The Afroasiatic Morphological Archive: A Paradigm Database

Gene Gragg
Oriental Institute
Depts. Linguistics, Near Eastern Langs & Civs
University of Chicago

AAMA: Afroasiatic Morphological Archive

(an extension of)

COMA: Cushitic-Omotic Morphological Archive

- Supported by Mellon Emeritus fellowship (Nov. 2008 – Mar. 2010)
- Computer expertise supplied by Gregg Reynolds of National Opinion Research Center.

OUTLINE

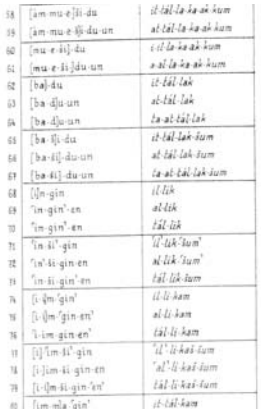
1. OBGT: Homage to an Earlier Project
2. The AAMA Project: Goals
3. Collaborative Development Context: git
4. Linked Data: RDF
   1. Morphological Data as Graph
   2. Morphological Data as Network of Statements (Excursus: Data Formats)
5. Querying Linked Data: SPARQL
6. Towards an Interface

1. OBGT: Homage to an earlier project:

(cf. Landsberger et al. 1956, Black 1991, Huber 2007)

OBGT 7
(Old Babylonian Grammatical Texts 7)

MSL 4 (1956)
p. 90



Section	Line	ObjNum	SubjNum	Mood	Ventive	Stem	ObjPers	Tense-Aspect	SubjPers
20	58	sg	sg	indicative	ventive	Gt	2	present	3
	59	sg	sg	indicative	ventive	Gt	2	present	1
21	60	sg	sg	indicative	ventive	G	2	present	3
	61	sg	sg	indicative	ventive	G	2	present	1
22	62	sg	sg	indicative	non-ven	Gt	Ø	present	3
	63	sg	sg	indicative	non-ven	Gt	Ø	present	1
	64	sg	sg	indicative	non-ven	Gt	Ø	present	2
23	65	sg	sg	indicative	non-ven	Gt	3	present	3
	66	sg	sg	indicative	non-ven	Gt	3	present	1
	67	sg	sg	indicative	non-ven	Gt	3	present	2
24	68	sg	sg	indicative	non-ven	G	Ø	preterite	3
	69	sg	sg	indicative	non-ven	G	Ø	preterite	1
	70	sg	sg	indicative	non-ven	G	Ø	preterite	2
25	71	sg	sg	indicative	non-ven	G	3	preterite	3
	72	sg	sg	indicative	non-ven	G	3	preterite	1
	73	sg	sg	indicative	non-ven	G	3	preterite	2
26	74	sg	sg	indicative	ventive	G	Ø	preterite	3
	75	sg	sg	indicative	ventive	G	Ø	preterite	1
	76	sg	sg	indicative	ventive	G	Ø	preterite	2
27	77	sg	sg	indicative	ventive	G	3	preterite	3
	78	sg	sg	indicative	ventive	G	3	preterite	1
	79	sg	sg	indicative	ventive	G	3	preterite	2

OBGT Paradigm Attributes and Values:

- Object Num sg, pl
- Subject Num sg, pl
- Modal indicative, modal
- Ventive ventive, non-ventive
- Stem G, Gt, Š, N
- Object Pers 1, 2, 3
- TAM pres, pret, imprtv, opt, cohor
- Subject Pers 1, 2, 3

2. The AAMA Project: Goals

AAMA: The challenge:

- Create a paradigm database whose
  - data can be:
    - curated (edited/created) -- and hopefully shared!
    - inspected
    - manipulated
    - queried
  - on individual machine (initial display: browser)

The Term:

- Paradigm: systematic listing, for a lexeme chosen as exemplary, of a set of word-forms illustrating all occurring value-combinations for each of a selected set of morphosyntactic attributes.
  - Taken together, the set of paradigms chosen for a language should illustrate all possible value-combinations for all possible attribute combinations attested in the language.

