J. Plasma Fusion Res. SERIES, Vol.7 (2006) 343-347

Automation of Plasma-Process Fultext Bibliography Databases: An On-Line Data-Collection, Data-Mining and Data-Input System

SUZUKI Manabu, PICHL Lukas[1], MURAKAMI Izumi[2], KATO Takako[2] and SASAKI Akira[3]

University of Aizu, Aizu-Wakamatsu 965-8580, Japan

[1] *International Christian University, Tokyo 181-8585, Japan*

[2] *National Institute for Fusion Science, Toki 509-5292, Japan*

[3] *Japan Atomic Energy Research Institute, Kyoto 600-8216, Japan*

(Received: 4 October 2004 / Accepted: 6 December 2005)

Abstract

Searching for relevant data, information retrieval, data extraction and data input are time- and resource-consuming activities in most data centers. Here we develop a Linux system automating the process in case of bibliography, abstract and fulltext databases. The present system is an open-source free-software low-cost solution that connects the target and provider databases in cyberspace through various web publishing formats. The abstract/fulltext relevance assessment is interfaced to external software modules.

Keywords:

atomic and molecular data, bibliography database, download robot, data retrieval

1. Introduction

Comprehensive and reliable databases of atomic and molecular data are indispensable, for instance, in accurate computer simulations of energy dissipation in fusion edge plasmas. In addition, the range of users is broadening, including astronomy and recently also experimental medicine techniques (hadron therapy or positron emission tomography). This brings new requirements on the scope of the databases and classification methods for the covered data.

The standard methodology for data collection has been the literature search, classification of bibliography output, paper retrieval, relevance evaluation, recognition of the atomic / molecular process and other classification fields, identification of relevant graphs and numerical data tables, digitalization of numerical data (whether by retyping these manually or facilitated by character recognition software) and data input. Data production activities by theorists and experimentalists in adjunct working groups at data centers exclude literature search and character recognition issues. Due to considerable human labor involved, the above described process for maintaining a specialized database, such as atomic and molecular processes in edge plasmas, has been very costly. In addition, outsourcing the database solution to software companies associates with large extra costs, if the data structure is to be extended or altered (*e.g.* adding synonymous labels for chemical species involved in a collision process).

We therefore study the automation of database development and maintenance along two lines. The first is a test of the open-source and free-software database alternative to commercially maintained database systems. The latter deals with automation of data collection, classification, and input, and builds on the former one, since this kind of testing and automation is usually impossible within a commercial database product. The covered data originate from our joint working group activities at NIFS [1]. Our restriction to bibliography data, although including abstract and fulltext fields, allows us to automate the entire data-handling process from the search phase up to the data input. The system is designed for network-accessible electronic articles and / or abstracts from various subscribed publishers; the available data in this kind of cyberspace are almost complete for articles published since 2000.

The paper is organized as follows. In Section 2, we discuss our open-source bibliographic databases in analogy with the outsourced NIFS system. We first provide a prototype bibliography database for electron-molecule collisions that combines chemical species and collision process search fields, then develop a source fulltext database for abstract / paper relevance assessment. Section 3 discusses the full automation of data-handling process in dynamically updated fulltext databases. In Section 4, automated data update, down-

Corresponding author's e-mail: lukas@icu.ac.jp

load robot protection and copyright issues are dealt with. Concluding remarks close the paper.

2. Bibliographic databases

The basic bibliographic entries are authors, journal, volume, issue, page number range, publication year; classification fields include method (theory, experiment), type of process, species involved, date of database entry and others; the fulltext feature covers abstracts (html formatted, as the special characters including Greek letters, superscripts etc. are very common) and article pdf files (external link or local disk copy).

Due to the size of atomic and molecular physics community, simultaneous database access rate is far below numbers common in electronic business etc. It is rare that more than 50 users connect at a time. Within this range, free software solutions fully compete with commercial software; also the requirements on hardware parameters are modest.

It was therefore decided to use a custom-built PC as a dedicated server. The hardware specification is as follows: Pentium 4 (FSB 533) 2 GHz CPU, 1 GB RAM and 120 GB HDD, which matches the needs of NIFS database users (large data set, limited access rates). The operating system is Linux (RedHat 9) running Apache 2 HTTP server. Database management system is MySQL 3.23.54 with the connecting logic layer for web interface written in PHP 4.2.2. MySQL was chosen since it suffices for our purpose and for its ease of manipulation; more query-rich object relational database management system, such as PostgreSQL, can also be used. The above operating system, web and database server, and programming language compiler are free-software open-source products. All databases discussed here are available online at the crdb.nifs.ac.jp server.

