name="value">	1042	
995	<value>ATZ</value>	1043	<define name="f.kat.det">
996	</attribute>	1044	<element name="f">
997	</empty/>	1045	<attribute name="value">
998		1046	<value>DET</value>
999		1047	</attribute>
1000		1048	</empty/>

1049		
1050		
1051		
1052		
1053		
1054		
1055		
1056		
1057		
1058		
1059		
1060		
1061		
1062		
1063		
1064		
1065		
1066		
1067		
1068		
1069		
1070		
1071		
1072		
1073		
1074		
1075		
1076		
1077		
1078		
1079		
1080		
1081		
1082		
1083		
1084		
1085		
1086		
1087		
1088		
1089		
1090		
1091		
1092		
1093		
1094		
1095		
1096		
1097		
1098		
1099		
1100		
1101		
1102		
1103		
1104		
1105		
1106		
1107		
1108		
1109		
1110		
1111		
1112		
1113		
1114		
1115		
1116		
1117		
1118		
1119		
1120		
1121		
1122		
1123		
1124		
1125		
1126		
1127		
1128		
1129		
1130		
1131		
1132		
1133		
1134		
1135		
1136		
1137		
1138		
1139		
1140		
1141		
1142		
1143		
1144		
1049	<attribute name="name" >	</attribute>
1050	<value>KAT</value>	<empty />
1051	</attribute>	</element>
1052	<element name="sym" >	</define>
1053	<attribute name="value" >	<define name="f.kat.haos" >
1054	<value>DET</value>	<element name="f" >
1055	</attribute>	<attribute name="name" >
1056	<empty />	<value>KAT</value>
1057	</element>	</attribute>
1058	</define>	<element name="sym" >
1059	</define>	<attribute name="value" >
1060	<define name="f.kat.eii" >	<element name="sym" >
1061	<element name="f" >	<attribute name="value" >
1062	<attribute name="name" >	<value>EII</value>
1063	<value>KAT</value>	</attribute>
1064	</attribute>	<empty />
1065	</element>	</define>
1066	<element name="sym" >	<define name="f.kat.eri" >
1067	<attribute name="value" >	<element name="f" >
1068	<value>EII</value>	<attribute name="name" >
1069	</attribute>	<value>KAT</value>
1070	</element>	</attribute>
1071	</define>	<empty />
1072	</define>	</element>
1073	<define name="f.kat.eri" >	</define>
1074	<element name="f" >	<define name="f.kat.eri" >
1075	<attribute name="name" >	<element name="f" >
1076	<value>KAT</value>	<attribute name="name" >
1077	</attribute>	<value>KAT</value>
1078	</element>	<element name="sym" >
1079	<element name="sym" >	<attribute name="value" >
1080	<attribute name="value" >	<value>ERI</value>
1081	<value>ERI</value>	</attribute>
1082	</attribute>	<empty />
1083	</element>	</define>
1084	</define>	<define name="f.kat.gra" >
1085	</define>	<element name="f" >
1086	<define name="f.kat.gra" >	<attribute name="name" >
1087	<element name="f" >	<value>KAT</value>
1088	<attribute name="name" >	</attribute>
1089	<value>KAT</value>	<element name="sym" >
1090	</attribute>	<attribute name="value" >
1091	<element name="sym" >	<value>ITJ</value>
1092	<attribute name="value" >	</attribute>
1093	<value>GRA</value>	<empty />
1094	</define>	</define>
1095	</define>	</define>
1096	</define>	</define>