Paradigm: a persistent notion

- Millennial pedagogical and descriptive practice
- Recent reevaluation in linguistic main-stream
- A radical (slightly earlier) view

The paradigm in the linguistic mainstream

- Hockett 1954
- Robins 1959
- Matthews 1972
- Zwicky 1985
- Aronoff 1994
- Stump 2001
- Blevins & Blevins 2009

AAMA Objectives - 1

- Archive: make available and comparable the major morphological paradigms of some fifty Cushitic and Omotic languages
  - longer term: tool that can help situate the morphologies of these two language families within Afroasiatic

AAMA Objectives - 2

- Database: Query, contrast and configure the complex paradigmatic structures it contains, within a given language and between languages.

AAMA Objectives -3

- Morphological Theory: Tool for exploration of typology and structure of the form of linguistic organization known as the paradigm.

3. Collaborative Development Context: git and GitHub

- git: distributed version control system (VCS)
  - developed by Linus Torvalds (Linux)
- GitHub: "social network" approach to projects
  - cf. "GitHub bootcamp" on https://github.com

AAMA (for now) . . .

https://github.com/gbgg/aama

(on branch "dev-aama": git *checkout* dev-aama)

- *download* zip – see what's there
- *fork* repository – follow, keep up to date
- *clone* – work with it on your own machine

What is there . . .

- Data (interface being developed elsewhere)
  - An extensive directory structure under a root aama/data
  - Overview (DocBook format):
    aama/docs/docbook/AAMADocumentation.html
  - [A copy of this presentation will be found under:]
    aama/docs/slides

The Language Data

Family	Lang	Family	Language	Family	Language	Family	Language
Beja	Arteiga	Omo-Tana	Somali		Kambaata		Koorete
	Bishari		Rendille		Sidaama		Maale
	Beniamer		Bayso	SE. Cush.	Tsamakko	(Semitic)	(Akkadian-OB)
	Hadendowa		Boni-jara		Gawwada		(Hebrew)
	Atmaan		Boni-kilii		Yaaku		(Syriac)
Agaw	Kemant		Arbore		Dahalo		(Arabic)
	Awngi		Dhaasanac	S. Cush.	Burunge		(Geez)
	Bilin		Elmolo		Iraqw	(Egypt.)	(middle)
	Khamtanga	HE. Cush.	Alaaba	Omotic	Dizi		(Coptic-Sahidic)
Sah-Afar	Afar		B urji		Shinassha		
	Saho		Gedeo		Yemsa		
Oromoid	Oromo		Hadiyya		Wolaytta		

Format for Individual Language Documentation in Overview

- II. Language Documentation: Beja
  - 4. Beja-Arteiga Language Data
    - General Information
      - Location-Speakers
      - Bibliographic Source
    - Paradigm Lexemes
    - Morphosyntactic Properties
    - Attested Archive Paradigms
    - Phonological Inventory

4.1 Linked Data: Morphology as Graph

The Data

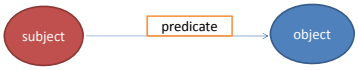
- A number of formats possible, all roughly in the XML orbit
- One chosen here is RDF datastore

RDF: the database

- "The **Resource Description Framework (RDF)** is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata data model. It has come to be used as a general method for conceptual description or modeling of information that is implemented in web resources, using a variety of syntax formats."

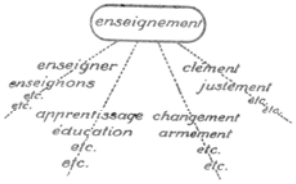
Resource Description Framework (RDF)

- "based upon the idea of making statements about resources (in particular Web resources) in the form of subject-predicate-object expressions"
- graphic visualization: node-edge-node



Graph Notation

- for example:



Saussure's generalization of "paradigm"

"rapports associatifs" (*Cours, p. 175)*

A radical view

– langue = "un système où tout se tient"

– the two axes of langue *(Cours de linguistique générale, chs. 5 & 6):*

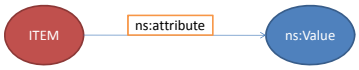
1. rapports syntagmatiques
2. rapports associatifs

– Viewed along its associative/paradigmatic axis:

- LANGUE IS A GRAPH

From out point of view . . .