Figure 1 shows a snapshot of our test bibliography database [2] for electron-molecule collisions as described above. In addition, a combined search based on molecular species and collision process has been inbuilt [3] and the source code is available online [4].

In order to facilitate development of relevance-assessment software for journal articles [5], we have also created an abstract and fulltext database [6] which stores 379 papers [7] previously evaluated for AMDIS database at NIFS (includes both articles accepted and rejected for entry to AMDIS). The list serves for calibration of learning vector quantization method (LVQ) by our collaborators [5]; the source code is also available on-line [4]. The major publishers covered in the list are APS, IOP and EDP Sciences. The abstracts in html format have been downloaded from databases of the three publishers; the data [7] are extracted from the html files and input by using PHP string matching scripts for automatic handling as described in the next section.

3. Automated data handling

There are two principal stages in building and maintaining bibliography databases: (1) retrieve, evaluate and input all relevant data until the set point t_0, and (2) retrieve, evaluate, extract and input data in regular intervals $t_0 + nT$. Focusing on the latter, the flow chart of the procedure we implemented follows:

1. Install initial skeleton for bibliography and fulltext database (Linux system in Sec. 2)
2. Identify on-line publisher database locations and the respective form of DB queries (APS, IOP, EDP at present)
3. Create a set of boolean queries recognized as relevant for the maintained Linux database (updated via on-line query-creation and query-management form interface; cf. Fig. 2)
4. Create interface script (PHP) that transforms queries in 3 to the form of on-line queries in 2.
5. Send the transformed queries to publisher sources and retrieve html output by using the wget command line software in regular time intervals (cron command on Linux server).
6. Follow the abstract and fulltext links in query output forms, download,
   - Call the interfaced software module for relevance assessment judgement: accept / discard the article entry;

   and concatenate to MySQL database tables (wget and PHP).
7. Extract relevant information from the query output html-formatted source and concatenate to MySQL data tables (PHP).
8. Wait time T, then continue from item 5.

In Fig. 2, the on-line database of APS (a) and IOP (b) are shown. Given the form of scientific articles and journals, the structure of available queries is practically identical. In addition to search fields and logical operations, there are search options: threshold, sorting, or number of records per page. The query range can be limited by fixing the publication date period or volume / issue interval; hits can be specified either as a word stem or exactly as a quote. The query transform script (item

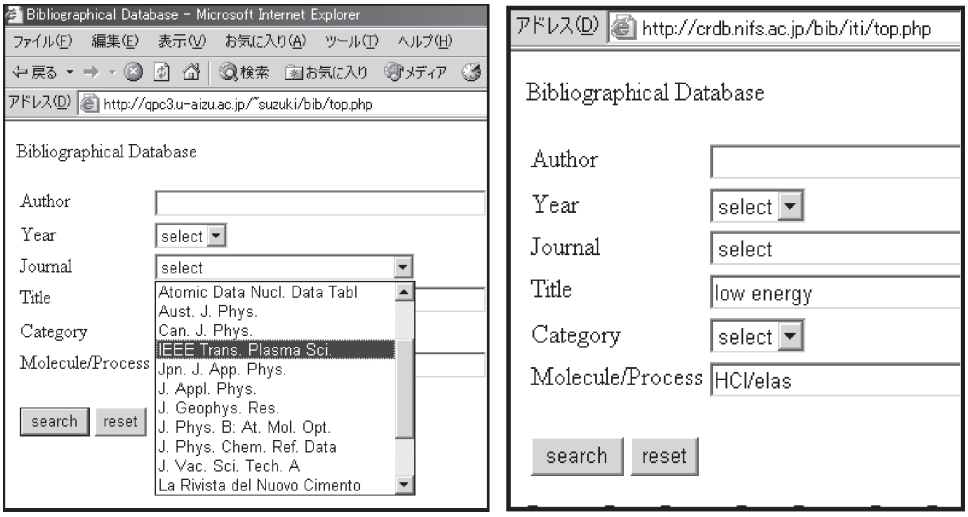


Fig. 1 Bibliography database for electron-molecule scattering designed by Prof. Y. Itikawa (NIFS) [2]. A combined search by molecule species and / or collision process is enabled. The system is implemented as a free-software open-source solution (the source code and installation instructions are available form from [4]).

4 in the flow chart above) is rather straightforward.

Nonetheless, the query output in Fig. 2 varies greatly, especially from the viewpoint of HTML code in output files (formatting, commercials, itemizing styles). Therefore the extraction of relevant database fields from the pool of files downloaded across publishers is non-trivial. A part of PHP script to send the download queries is shown at the bottom of Fig. 2; also the string matching scripts to extract author, title, page number and other fields are written in PHP language.