1145	<define name="f.kat.ize">	1193	<attribute name="value">
1146	<element name="f">	1194	<value>PRT</value>
1147	<attribute name="name">	1195	</attribute>
1148	<value>KAT</value>	1196	<empty/>
1149	</attribute>	1197	</element>
1150	<element name="sym">	1198	</element>
1151	<attribute name="value">	1199	</define>
1152	<value>IZE</value>	1200	<!-- Kategoria osatuak (goi mailako ezaugarrienak) -->
1153	</attribute>	1201	
1154	<empty/>	1202	
1155	</element>	1203	
1156	</define>	1204	<define name="f.kat-osatua.adb">
1157		1205	<element name="f">
1158		1206	<attribute name="name">
1159	<define name="f.kat.lot">	1207	<value>KAT</value>
1160	<element name="f">	1208	</attribute>
1161	<attribute name="name">	1209	<element name="sym">
1162	<value>KAT</value>	1210	<attribute name="value">
1163	</attribute>	1211	<choice>
1164	<element name="sym">	1212	<value>ADB</value>
1165	<attribute name="value">	1213	<value>ADB_IZEELI</value>
1166	<value>LOT</value>	1214	<value>ADB_IZEELI_IZEELI</value>
1167	</attribute>	1215	<value>ADB_IZEELI_IZEELI_IZEELI</value>
1168	<empty/>		↳ value>
1169	</element>	1216	</choice>
1170	</define>	1217	</attribute>
1171		1218	<empty/>
1172		1219	</element>
1173	<define name="f.kat.mar">	1220	</define>
1174	<element name="f">	1221	
1175	<attribute name="name">	1222	
1176	<value>KAT</value>	1223	<define name="f.kat-osatua.adl">
1177	</attribute>	1224	<element name="f">
1178	<element name="sym">	1225	<attribute name="name">
1179	<attribute name="value">	1226	<value>KAT</value>
1180	<value>MAR</value>	1227	</attribute>
1181	</attribute>	1228	<element name="sym">
1182	<empty/>	1229	<attribute name="value">
1183	</element>	1230	<choice>
1184	</define>	1231	<value>ADI</value>
1185		1232	<value>ADI_IZEELI</value>
1186		1233	<value>ADI_IZEELI_IZEELI</value>
1187	<define name="f.kat.prt">	1234	<value>ADI_IZEELI_IZEELI_IZEELI</value>
1188	<element name="f">		↳ value>
1189	<attribute name="name">	1235	</choice>
1190	<value>KAT</value>	1236	</attribute>
1191	</attribute>	1237	<empty/>
1192	<element name="sym">	1238	</element>

1239	</element>	1285	<element name="sym">
1240	</define>	1286	<attribute name="value">
1241		1287	<choice>
1242	<define name="f.kat-osatua.adj">	1288	<value>ADT</value>
1243	<element name="f">	1289	<value>ADT_IZEELI</value>
1244	<attribute name="name">	1290	<value>ADT_IZEELI_IZEELI</value>
1245	<value>KAT</value>	1291	<value>ADT_IZEELI_IZEELI_IZEELI</value>
1246	</attribute>		↳ value>
1247	</element>	1292	</choice>
1248	<element name="sym">	1293	</attribute>
1249	<attribute name="value">	1294	<empty/>
1250	<choice>	1295	</element>
1251	<value>ADJ</value>	1296	</define>
1252	<value>ADJ_IZEELI</value>	1298	<define name="f.kat-osatua.bst">
1253	<value>ADJ_IZEELI_IZEELI</value>	1299	<element name="f">
1254	<value>ADJ_IZEELI_IZEELI_IZEELI</value>	1300	<attribute name="name">
1255	↳ value>	1301	<value>KAT</value>
1256	</choice>	1302	</attribute>
1257	</element>	1303	<element name="sym">
1258	</define>	1304	<attribute name="value">
1259		1305	<choice>
1260		1306	<value>BST</value>
1261	<define name="f.kat-osatua.adl">	1307	<value>BST_IZEELI</value>
1262	<element name="f">	1308	<value>BST_IZEELI_IZEELI</value>
1263	<attribute name="name">	1309	<value>BST_IZEELI_IZEELI_IZEELI</value>
1264	<value>KAT</value>	1310	<value>BST_IZEELI_IZEELI_IZEELI_IZEELI</value>
1265	</attribute>		↳ value>
1266	</element>	1311	</choice>
1267	<element name="sym">	1312	</attribute>
1268	<attribute name="value">	1313	<empty/>
1269	<choice>	1314	</element>
1270	<value>ADL</value>	1315	</define>
1271	<value>ADL_IZEELI</value>	1316	<define name="f.kat-osatua.det">
1272	<value>ADL_IZEELI_IZEELI</value>	1317	<element name="f">
1273	↳ value>	1318	<attribute name="name">
1274	</choice>	1319	<value>KAT</value>
1275	</attribute>	1320	</attribute>
1276	<empty/>	1321	<element name="sym">
1277	</element>	1322	<attribute name="value">
1278	</define>	1323	<choice>
1279		1324	<value>DET</value>
1280	<define name="f.kat-osatua.adt">	1325	<value>DET_IZEELI</value>
1281	<element name="f">	1326	<value>DET_IZEELI_IZEELI</value>
1282	<attribute name="name">	1327	<value>DET_IZEELI_IZEELI_IZEELI</value>
1283	<value>KAT</value>	1328	<value>DET_IZEELI_IZEELI_IZEELI_IZEELI</value>
1284	</attribute>	1329	↳ value>