- item situated with respect to resources



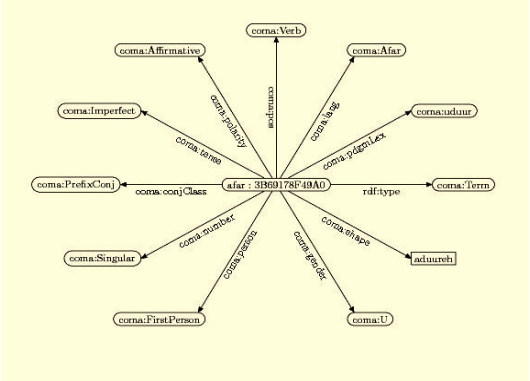
- where "ns:" uniquely identifies the "namespace" of a resource – where you can find it (URI/URL)
  - cf. "dc:" ("Dublin Core") widely used by libraries for exchange of documents and catalogue references.
- So . . . morphosyntactic attribute/value terminology is a RESOURCE

So … given an Afar paradigm

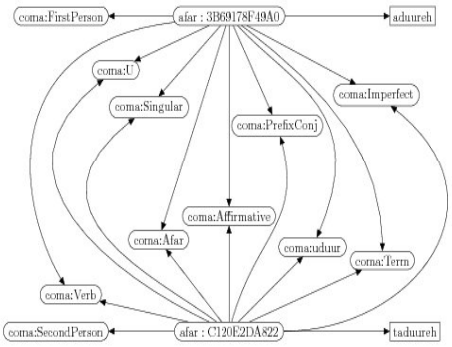
lexeme=duur_, conjugation=prefixConj polarity=affirmative

tense	number	person	gender	shape
imperf	sg	p1		aduureh
imperf	sg	p2		taduureh
imperf	sg	p3	m	yaduureh
imperf	sg	p3	f	taduureh
imperf	pl	p1		naduureh
imperf	pl	p2		taduureenih
imperf	pl	p3		yaduureenih

graph visualization: aduureh



paradigm: aduureh, taduureh . . .



4.2 Linked Data: Morphology as Network of Statements

In Practice . . .

- A network of statements ("triples")
- Syntax: ttl
  - (pronounced/written-out:
    - "turtle": Terse RDF Triple Language)
- Statement Format: "s p o ."
- Generally with interpretation:
  - (morphological-)object property value .

Three Main Classes of Morphological Object ("Subject")

- aamas:Term -- what you find in a pdgm cell
  - properties: tense, number, person, gender, etc. etc.
- aamas:Lexeme -- the paradigm lexeme
  - properties: lemma, gloss, etc. etc.
- aama:TermSet -- a "language"
  - properties: name, variety, family etc. etc.

Properties and Values: Namespace-controlled vocabularies

- The namespace prefixes of AAMA:

```
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix aama:    <http://id.oi.uchicago.edu/aama/2013/> .
@prefix aamas:   <http://id.oi.uchicago.edu/aama/2013/schema/> .
@prefix aamav:   <http://id.oi.uchicago.edu/aama/2013/value/> .
```

Properties and Values: Inference-1

- aamas:Cluster rdfs:subClassOf rdfs:Class .
- aamas:Lexeme rdfs:subClassOf aamas:Cluster .
- aamas:MuExponent rdfs:subClassOf rdfs:Class .
- aamas:muProperty rdfs:subProperty rdfs:Property .
- aamas:MuScheme rdfs:subClassOf aamas:Cluster .
- aamas:MuTerm rdfs:subClassOf aamas:Cluster .
- aamas:Term rdf:type aamas:Lexeme .
- aamas:Term rdf:type aamas:MuScheme .
- aamas:Term rdf:type aamas:MuTerm .
- aamas:Text rdfs:subClassOf rdfs:Class .
- aamas:Token rdfs:subClassOf rdfs:Class .

Properties and Values: Inference-2

- aamav:Absolutive aama:lang aama:Oromo .
- aamav:Absolutive rdf:type aamav:Case .
- aamav:Absolutive rdfs:label "Absolutive" .
- aama:case rdfs:subPropertyOf aama:muProperty .
- aama:case rdfs:label "case" .
- aama:case rdfs:Range aamav:Case .
- aama:case rdfs:Domain aamas:Term .
- aama:case aama:lang aama:Oromo .
- aamav:Case rdfs:subClassOf aamav:MuExponent .
- aamav:Case rdfs:label "case exponents" .
- aamav:Case aama:lang aama:Oromo .