Since creating new scripts for different publishers or after any web-interface update would be cumbersome, we have also created a simpler script-development interface in order to avoid programming in PHP at the user stage. The user selects itemizing marks on the input site (*e.g.* a "checkbox" field), and thus classifies the beginning and the end of an entry. Thus is how the particular parameters enter the PHP code skeleton. To save the expert work on relevance assessment of the downloaded article, a software module [5] interface is added in item 6 of the flowchart above. Hence the entire procedure becomes automated; human intervention is required only to (1) create initial set of relevant queries (select the keywords and logical operators) and (2) to use the PHP script development form (that analyzes html query output structure) when some new online database is added (or the html interface updates). Based on the analyzer scripts, we finally developed a joint database search form that sends query requests and collects query answers across publisher web sites [8].

4. Scheduling of article retrieval

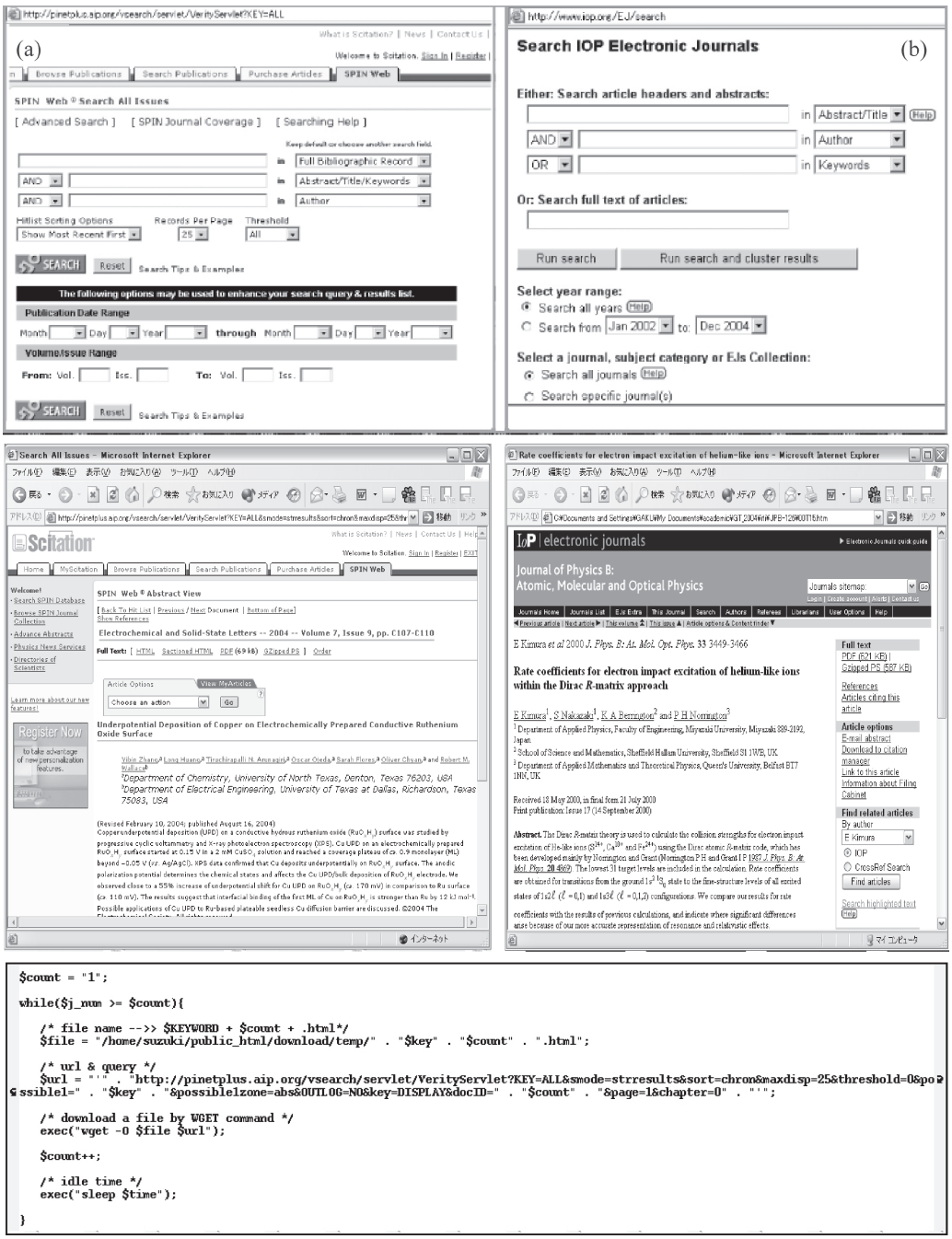
To avoid any kind of interference with download robot protection software on the side of publisher, the abstract and fulltext cannot be retrieved without intermissions. Although the retrieval rate is much restricted by query structure and generally fits within the conditions of subscription, zero-approaching interval of subsequent download requests often triggers download robot protection scripts on the side of publisher and may result in immediate suspension of the subscription service. We have therefore added artificial waiting intervals. The download interval τ is set by