1330			
1331			
1332			
1333			
1334			
1335			
1336			
1337			
1338			
1339			
1340			
1341			
1342			
1343			
1344			
1345			
1346			
1347			
1348			
1349			
1350			
1351			
1352			
1353			
1354			
1355			
1356			
1357			
1358			
1359			
1360			
1361			
1362			
1363			
1364			
1365			
1366			
1367			
1368			
1369			
1370			
1371			
1372			
1373			
1374			
1375			
1376			
1377			
1378			
1379			
1380			
1381			
1382			
1383			
1384			
1385			
1386			
1387			
1388			
1389			
1390			
1391			
1392			
1393			
1394			
1395			
1396			
1397			
1398			
1399			
1400			
1401			
1402			
1403			
1404			
1405			
1406			
1407			
1408			
1409			
1410			
1411			
1412			
1413			
1414			
1415			
1416			
1417			
1418			
1419			
1420			
1421			

1422		
1423		
1424		
1425		
1426		
1427		
1428		
1429		
1430		
1431		
1432		
1433		
1434		
1435		
1436		
1437		
1438		
1439		
1440		
1441		
1442		
1443		
1444		
1445		
1446		
1447		
1448		
1449		
1450		
1451		
1452		
1453		
1454		
1455		
1456		
1457		
1458		
1459		
1460		
1461		
1462		
1463		
1464		
1465		
1466		
1467		
1468		
1469		
1470		
1471		
1472		
1473		
1474		
1475		
1476		
1477		
1478		
1479		
1480		
1481		
1482		
1483		
1484		
1485		
1486		
1487		
1488		
1489		
1490		
1491		
1492		
1493		
1494		
1495		
1496		
1497		
1498		
1499		
1500		
1501		
1502		
1503		
1504		
1505		
1506		
1507		
1508		
1509		
1510		
1511		
1512		
1513		