Properties and Values: Alternate Terminologies

- E.g.:
  - aamav:Aorist owl:SameAs reinisch1873:Plusquamperfectum
  - aamav:Aorist owl:SameAs somebody-else:PunctualPast

The end of the rainbow?

- The GOLD standard: "gold:"
- "General Ontology for Linguistic Description"
- Terry Langendoen,
  - http://www.linguistics-ontology.org/
- Cf. also e-Linguistics (Toolkit)
  - http://uakari.ling.washington.edu/e-linguistics/

ttl syntax morphology – verbose

(ms/MS = "morphosyntactic")

("a" official abbreviation for "rdf:type")

ID_x	a	aama:$MSObject_i$.
ID_x	aama:$msAttribute_a$	aama:$MSValue_j$.
ID_x	aama:$msAttribute_b$	aama:$MSValue_k$.
ID_x	aama:$msAttribute_c$	aama:$MSValue_l$.
	…	…

ttl Morphology - concise

ID_x	a	aama:pdgmCell ;
	aama:$msAttribute_a$	aama:$MSValue_j$;
	aama:$msAttribute_b$	aama:$MSValue_k$;
	aama:$msAttribute_c$	aama:$MSValue_l$;
	…	…
.		

ttl: Afar

```
## sg,1,U,aduureh
afar:Tok_3B69178F49A0
    a                   aama:Term ;
    aama:pdgmLex        aama:_uduur_;
    aama:lang           aama:Afar;
    aama:pos            aama:Verb;
    aama:polarity       aama:Affirmative;
    aama:tense          aama:Imperfect;
    aama:conjClass      aama:PrefixConj;
    aama:number         aama:Singular;
    aama:person         aama:FirstPerson;
    aama:gender         aama:U;
    aama:shape          "aduureh";
    .
```

THUS:

- all of Afar paradigmatic morphology can be stated as a (large) set of ttl statements .
- all of the AAMA database can be stated as a (very large) set of ttl statements.
- all of an Afroasiatic database could be stated as a mind-boggling huge set of ttl statements.
- … … …

RDF Datastore:

- Of course in ttl representations, you do not immediately see the connections between the various systems of attributes and values as in the earlier graphic representations (although you can always trace them through by hand).

RDF Datastore

- But the point is that the software which is an integral part of an RDF datastore does see them, all of them, virtually at once .
- Whence the ability of software that can handle ttl to respond to queries for: "all the 2nd pers fem forms in Berber, Cushitic, and Semitic", "all the imperfective and perfective suffixConjugation forms", etc. etc.

RDF Datastore

- In short: ***RDF databases provide a tool not only for retrieving canonical paradigms, but for tearing them apart, and reassembling them in less canonical, but possibly insightful, ways.***

Excursus: Data Formats

Any Convenient & Consistent Data Entry Format:

e.g. data/beja/arteiga/src/beja-arteiga-pdgms.txt

```
PDGM:
ID: beja_H-VPrefAffCCCAor
NAME: beja_H-VPrefAffCCCAor
FEATURES:
        lang = beja-arteiga
        pos = verb
        tam = aorist
        polarity = affirmative
        rootClass = CCC
        conjClass = prefix
        pdgmLex = dbl
FORMS:
        number  |       person  |       gender  |       token
        sg      |       p1      |       -       |       ?-iidbíl
        sg      |       p2      |       m       |       t-iidbil- `a
        sg      |       p2      |       f       |       t-iidbil- `i
        sg      |       p3      |       m       |       ?-iidbíl
        sg      |       p3      |       f       |       t-iidbíl
        pl      |       p1      |       -       |       n-iidbíl
        pl      |       p2      |       -       |       t-iidbil- `na
        pl      |       p3      |       -       |       ?-iidbil- `na
```

Explicit Static (persistent) format

- Need explicit format for
  - storage
  - basis for further format transformations
  - data exchange
- XML is good (of course) . . .