$$\begin{aligned}&\text{function } \tau\{\\&\text{while(1)}\{\\&\quad t \leftarrow t_0 + 2T(r(\,) - 0.5)\\&\quad c \leftarrow f_m r(\,)\\&\quad \text{if}(c < f(t)) \text{ return } t\\&\ \}\\&\}\end{aligned} \tag{1}$$

where $0 \le r(\,) < 1$ is a uniform random number and $f(\,)$ is a probability density function with maximum value f_m. We set $t_m = 10.0\,\mathrm{s}$, $\delta = 3.5\,\mathrm{s}$, and use the normal distribution for the download time envelope profile, $f(\,) = N(t_m, \delta)$. Although this suffices for our purpose, an optimal solution would be to measure the access rate from within the particular subscription domain and replace $f(\,)$ with such a histogram of network connections.

5. Concluding remarks

We have reported our new free-software open-source bibliography databases for the National Institute for Fusion Science. These include electron-molecule collision process bibliography and a 379-item article database which calibrates the include - reject scope of AMDIS database at NIFS. In addition, we have developed an au-




```
$count = "1";

while($j_num >= $count){

   /* file name -->> $KEYWORD + $count + .html*/
   $file = "/home/suzuki/public_html/download/temp/" . "$key" . "$count" . ".html";

   /* url & query */
   $url = "'" . "http://pinetplus.aip.org/vsearch/servlet/VerityServlet?KEY=ALL&smode=strresults&sort=chron&maxdisp=25&threshold=0&po
ssible1=" . "$key" . "&possible1zone=abs&OUTLOG=NO&key=DISPLAY&docID=" . "$count" . "&page=1&chapter=0" . "'";

   /* download a file by WGET command */
   exec("wget -O $file $url");

   $count++;

   /* idle time */
   exec("sleep $time");

}
```

Fig. 2 Automated data-handling. Data-provider query forms: APS (a) and IOP (b), structure of html output files (middle), trigger and analyzer PHP script (bottom).

tomated Linux system that fully handles data search, retrieval, extraction and input into fulltext databases. Article assessment programs, such as LVQ algorithm, are conveniently interfaced in the form of a software module. The present work is considered useful in saving database maintenance costs. Further research will focus on the automation of data-handling for cross-section databases with graphical user interface.

L. P. acknowledges partial support by a Grant-in-Aid of JSPS and Academic Frontier Research Program of MEXT. This work was performed with the support and under the auspices of the NIFS Collaborative Research Program.

References

[1] I. Murakami, T. Kato, A. Igarashi, M. Imai, Y. Itikawa, D. Kato, M. Kimura, T. Kusakabe, K. Moribayashi, T. Morishita, K. Motohashi, L. Pichl, *AMDIS and CHART update I*, NIFS-DATA Series (ISSN 0915-6364), Issue 70, Oct. 2002.

[2] On-line database of electron-molecule scattering bibliography, http://crdb.nifs.ac.jp/bib/iti/top.php

[3] Electron-molecule collision bibliography data, Prof. Y. Itikawa, private communication.

[4] Source code and installation instructions (free-software, GNU licence) http://crdb.nifs.ac.jp/crdb/source/

[5] A. Sasaki, K. Joe, H. Kashiwagi, C. Watanabe, M. Suzuki, L. Pichl, M. Ohishi, D. Kato, M. Kato and T. Kato, *Design and implementation of an evolutional data collecting system for the atomic and molecular databases*, Joint ICAMDATA and ITC14 conference 2004, Toki, Japan.

[6] On-line database of input abstracts for relevance assessment by LVQ method, http://crdb.nifs.ac.jp/iti/list/list_top.php

[7] Y. Itikawa, ATOMIC DATA AND NUCLEAR DATA TABLES **64**, 151–315 (1996).

[8] Joint search across publishers, http://crdb.nifs.ac.jp/j_search/js_top.php