

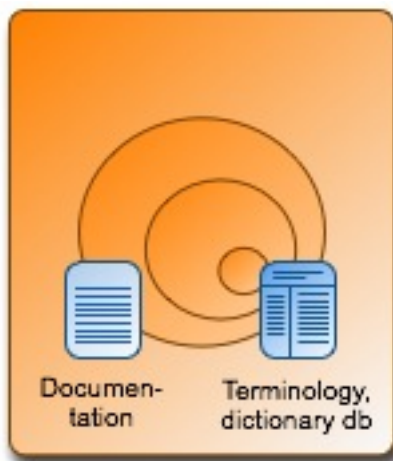
Architecture

Table of contents

1 Overview.....	2
2 Server.....	3
3 Database.....	3
4 Terminology.....	3

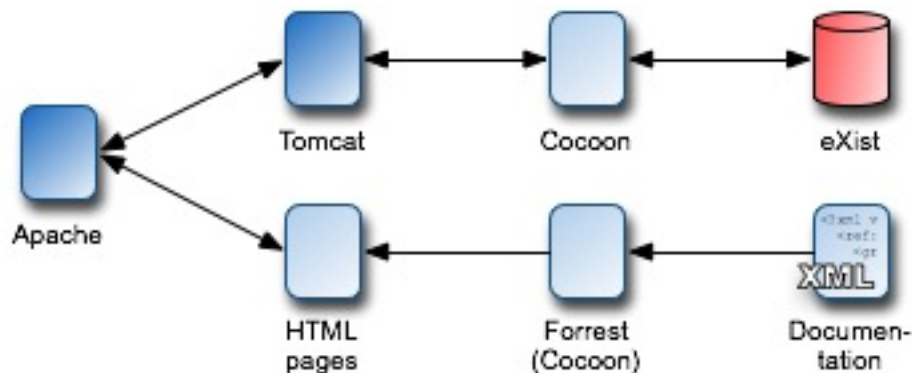
1. Overview

The Sámi dictionary and terminology portal is divided into two main parts: the documentation section, and the database proper. Each of these two parts are implemented as web applications (webapps) deployed within the application server [Tomcat](#). Thus, it can be depicted like below.



Portal/webapp server
basic components

Both components are implemented using [Cocoon](#), the documentation part through [Forrest](#), and the database using the Cocoon delivered with the database. The database itself is built using [eXist](#), a native XML database. Tomcat is running in tandem with the Apache web server such that Apache will serve static files, and Tomcat dynamically generated files, at least that's the idea. The complete setup is as follows:



portal components

All content and database files are written in XML, and transformed to HTML on the fly using Cocoon. Cocoon's basic design goal of separating the concerns of logic, structure and presentation allows the visual design of the portal to be largely independent of the

structure and logic of the documents and the database. This eases maintenance and makes it easy to change the presentation.

Using a native XML database like eXist means that one can work with one type of data (XML) throughout the application, and will ease exchange of data a lot. As (parts of) the content is planned to be available for download and use by e.g. translators, and it will also be exchanged with [the language technology project at University of Tromsø](#), the data exchange aspect is important.

2. Server

The server is a Windows computer running Windows 2000 Server, with 512 Mb RAM. Presently the default software coming with the OS is installed, and the software used by the portal (Apache, Tomcat, and the webapps), and all default processes are running. To make the server faster, one can consider turning off all non-used services to save processor cycles and memory.

3. Database

4. Terminology