Parametarization of DIENST Metadata Search Fields

SARANTOS KAPIDAKIS* PANAGIOTIS ALEXAKOS† HARIKLIA TSALAPATA‡

April 19, 1999

Abstract

Digital library software provides advanced ways of on-line material search, retrieval, and presentation. In order to support collections of different types of objects such as text documents (e.g. technical reports), images, audio, and others and to increase the applicability and use of digital library software worldwide search tools must have the ability to search for and retrieve information from any digital library independent of the metadata that describe it's contents. DIENST[1] is a digital library distributed search tool developed at Cornell University. A limitation of DIENST is that it is tailored for digital libraries of technical reports and requires code modifications to become applicable to digital libraries holding other types of objects.

This report presents an efficient and user friendly extention of DIENST for managing digital libraries whose contents are described by any set of metadata fields, including libraries whose individual repositories are described by different sets of metadata.

1 Introduction

The importance of digital libraries becomes more obvious every day. Digital libraries hold material online, in a digital form, and provide advanced ways for material search, access, and presentation. In addition, they support distributed search over collections of digital data located all over the world. Users of the libraries can retrieve the data they are interested in and study digital copies of them without ever having to visit the building of the library itself.

The increasing number of libraries holding a variety of objects (e.g. text, images, audio, video, etc.) emphasizes the importance of a powerful digital library search engine that is able to locate and retrieve objects of different types and formats that may be described by different sets of metadata fields. This report presents our work in developing such a tool. The work is based on DIENST [1], a search tool for distributed digital libraries of technical reports developed at Cornell University and used by the Networked Computer Science Technical Report (NCSTRL) library. While keeping the basic functionality of the original version of DIENST our approach extends it to enable search for objects described by any set of metadata fields. We call this new version of DIENST "parametarizable" because it enables digital library administrators to specify the set of metadata fields that describe library objects as parameters.

The organisation of the report is as follows. Section 2 gives a brief description of how DIENST works by describing the steps followed to answer a query. Section 3 describes the

*Institute of Computer Science, Foundation for Research and Technology – Hellas, P.O Box 1385 Heraklion 71110, Greece. E-mail: `sarantos@csi.forth.gr` Tel: +30-81-391678, Fax: +30-81-391671. (contact person)

†Institute of Computer Science, Foundation for Research and Technology – Hellas, Heraklion 71110, Greece. E-mail: `jalexak@csi.forth.gr`

‡Institute of Computer Science, Foundation for Research and Technology – Hellas, Heraklion 71110, Greece. E-mail: `htsalapa@csi.forth.gr`

[1]Multilingual and other versions maintained by the Institute of Computer Science, Greece, can be found at http://www.csi.forth.gr/~dienst

features added to the original version of DIENST to construct the new parametarized version. Section 4 gives some implementation details and discusses how the original DIENST code was modified to support the new features. Section 5 explains the steps that must be followed to configure and run a parameterized DIENST server as well as to run several parametarized DIENST servers in a distributed environment. Finally, section 6 concludes the report.

2 A brief description of DIENST functionality

DIENST is a client-server application. Users contact and query the DIENST server through a web browser. The server replies with a list of references to matched objects that are presented to the user through the web browser. Each object of the digital library must be "registered" with the DIENST server. Registration of an object makes the server aware of it's existence and allows it's on-demand access. Each registered object may be available in several digital formats and has *metadata* associated with it. Metadata (or "data for data") is information describing the object. For example, the metadata for technical reports may include their title, author, and abstract fields.

Users query the system through keywords. DIENST uses a set of pre-built indexes (one for each metadata field) to match the keywords against the metadata and returns a list of references to relevant objects. Each reference contains a description of the corresponding object and a list of the formats the object is available in. The user can retrieve the object in one of the available formats.

To enable DIENST to locate digital library objects independently of the metadata set that describe them we introduced a new configuration parameter that holds the metadata fields of each collection.

When two or more DIENST servers are available in a distributed environment the search proccess is done by dispatching one process for each DIENST server in the system and querying all DIENST servers in parallel. Each DIENST server searches it's local collection and returns references to matched objects to the user. For this type of distributed search additional software is required. This additional software is called a "metaserver". It provides each DIENST server with information about the other servers available in the system.

