AAMA: Afro-Asiatic Morphological Archive

Architecture:

- Data: RDF (http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/) , RDF Schema (http://www.w3.org/TR/rdf-schema/);
- Backend (web services): Clojure (http://clojure.org), Google App Engine (Java) https://developers.google.com/appengine/), Appengine-magic (https://github.com/greynolds/appengine-magic)
- Backend (triple store): Jena (http://jena.apache.org/index.html), Fuseki (http://jena.apache.org/documentation/serving_data/) SPARQL (http://www.w3.org/TR/sparql11-overview/)
- Frontend: Cappuccino (http://www.cappuccino-project.org)

Currently, the authoritative data is coded in XML files, from which Turtle RDF is generated (since Turtle is readable), from which RDF/XML is generated (since RDF/XML is not very readable), which is loaded to the triple store.

Tools:

XML/XSL: saxon; xalan;

RDF: Jena Eyeball - “lint” for rdf; several online RDF syntax checkers are available (e.g. http://www.rdfabout.com/demo/validator/)
rdf2rdf for conversion between RDF formats
Turtle (http://www.w3.org/TR/turtle/)
JSON-LD (http://www.w3.org/TR/2013/WD-json-ld-20130411/)
Shell scripts (Bash);

SPARQL: Sparql requests can be submitted using Curl, but Fuseki includes a set of CLI tools that make this much easier (“SOH” - SPARQL Over Http http://jena.apache.org/documentation/serving_data/soh.html)

HTTP: Curl (http://curl.haxx.se) is indispensible for testing http requests and responses. Firefox LiveHTTPHeaders is also very useful. Browsers these days (Chrome, Safari, Opera) also have good HTTP debugging facilities.

Clojure: a lisp dialect targeting the JVM. leiningen is a “Make” replacement for proj mgmt. Emacs is the preferred editor (but others available). nrepl and ritz provide sophisticated dev/debug facilities.

Appengine-magic is a clojure wrapper around the Google App Engine API (new version under development)

Frontend: Cappuccino is a web application framework based on Apple’s Cocoa framework. Objective-J is an implementation of Objective-C (the official Apple dev language) in Javascript. Most of the Apple documentation applies to Objective-J and Cappuccino as well.

For testing we use Chrome, Safari, and Firefox.