1514	<element name="sym">	1562	<element name="f">
1515	<attribute name="value">	1563	<attribute name="name">
1516	<choice>	1564	<value>AZP</value>
1517	<value>SIN</value>	1565	</attribute>
1518	<value>FAK</value>	1566	<element name="sym">
1519	<value>ADK</value>	1567	<attribute name="value">
1520	<value>ADP</value>	1568	<choice>
1521	</choice>	1569	<value>DZH</value>
1522	</attribute>	1570	<value>ERKARR</value>
1523	<empty/>	1571	<value>DZG</value>
1524	</element>	1572	<value>ERKIND</value>
1525	</define>	1573	<value>ORD</value>
1526		1574	<value>ORO</value>
1527		1575	<value>NOLGAL</value>
1528	<define name="f.azpkat.adj">	1576	<value>NOLARR</value>
1529	<element name="f">	1577	<value>BAN</value>
1530	<attribute name="name">	1578	</choice>
1531	<value>AZP</value>	1579	</attribute>
1532	</attribute>	1580	<empty/>
1533	<element name="sym">	1581	</element>
1534	<attribute name="value">	1582	</define>
1535	<choice>	1583	
1536	<value>ARR</value>	1584	<define name="f.azpkat.ior">
1537	<value>GAL</value>	1585	<element name="f">
1538	</choice>	1586	<attribute name="name">
1539	</attribute>	1587	<value>AZP</value>
1540	<empty/>	1588	</attribute>
1541	</element>	1589	<element name="sym">
1542	</define>	1590	<attribute name="value">
1543		1591	<choice>
1544		1592	<value>PERARR</value>
1545	<define name="f.azpkat.bst">	1593	<value>IZGMB</value>
1546	<element name="f">	1594	<value>PERIND</value>
1547	<attribute name="name">	1595	<value>ELK</value>
1548	<value>AZP</value>	1596	<value>IZGGAL</value>
1549	</attribute>	1597	</choice>
1550	<element name="sym">	1598	</choice>
1551	<attribute name="value">	1599	</attribute>
1552	<choice>	1600	<empty/>
1553	<value>ARR</value>	1601	</element>
1554	</choice>	1602	</define>
1555	</attribute>	1603	
1556	<empty/>	1604	
1557	</element>	1605	<define name="f.azpkat.ize">
1558	</define>	1606	<element name="f">
1559		1607	<attribute name="name">
1560		1608	<value>AZP</value>
1561	<define name="f.azpkat.det">	1609	</attribute>

HAP mastera

```

1610 <element name="sym">
1611   <attribute name="value" />
1612 </choice>
1613   <value>ARR</value>
1614   <value>LIB</value>
1615   <value>IZB</value>
1616   <value>ZKI</value>
1617 </choice>
1618 </attribute>
1619 <empty />
1620 </element>
1621 </define>
1622
1623 <define name="f.azpkat.lot">
1624   <element name="f">
1625     <attribute name="name" />
1626     <value>AZP</value>
1627 </attribute>
1628 </element>
1629 <attribute name="sym">
1630   <attribute name="value" />
1631 </choice>
1632   <value>LOK</value>
1633   <value>JNT</value>
1634   <value>MEN</value>
1635 </choice>
1636 </attribute>
1637 <empty />
1638 </element>
1639 </define>
1640
1641 </grammar>
1642

```

```

1 <!-- motak.rng -->
2
3 <?xml version="1.0" encoding="UTF-8" ?>
4 <grammar xmlns="http://relaxng.org/ns/structure/1.0"
5   xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
6   datatypeLibrary="http://www.w3.org/2001/XMLSchema-1
7     datatypes">
8   <define name="mota.str">

```

```

9   <element name="str">
10     <data type="string" />
11 </element>
12 </define>
13 <define name="mota.nbr">
14   <element name="nbr">
15     <attribute name="value" />
16     <data type="nonNegativeInteger" />
17 </attribute>
18 </element>
19 </define>
20
21 <define name="mota.fs.gako">
22   <element name="fs">
23     <attribute name="type" />
24     <value>Gako</value>
25 </attribute>
26 </element>
27 <attribute name="f">
28   <attribute name="name" />
29     <value>Sarrera</value>
30 </attribute>
31 <ref name="mota.str" />
32 </element>
33 <element name="f">
34   <attribute name="name" />
35     <value>Homografo-Id</value>
36 </attribute>
37 <ref name="mota.nbr" />
38 </element>
39 </define>
40
41 <define name="mota.plusminus">
42 <choice>
43   <element name="plus" />
44   <empty />
45 </element>
46 <element name="minus" />
47 <empty />
48 </element>
49 </choice>
50 </define>
51
52 </grammar>
53