As will be descibed later, in the parameterized version of DIENST the metaserver code was extended in order to provide the additional information of the fields of metadata supported by each DIENST server.

Apart of the above additions and modifications, the functionality of the original version of DIENST was maintained. The DIENST version that was used as the base for this work is the original version developed at Cornell University enhanced with multilingual search features by ICS-FORTH.

3 Features added

3.1 Features

As mentioned above, the main feature introduced in the new version of DIENST is the ability to search digital libraries independetly of the metadata set that describes the stored objects and including digital libraries whose individual collections are described by different sets of metadata. The DIENST server administrator needs only to create a new configuration file in the *Config* directory named "*Collections_Fields.config*". The file holds the metadata sets of each collection of the digital library. After creating this configuration file, the administrator should rerun the configuration script *auto_config.pl*. More information on the configuration of a parametarizable Dienst server is provided in section 5.

The second important feature supported by this new version is that two or more DIENST servers operating on digital libraries whose metadata sets are different can cooperate in a distributed manner, each of them being able to send queries and receive answers from the others. The following paragraphs describe the query process in the new version of DIENST.

DIENST has two ways of searching: fielded search and simple search.

- **Fielded search**: In fielded search the user specifies keywords for specific metadata fields. The search engine matches the keywords against the metadata field indices. In a distributed environment including several collections the user can select the collections to be queried. Queries can be posed on the intersection of the metadata fields of the selected collections. If the intersection is the empty set then the search is performed through the "*anything field*" that will be discussed at the end of this subsection. For example, let's assume a system with three collections, the first one having *title*, *author*, and *abstract* as metadata fields, the second one *title* and *author*, and the third one *title* and *abstract* (*figure 1*). When the user wants to search the first two collections only, he is presented with the option of specifyng values for *title* and *author* (the intersection of the metadata fields of the first two collections) since it is meaningless to attempt to search the second collection on *abstract*. Similarly, when the user wants to search all three collections he can specify query keywords only for *title*, while if he wants to search the first and third collections he can specify query keywords for *title* and *abstract*.

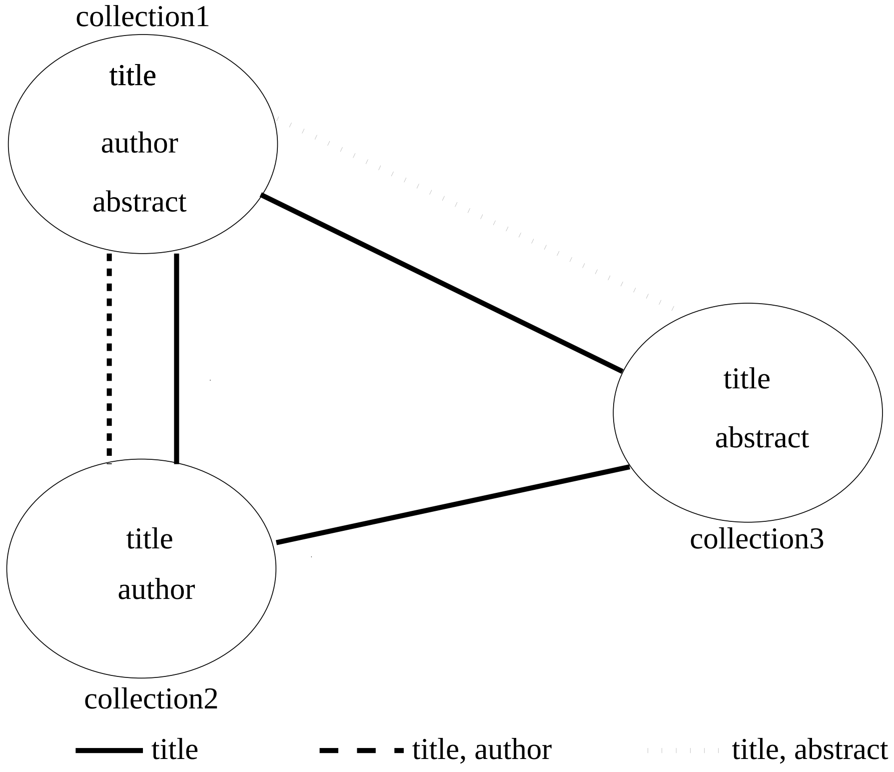

figure 1: Cooperation of three DIENST servers each of which operates on collections of different sets of metadata fields

- **Simple search**: In simple search the keywords specified by the user are matched by the search engine against the indexes of all metadata fields in all collections available in the system.