Static Format: XML (pdgm term)
data/beja/arteiga/beja-arteiga-pdgms.xml

```
<pdgm >
    <common-properties>
      <prop type="conjClass" val="Prefix"/>
      <prop type="lang" val="Beja-Arteiga"/>
      <prop type="lexlabel" val="dbl"/>
      <prop type="polarity" val="Affirmative"/>
      <prop type="pos" val="Verb"/>
      <prop type="rootClass" val="CCC"/>
      <prop type="tam" val="Aorist"/>
    </common-properties>
    <termcluster>
      <term id="ID05c2c97a-8896-402b-9e3c-45f99d951510">
        <prop type="gender" val="Masc"/>
        <prop type="number" val="Singular"/>
        <prop type="person" val="Person3"/>
        <prop type="token" val="?-iidbil"/>
      </term>
      . . .  . . . .
    </termcluster>
</pdgm>
```

Static format

- XML is good (of course) . . . but for our purposes
- JSON is better

Static format: JSON

```
pdgm = {
    "pdgmattributes": {
        "conjClass": "prefix",
        "rootClass":"CCC",
        "lang": "beja-arteiga",
        "pdgmLex": "dbl",
        "polarity": "affirmative",
        "pos": "verb",
        "tam": "aorist"
        },
"forms": [
        ["number", "person", "gender",
           "token"],
        ["sg", "p1", "U", " ?-iidbíl "],
        ["sg", "p2", "m", " t-iidbil- `a "],
        ["sg", "p2", "f", " t-iidbil- `i "],
        ["sg", "p3", "m", " ?-iidbíl "],
        ["sg", "p3", "f", " t-iidbíl "],
        ["pl", "p1", "U", " n-iidbíl "],
        ["pl", "p2", "U", " t-iidbil- `na "],
        ["pl", "p3", "U", " ?-iidbil- `na "]

    ]
}
```

json Format (1)

- short for **JavaScript Object Notation**
- **http://www.json.org/**
- "lightweight computer data interchange format . . . alternative to XML"

json Format (2)

- text-based, human-readable natural format for registering paradigms
- identical, or nearly so, with attribute/value arrays in many current scripting and programming languages
- trivially transformable to XML

Finally - RDF (ttl) format:

cf. a pdgm term in data/beja/arteiga/beja-arteiga-pdgms.ttl

```
aama:ID05c2c97a     a                   aamas:Term ;
                    aamas:lexeme        aama:Beja-Arteiga-dbl;
                    aama:conjClass      aamav:Prefix  ;
                    aama:lang           aama:Beja-Arteiga ;
                    aama:polarity       aamav:Affirmative  ;
                    aama:pos            aamav:Verb  ;
                    aama:rootClass      aamav:CCC  ;
                    aama:tam            aamav:Aorist  ;
                    aama:gender         aamav:Masc  ;
                    aama:number         aamav:Singular  ;
                    aama:person         aamav:Person3  ;
                    aama:token          "?-iidbíl"
.
```

5. Querying Linked Data: SPARQL

RDF Databases

- a number of SQL-like languages have been developed to query RDF databases:
  - SPARQL
    - (so-called "recursive name":
      - SPARQL Protocol And RDF Query Language)
- there is an increasing use of RDF databases in many domains, including (documentary) linguistics

Display: Getting back the paradigms

- SPARQL
  - W3C specifications:
    - http://www.w3.org/TR/rdf-sparql-query/
  - Bob DuCharme, *Learning SPARQL: Querying and Updating with SPARQL 1.1* (O'Reilly, 2011)
  - Fuseki
    - http://jena.apache.org/
  - Set of queries for AAMA
    - aama/sparql/rq-ru/

Example of Simple SPARQL Query

```
SELECT  ?number ?person ?gender ?token
WHERE
{
        ?s      aamas:lexeme        aama:Beja-Arteiga-dbl .
        ?s      aama:conjClass      aamav:Prefix  .
        ?s      aama:lang           aama:Beja-Arteiga .
        ?s      aama:polarity       aamav:Affirmative  .
        ?s      aama:pos            aamav:Verb  .
        ?s      aama:tam            aamav:Present .
        ?s      aama:number         ?number .
        ?s      aama:person         ?person .
        ?s      aama:gender         ?gender .
        ?s      aama:token          ?token .
}
ORDER BY   DESC(?number) ?person DESC(?gender)
```

SPARQL Query (2)

- Of course in an actual user interface, query format details actually sent to datastore server would be hidden.
- User would specify details and format of desired display from graphic pick-lists, drop-down lists, etc.

