

Table of Contents

.....	1
<u>The Alexis Server And Client Components (Chakriya and Columbine)</u>	1
<u>INTRODUCTION</u>	1
<u>AN OVERVIEW OF THE ARCHITECTURE</u>	1
<u>A PROCEEDURAL DESCRIPTION OF A REQUEST AND RESPONSE</u>	2
<u>SOFTWARE COMPONENTS</u>	3

▪

The Alexis Server And Client Components (Chakriya and Columbine)

please send comments and questions to: matthew@ella-associates.org

INTRODUCTION

This document attempts to explain some of the links between the classes used by the Chakriya Server. This is the server component of the Alexis program. Alexis is a program which aims to allow lexicographers to collaborate in the creation of dictionaries. Alexis attempts to harness the power of Java and the Internet to make this possible. This document is not complete.

AN OVERVIEW OF THE ARCHITECTURE

Alexis is a Java Based Client/Server Application. The Client component is based upon the NetBeans IDE and Development Platform. The Client and Server communicate with an XML based protocol which is transported over HTTP. It was decided to use an XML on HTTP based protocol in order to avoid problems involving proxy servers and fire-walls. The server component is a custom java application which communicates to the client using a Java Servlet running on Apache Tomcat.

All data for the application is stored in a MySQL Database running on a Debian Linux System. In some instances, the data is stored within the MySQL Database in plain text fields which contain XML. The Server Component provides a hierarchical file-system type 'view' of the underlying database. In other words, the client is able to interact with the data as if it were a file-system. Each retrieval query is represented by a unique file-path.

A PROCEEDURAL DESCRIPTION OF A REQUEST AND RESPONSE

When the Alexis Client makes a request of the Server it is currently sent to the URL:`http://ella-associates.org/alexis/alexisServer/`

This URL contains a Java Servlet running under the Apache/ Tomcat Web-server. The servlet is currently located in the development directory structure at `/alexis/webApp/WEB-INF/classes/ChakriyaServletWrapper.java`

To achieve its work the ChakriyaServletWrapper uses three classes ChakriyaManager, ChakriyaCommunicator, ChakriyaCommunication

The ChakriyaCommunication class encapsulates the XML text data that is sent back and forth between the Alexis Server (Chakriya) and the Alexis Client. The ChakriyaCommunicator only has one significant method, namely its 'communicate' method. This method examines the ChakriyaCommunication object to determine if the request type is a Login Request or a FileSystem Request. Based on this evaluation the ChakriyaCommunicator then passes the ChakriyaCommunication object to the correct 'processor' class, either the 'LoginProcessor' class or the 'FileSystemRequestProcessor' class.

The 'communicate' method then calls the appropriate methods on the processor class to ensure that a response from the Server is generated. The 'communicate' method then places that response in the 'ChakriyaCommunication' class.

Usually the requests are parsed into 'beans' which in this context are Java Classes with the same structure as the Xml document which constitutes the request. For example, a 'login request' is parsed into a 'requestBean' Object, which is a property of the RequestProcessor class. This parsing is done by the 'setSourceNode()' method of the RequestProcessor class.

It is possible to see the XML protocol which is being passed back and forth from the Alexis client and the Chakriya Server by using the 'Output Window' which is in the Client application. Currently there are 4 main types of requests: Login Requests and 3 types of 'File-system Requests'

A Login Request, obviously, is sent by the Alexis Client (Columbine) in order to initiate a session for a particular user. The Login Request transports a user-name and password. The Login Response sent by the Alexis Server, either:

- validates the user-name and password and sends back a 'session-id' number or,
- Informs the client that the user-name is valid or,
- Returns nothing, (when neither item is valid)

The 3 types of 'file-system' request correspond to the actions of retrieving a list of sub-folders, retrieving a list of word data, or updating modified word data.

SOFTWARE COMPONENTS

Java 1.4

ava.sun.com

Netbeans Java IDE and Development Platform

www.netbeans.org

MySQL Relational Database Management System

www.mysql.com

Apache Tomcat Java Servlet Engine.

www.apache.org

please send comments and questions to: matthew@ella-associates.org

Author: m.j.bishop