The last feature added to DIENST is the "*anything field*". The DIENST server handles the anything field as a regular metadata field. The index of the anything field is created by merging the indices of all metadata fields. As an example, the results of querying on the word *test* in the *anything field* will be the same as the results of a simple search on *test*.

3.2 New user interface

In order to support the second of the above features the user interface of fielded search was extended with two additional buttons ("Show fields common to the selected collections" and "Show fields common to all collections") as displayed in figure "2".

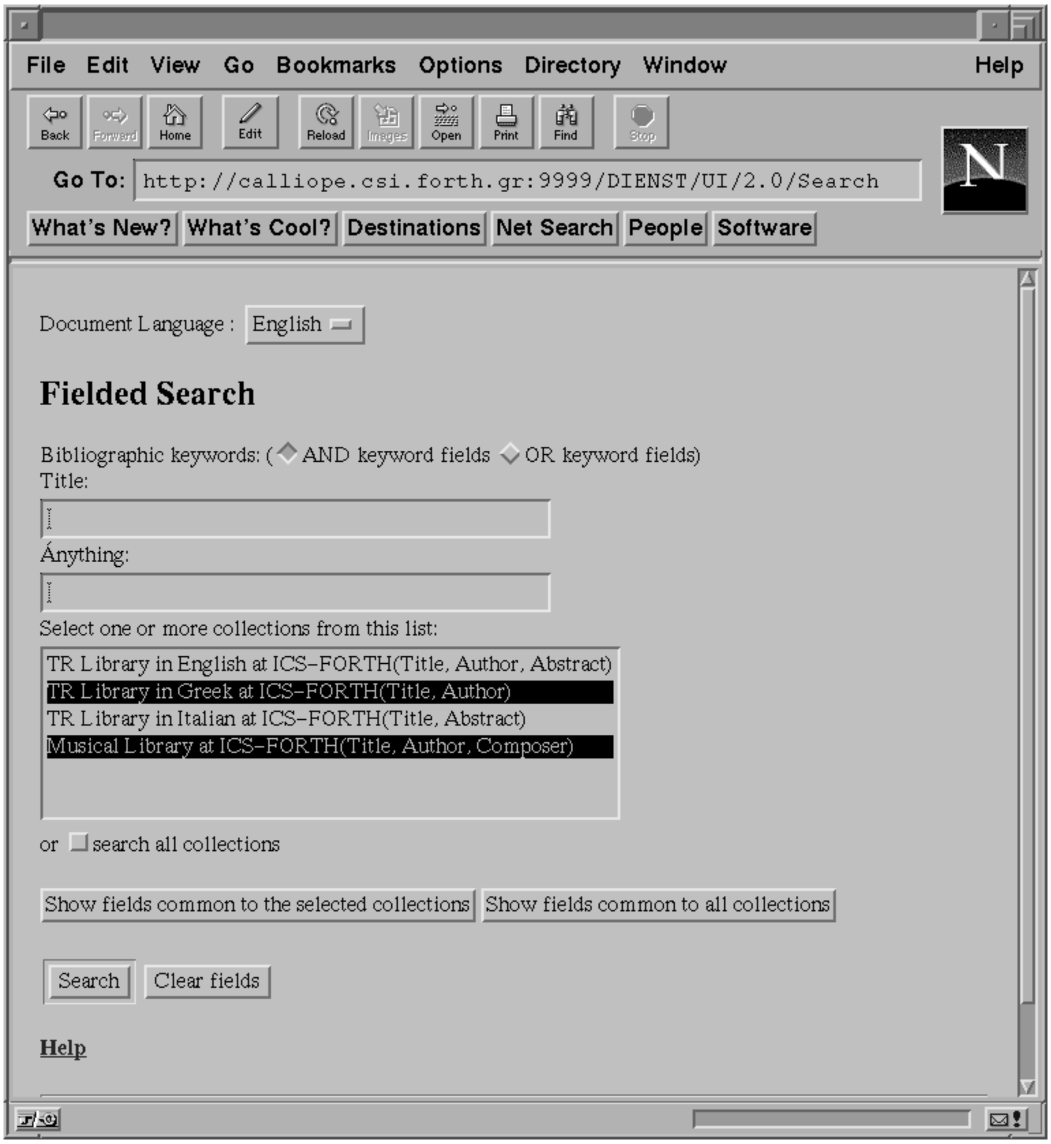

figure 2: The new DIENST interface with the two additional buttons

The user can query on a subset of the collections available in the system. After selecting the collections to be queried, the user can click on the "Show fields common to the selected collections" button to see the intersection of the metadata fields of the selected collections. The user can return to the beginning state, displaying all collections in the system and their common metadata fields, through the "Show fields common to all collections" button. Note that if all metadata fields of some collection e.g "A" are in the intersection of the metadata fields of the collections that the user selected, the collection will appear in the list of collections after pressing the "Show fields common to the selected collections" button. All collections except "A" will be selected (highlighted) in the new page. This is displayed in figure "3" which shows the result of pressing the "Show fields common to the selected collections" button on the screen displayed in figure "2".