Output of query: Beja Verb, present.

num	pers	gen	shape
sg	1		*ʔa-danbíil*
	2	m	*danbiil-`a*
		f	*danbiil-`i*
	3	m	*danbíil*
		f	*danbíil*
pl	1		*n-eedbíl*
	2		*t-eedbil-`na*
	3		*ʔ-eedbil-`na*

Add to Query (essentially):

- {aama:tam aama:Present}
  UNION
  {aama:tam aama:Past}

Output of query:

num	pers	gen	tense	
			present	**past**
sg	1		*ʔa-danbíil*	*ʔa-dbíl*
	2	m	*danbiil-`a*	*ti-dbil-`a*
		f	*danbiil-`i*	*ti-dbil-`i*
	3	m	*danbíil*	*ʔi-dbíl*
		f	*danbíil*	*ti-dbíl*
pl	1		*n-eedbíl*	*ni-dbíl*
	2		*t-eedbil-`na*	*ti-dbil-`na*
	3		*ʔ-eedbil-`na*	*ʔi-dbil-`na*

Add to above:

- {aama:lang aama:Beja-Arteiga}
  UNION
  {aama:lang aama:Afar}

tense	num	pers	gen	Beja	Afar
pres	sg	1		*ʔa-danbíil*	*asgaadeh*
		2	m	*danbiil- `a*	*tasgaadeh*
			f	*danbiil - `i*	- -
		3	m	*danbíil*	*yasgaadeh*
			f	*danbíil*	*tasgaadeh*
	pl	1		*n-eedbíl*	*nasgaadeh*
		2		*t-eedbil- `na*	*tasgaadeenih*
		3		*ʔ-eedbil- `na*	*tasgaadeenih*
past	sg	1		*ʔa-dbíl*	*usguudeh*
		2	m	*ti-dbil- `a*	*tusguudeh*
			f	*ti-dbil- `i*	- -
		3	m	*ʔi-dbíl*	*yusguudeh*
			f	*ti-dbíl*	*tusguudeh*
	pl	1		*ni-dbíl*	*nusguudeh*
		2		*ti-dbil- `na*	*tusguudeenih*
		3		*ʔi-dbil- `na*	*tusguudeenih*

Manipulate rows/cols of second paradigm –
Row: pers gen. Col: tense num

pers	gen	pres		past	
		sg	**pl**	**sg**	**pl**
1		*?a-danbíil*	*n-eedbíl*	*?a-dbíl*	*ni-dbíl*
2	m	*danbiil-`a*	*t-eedbil-`na*	*ti-dbil-`a*	*ti-dbil-`na*
	f	*danbiil-`i*		*ti-dbil-`i*	
3	m	*danbíil*	*?-eedbil-`na*	*?i-dbíl*	*?i-dbil-`na*
	f	*danbíil*		*ti-dbíl*	

Manipulate rows/cols of third paradigm –
Row: num pers gen. Col: lang tense

num	pers	gen	Beja		Afar	
			pres	**past**	**imperf**	**perf**
sg	1	com	*?a-danbíil*	*?a-dbíl*	*asgaadeh*	*usguudeh*
sg	2	m	*danbiil-`a*	*ti-dbil-`a*	*tasgaadeh*	*tusguudeh*
sg	2	f	*danbiil -`i*	*ti-dbil-`i*	- -	- -
sg	3	m	*danbíil*	*?i-dbíl*	*yasgaadeh*	*yusguudeh*
sg	3	f	*danbíil*	*ti-dbíl*	*tasgaadeh*	*tusguudeh*
pl	1	com	*n-eedbíl*	*ni-dbíl*	*nasgaadeh*	*nusguudeh*
pl	2	com	*t-eedbil-`na*	*ti-dbil-`na*	*tasgaadeenih*	*tusguudeenih*
pl	3	com	*?-eedbil-`na*	*?i-dbil-`na*	*tasgaadeenih*	*tusguudeenih*

6. Towards an Interface

What's not there (yet) . . .

- The interface for paradigm manipulation
  - Past prototype: Mozilla-Firefox XUL
  - The future: Cappuccino
    - http://www.cappuccino-project.org/

What could this lead to?

1. Agreement on format for exchange of paradigm data (JSON?).
2. Controlled (and inter-translatable) vocabularies for morphosyntactic attributes and values.
3. Uniform query language: SPARQL.

Given 1. & 2. (and possibly 3.), any number of display formats, datastore architectures, and front-ends can be constructed.