```


Erreferentziak

- I. Aduriz, I. Aldezabal, I. Alegria, X. Artola, N. Ezeiza, eta R. Urizar. EUSLEM: A Lemmatiser / Tagger for Basque. In *EURALEX'96, Part 1, 17-26.*, Göteborg, 1996.
- I. Aduriz, E. Agirre, I. Aldezabal, I. Alegria, X. Arregi, J. M. Arriola, X. Artola, K. Gojenola, A. Maritxalar, K. Sarasola, eta M. Urkia. A Word-grammar based morphological analyzer for agglutinative languages. In *Proc. of International Conference on Computational Linguistics. COLING'2000*, Saarbrücken (Germany), 2000.
- I. Aduriz, M. Aranzabe, J. M. Arriola, A. Díaz de Ilarraza, K. Gojenola, M. Oronoz, eta L. Uria. *Computational Linguistics and Intelligent Text Processing*, chapter A Cascaded Syntactic Analyser for Basque, pages 124–135. 2945 LNCS Series. Springer Verlag, 2004.
- I. Aldezabal, O. Ansa, B. Arrieta, X. Artola, A. Ezeiza, G. Hernández, eta M. Lersundi. EDBL: a general lexical basis for the automatic processing of Basque. In *IRCS Workshop on linguistic databases.*, Philadelphia. USA, 2001.
- I. Alegria, X. Artola, K. Sarasola, eta M. Urkia. Automatic morphological analysis of Basque. *Literary & Linguistic Computing*, 11(4):193–203, 1996.
- I. Alegria, O. Ansa, X. Artola, N. Ezeiza, K. Gojenola, eta R. Urizar. Representation and Treatment of Multiword Expressions in Basque. In *ACL workshop on Multiword Expressions*, Barcelona, 2004.
- M. Aranzabe, J. M. Arriola, A. Díaz de Ilarraza, eta K. Gojenola. Towards a dependency parser for Basque. In *Proc. of International Conference on Computational Linguistics. COLING'2004*, Geneva, 2004.
- X. Artola, A. Díaz de Ilarraza, N. Ezeiza, K. Gojenola, G. Hernández, eta A. Soroa. A class library for the integration of NLP tools: Definition and implementation of an abstract data type collection for the manipulation of SGML documents in a context of stand-off linguistic annotation. In *Proc. of the Third Int. Conf. on Language Resources and Evaluation*, Las Palmas (Spain), 2002.
- X. Artola, A. Díaz de Ilarraza, N. Ezeiza, K. Gojenola, A. Sologaitoa, eta A. Soroa. EU-LIA: a graphical web interface for creating, browsing and editing linguistically annotated corpora. In *LREC 2004. Workshop on XbRAC*, Lisbon (Portugal), 2004.
- X. Artola, A. Díaz de Ilarraza, N. Ezeiza, G. Labaka, K. Gojenola, A. Sologaitoa, eta A. Soroa. A framework for representing and managing linguistic annotations based on typed feature structures. In *RANLP 2005*, Borovets (Bulgaria), 2005.
- X. Artola, A. Díaz de Ilarraza, A. Soroa, eta A. Sologaitoa. Dealing with complex linguistic annotations within a language processing framework. *IEEE Transactions on Audio, Speech and Language Processing*, 17(5):904–915, 2009.

- Bob Carpenter. Typed feature structures: A generalization of first-order terms. In V. Saraswat eta K. Ueda, editors, *Logic Programming: Proc. of the 1991 International Symposium*, pages 187–201. MIT Press, Cambridge, MA, 1991.
- N. Ezeiza, I. Aduriz, I. Alegria, J. M. Arriola, eta R. Urizar. Combining Stochastic and Rule-based Methods for Disambiguation in Agglutinative Languages. In *Proc. of COLING-ACL'98*, pages 10–14, Montreal (Canada), 1998.
- N. Ide eta J. Véronis, editors. *Text Encoding Initiative. Background and Context*. Kluwer Academic Pub, 1995.
- C. M. Sperberg-McQueen eta L. Burnard, editors. *TEI P4: Guidelines for Electronic Text Encoding and Interchange*. Oxford, 4 edition, 2002.