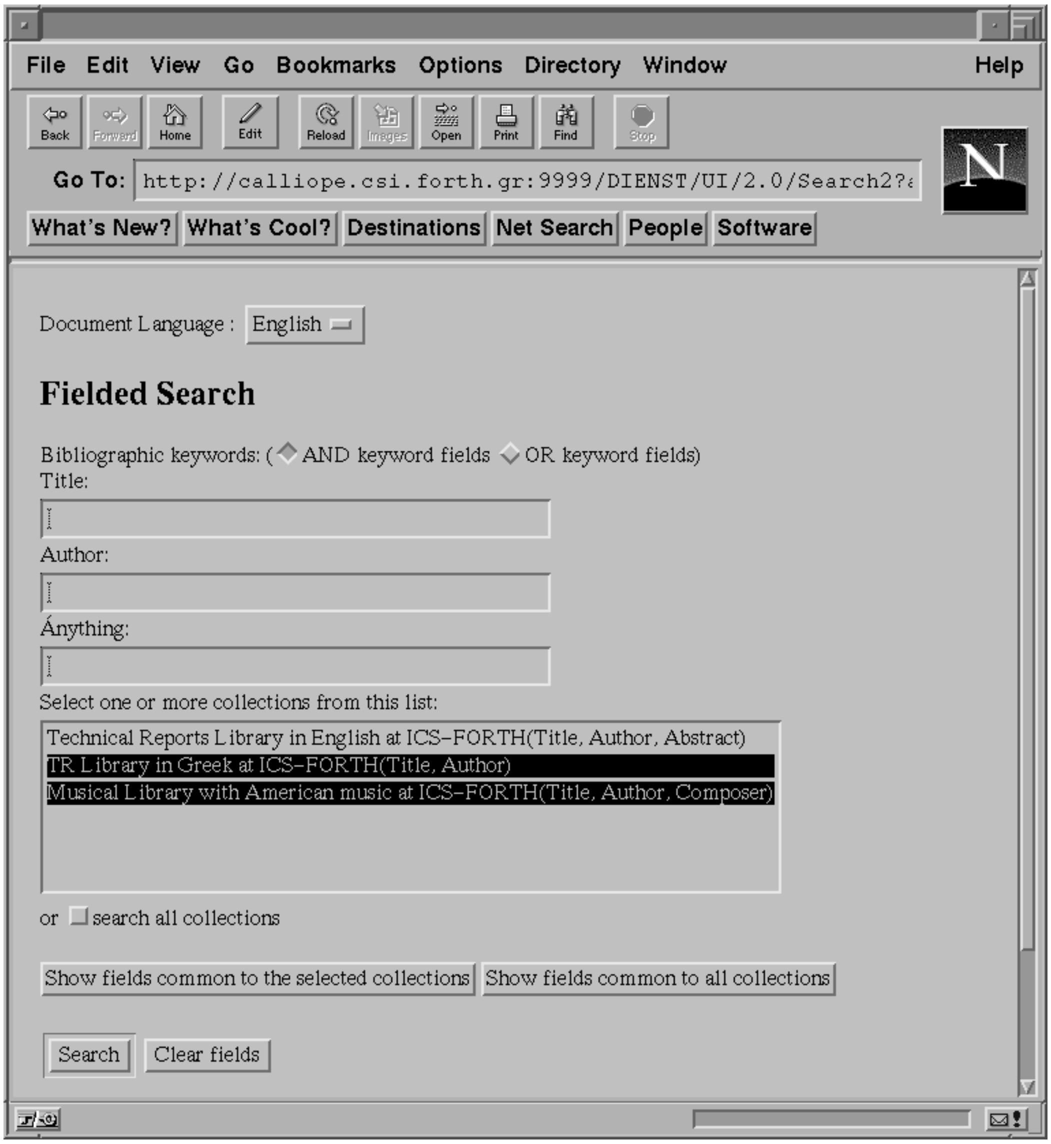

figure 3: The result of "Show fields common to the selected collections" button's use

3.3 Backward compatibility

The parametarizable version of DIENST is not backwards compatible with the original version. This means that the parametarizable version of DIENST cannot cooperate with DIENST servers running the original software in a distributed environment. Even though the new version of DIENST has protocol and message exchange compatibility with the original version, problems arise when a server running the original version attempts to display hits received from a parametarized server through a distributed query. Since servers running the original version support only three fields of metadata (title, author, and abstract) they do not recognize the format of other metadata fields. In addition, the original software does not support the "*anything field*".

4 Implementation

4.1 Code Modifications

Code changes and additions to the source code are written in PERL. Some additional configuration variables have been introduced to support parametarization. Variable %fields_type holds the searchable metadata fields. It is added to file *Config/config_constants.pl* by the configuration script *Config/auto_config.pl*.

The metadata field sets of the collections supported by a DIENST server are specified in the hash table %authority_fields defined in file *Config/config_constants.pl*. The keys of this hash table are the names of the collections and the key values the corresponding metadata fields seperated by a "&". The information about the metadata fields of each collection is taken either from file *Config/Collections_Fields.config*, or from the metaserver (the metaserver modifications will be discussed later in this section).

The support of the "anything field" required some modifications to the indexer code. Code has been added to files *Indexer/build_inverted_indexes.pl* and *Indexer/parse_bib_file.pl*. The new code merges the index files of all metadata fields and stores the result to file *Indexer/Indexes/anything Words.ind*. This method was implemented in order to make the "anything field" behave as a regular metadata field.

