

CSC6362 — System/Software Architecture

Fall 2001

Project III: A Growing Web Search Engine

Due: 5:30pm Thursday December 6 (in class)

“... All the other arts were obedient and submitted to the discipline of architecture.” Victor Hugo

— “The Art of Systems Architecting”.

I. Summary

As system/software architects of a renowned company, your team is to architect a web search engine, **Cyberminer****, using **KWIC***** which shall be an extension to the **KWIC**** software system you implemented as part of Project II. For this project, however, you will use an Implicit Invocation architectural style in building a Java applet, instead of an OO style. **Cyberminer**** shall be accessible through your own web page.

II. *Cyberminer*** - A growing web search engine

Functional Requirements:

Cyberminer** shall accept a list of keywords and return a list of URLs whose descriptions contain any of the given keywords. No noise word shall be given as part of the list of keywords.

Cyberminer** shall use another software system, **KWIC***** - a KWIC (Key Word in Context) index system, in order to efficiently maintain a database of URLs and the corresponding descriptions. **KWIC**** shall accept an ordered set of lines, where each line consists of two parts:

- *the URL part* (e.g., <http://www.utdallas.edu>)
- *the descriptor part* is an ordered set of alphanumeric words,

where each word is an ordered set of characters excluding any punctuations symbols or control characters. The descriptor part of any line shall be “circularly shifted” by repeatedly removing the first word and appending it at the end of the line. The KWIC index system shall output a list of all circular shifts of the descriptor parts of all lines in descending alphabetical order, together with their corresponding URLs. No line in the output list shall start with any noise word such as “a”, “the”, and “of”.

KWIC*** shall allow for two modes of operation: i) for building an initial KWIC indices; and ii) for growing the indices with later additions.

Non-Functional Requirements:

Cyberminer** shall be easily understandable, portable, enhanceable and reusable with good performance. **Cyberminer**** must also be user-friendly, responsive, and adaptable.

III. The Deliverable

Your description should be elegant and comprehensible. Your deliverable should be available as both on-line and off-line specifications:

- 1. Architecture** Describe i) an implicit invocation style of the software architecture; and then map it into an OO style. For both styles, describe both pictorially and textually the constituents of the architecture, namely, components/elements, interactions/connections, any constraints, any patterns. Also present a discussion of advantages and disadvantages of your architecture.
- 2. Java applets** Build an applet which shall run with a web browser (preferably with recent versions of both NetScape and IE).
- 3. User Manual** Briefly describe the typical interactions between the user and the system, e.g., what are the steps the user has to follow in using the system. Also describe any operating requirements. The manual shall use screendumps whenever possible.