Finally, in order to support the multilingual extentions to DIENST implemented at ICS-FORTH the parametarizable version was based on the multilingual code. The parametarizable version supports the configuration parameters introduced by the multilingual version: the variable %trans_fields which, like variables %fields_type and %authority_fields, is defined in file *Config/config_constants.pl*. This hash variable has as keys the names of the metadata fields and as values their translations seperated by a "#". These translations, like the information about the metadata fields of each collection, are taken either from file *Config/Collections_Fields.config* or from the metaserver.

4.2 Metaserver Extentions

The original DIENST metaserver answered three queries regarding the DIENST servers in a distributed system: Publishers, Repositories, and Indices. The original metaserver has been extended to also provide information regarding the metadata fields of each DIENST server. The new query is: */Dienst/Meta/2.0/Collections*. Through this query each server receives the metadata fields of the other servers in the system and stores them in variable %authority_fields defined in file *tmp/dienst_meta_data.pl*.

4.3 Modification in the transmission of queries

- **Fielded search**: In fielded search querying is performed in exactly the same way as with the original version. The only difference is the addition of the "Show fields common to the selected collections" button to the fielded search screen. Clicking on this button sends a query to the DIENST server with parameters the names of the selected authorities (collections). The result of the query is the intersection of the corresponding metadata field sets. If the user has entered "authorities == All" the intersection of all collections is returned. As an example, the request

$$/Dienst/UI/2.0/Search?authorities = english : italian$$

will return the common fields of the collections named "english" and "italian". You can see that the two authorities are seperated with a ":".

- **Simple search**: In the original version of DIENST simple search was performed as a fielded search query where the keywords entered by the user were matched against all metadata fields. In this version the existance of the "anything field" allows the implementation of simple search as a search on "anything". This provides the following benefit: the older technique for simple search required retrieving the metadata fields of each DIENST server in the system, then creating a separate query for each server based on the server's metadata fields. Using the "anything field" the same query can be sent to all servers. For example, consider a simple search with keyword "*test*" against two dienst servers with metadata fields *title* and *author* for the first server and *singer* and *composer* for the second. The older technique requires the query $/Dienst/UI/2.0/Query/?boolean = or\&title = test\&author = test$ to be sent to the first server and the query $/Dienst/UI/2.0/Query/?boolean = or\&singer = test\&composer = test$ to be sent to the second. The new method will send the query

$$/Dienst/UI/2.0/Query/?boolean = or\&anything = test$$

to both servers.

5 Configuring a new distributed DIENST environment

The process of configuring a new dienst server of the parameterized version of DIENST consists of the following steps:

- Create a file *Config/Collections_fields.config* with the metadata fields of each collection. This file consists of records like the following:

  PublisherID = english
  Publisher_name = Technical Reports Library in English at ICS-FORTH
  AuthorityID = english
  Fields = 1:title#Title##&2:author#Author##
  Descr_Fields = 11:abstract#Abstract##

  The first three lines define the PublisherID, Publisher_name and AuthorityID of the collection. The values of PublisherID and AuthorityID are always the same. The last two lines define the metadata fields of the collection. "Fields" defines the non-described fields and "Descr_Fields" the described fields. Non-described fields are the fields that are displayed in the query results html page. Described fields are displayed only if the user wants to see more details about a search hit. These two lines have the same format. They are lists of fields seperated by a "&". Each field is a multilingual string. A multilingual string is a list of strings seperated by a "#". The first of these strings is the name of the metadata field and the rest are the translations of the field to the languages supported by the collection [2]. The number that appears at the beginning of each field and is delimited by a ":" defines the order in which the field will appear in the search results page.

The user may define a "default" entry in *Collections_Fields.config* that will be used for all collections in the system that are not defined in *Collections_Fields.config*. The default entry has "*" as the PublisherID and AuthorityID. Fields, Descr_Fields, and Publisher_name are defined as above. Following is an example *Collections_Fields.config* file.

PublisherID = english
Publisher_name = Technical Reports Library in English at ICS-FORTH
AuthorityID = english
Fields = 1:title#Title # #&2:author#Author # #
Descr_Fields = 11:abstract#Abstract # #

PublisherID = greek
Publisher_name = Technical Reports Library in Greek at ICS-FORTH
AuthorityID = greek
Fields = 1:title#Title ##&2:author&3:composer#Composer##
Descr_Fields = 11:lyrics#Lyrics##

*PublisherID = **
Fields = 1:title&3:composer&4:lyricswriter&5:singer#Singer##&6:abstract
Descr_Fields = 2:lyrics&12:instruments#Instruments##

This example file includes definitions for two Collections, *english* and *greek*, with metadata fields *title*, *author*, and *abstract*, and *title*, *author*, *composer*, and *lyrics* respectively. It also includes a default entry at the end of the file.

- Run script *Config/auto_config.pl*

In a distributed system an additional file must be created. The file is read by the metaserver and is named *Collections*. It follows the format of *Collections_Fields.config* but does not include a default entry (File Collections is included in directory dienst/LOCAL_META_DATA).

6 Conclusions

Digital libraries can store a variety of object types. The type of objects stored in a digital library and the purpose of the library may result in a variety of metadata field sets to describe the objects. A distributed search tool that operates on digital libraries described by a variety of metadata fields, including collections described by different sets of metadata, is essential. This report describes an approach for parametarizing the metadata fields of a DIENST server. The approach has been implemented at ICS-FORTH. The report provides implementation details and instructions on configuring the parametarizable version of DIENST.

The parameterizable version of DIENST has been tested under the Netscape 3.0 browser. Other browsers may be used provided that they support JavaScript.

References

[1] Carl Lagoze and Jim Davis: *"DIENST: An Architecture for Distributed Documents Libraries"*, Communications of the ACM, 38(4), April 1995, p. 47.

[2] Sarantos Kapidakis, Iakovos Mavroidis, Hariklia Tsalapata: *"Multilingual Extensions to DIENST"*, FORTH-ICS TR-248, May 1999.