AN INTERNET ENABLED IMPACT LIMITER MATERIAL DATABASE

S. Wix, F. Kanipe, W. McMurtry

Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM

SUMMARY

This paper presents a detailed explanation of the construction of an internet enabled database, also known as a database driven web site. The data contained in the internet enabled database are impact limiter material and seal properties. The techniques used in constructing the internet enabled database presented in this paper are applicable when information that is changing in content needs to be disseminated to a wide audience.

INTRODUCTION

The rapid expansion of the Internet and Internet enabling technologies has allowed rapid communication of large amounts of information. With Internet browsers such as Netscape and standards such as the Hypertext Transport Protocol (HTTP) and the Hypertext Markup Language (HTML), graphical information as well as text information can be delivered anywhere in the world with a single click of a mouse. One of the more promising Internet enabling technologies is the database driven web site. The database driven web site allows on-the-fly building of web pages with dynamic data sources such as databases. This technology gives a constant look and feel to the web site with dynamic data content and is easy to update and maintain. These enabling technologies allowed the development of the Sandia National Laboratories Transportation Systems Impact Limiter Material Database (ILMD) and Seals Database (SD).

Both databases were developed using low-cost, readily available software designed for database accessibility from the Internet. The software was chosen based on rapid deployment, good Internet performance, low cost and ease-of- use. The computational platform is a Hewlett-Packard 9000/715 with the HPUX UNIX operating system.

SOFTWARE COMPONENTS

The software used for the ILMD and SD consists of mSQL as the database engine, WDB as the Internet/database connecting middleware, the PERL language, MSQLPERL as the middleware between PERL and mSQL, and the NCSA HTTP server. Each software component is discussed in detail below.

MSQL

MSQL is also known as mini SQL, which is light weight database engine developed by David J. Hughes at Bond University, Australia. MSQL has been designed to provide fast access to stored data with low memory requirements. As the name implies, mSQL offers a subset of the Structured Query Language (SQL) as its query interface. Although mSQL only supports a subset of SQL,

everything it supports is in accordance with the ANSI SQL specification. The mSQL package includes the database engine, a terminal "monitor" program, a database administration program, a schema viewer, and a C language Application Programming Interface (API). The API and the mSQL have been designed to work in a client/server environment over a TCP/IP network. The mSQL web site is **http://Hughes.com.au/**. On line information and links to tools used to connect mSQL to the web server are at **http://cscsun1.LaRC.NASA.Gov/~beowulf/msql/.**

The mSQL language offers a significant subset of the features provided by ANSI SQL. MSQL allows a program or user to store, manipulate and retrieve data in table structures. MSQL does not support relational capabilities such as views or nested queries. Although mSQL does not support all the relational operations defined in the ANSI specification, it does provide the capability of "joins" between multiple tables.

The mSQL daemon, msqld, is a stand-alone application that listens for connections on a TCP/IP socket. Msqld is a single process engine that will accept multiple connections and serialize the queries received. Msqld utilizes memory mapped input/output and cache techniques to offer rapid access to the data stored in a database. Testing of mSQL has shown that for simple queries, the performance of mSQL is comparable to or better than other freely available database packages. An example from the mSQL documentation shows that, on a set of sample queries including simple inserts, updates and selects, mSQL performed roughly 4 times faster than Ingres and over 20 times faster than Postgres on an Intel 486 class machine running Linux.

WDB

WDB consists of the WDB script which is written in PERL and a set of high-level form definition files (FDF). Each FDF describes a different view of the database. WDB automatically creates HTML forms on-the-fly to allow the users to query the database. WDB will query the database, given user query constraints, and present the results to the user in an HTML format. WDB also comes with a utility to extract information about a table from the database and create a working template form definition file. WDB was written by Bo Frese Rasmussen for the Space Telescope - European Coordinating Facility. Modifications to WDB by Jeff Rowe were used to construct the ILMD and SD.The WDB web site is **http://arch-http.hq.eso.org/wdb/html/wdb.html** and the modifications are available at **http://cscsun1.larc.nasa.gov/~beowulf/tutor/**.

A number of conversions and formats, such as hours to seconds and multiple date formats, are possible on the data coming from the database before they are shown to the users. a very useful feature is converting data from the database into a hypertext link. Using this feature can turn the entire database into a hypertext document. These hypertext links can be links to other elements in the database which provides a mechanism of relating information in the database and presenting the information to the user in HTML. The links could be any document on the Web, thus providing easily integration between data in the database and related documents on the Web.

WDB use an FDF as a template to describe a a table or tables in the database. The FDF contains information about the database name, table names, and a list of fields including their length, data type, and which table the field belongs to, in the case of a multi-table join.

The FDF files are custom-built and some of the customizing includes creating new PERL routines, reformatting the input and output, adding HTML tags for headers, footers, and inline images, and specify join conditions. Several different FDFs can access the same database, thus, displaying the information in formats depending on the needs of the users.

PERL

PERL is a full fledged programming language that is used quite often on the Internet as a Common Gateway Interface (CGI) program. PERL is an interpreted language which allows portability from platform to platform and shortened design time. PERL combines many of the features of shells, awk, sed, and C, without some of the drawbacks of other languages such as allowing a program to access memory due to a bug in the code resulting in a segmentation fault or a bus error. PERL adds some features of its own that make it a preferred language to use for CGI programs accessed from the Web. The PERL web site is at **http://www.perl.com/index.html.**

PERL is an interpreted language, which means there is no executable program that exists separate from the source code. The source code is interpreted at run time by the PERL parser. This means when a PERL script is run, the PERL program finds the PERL code, parses the code and checks for syntax errors, compiles the code into a bytecode format, and executed the code. Every time a PERL scripts runs, the previously described process occurs.

The interpreted nature of the PERL language has both benefits and drawbacks. Some of the benefits are:

• If the source code changes, the code can be executed immediately. The immediate code execution causes faster code development.

• PERL scripts are very portable. If the correct version of the PERL interpreter is on a computer, the PERL script will execute Instead of having to port programs to various operating systems, the PERL interpreter is ported thus removing platform dependencies from the PERL scripts.

• PERL code is fast to create and doesn't require the include files and the overhead of a C compiler, thus reducing overall cycle time development.

Some of the drawbacks are:

• Once the PERL script is parsed and compiled, it generally runs as fast as most C programs. However, there is the overhead of loading the PERL parser into computer memory, parsing, and compiling the script. With a traditional compiled executable, the long compile cycle is done once, then the program is loaded and executed quickly each time.

• Since the source script is needed to execute the program, anyone can read and edit the source script.

• In order to run the script on various machines, the PERL interpreter has to be ported and built on each machines.

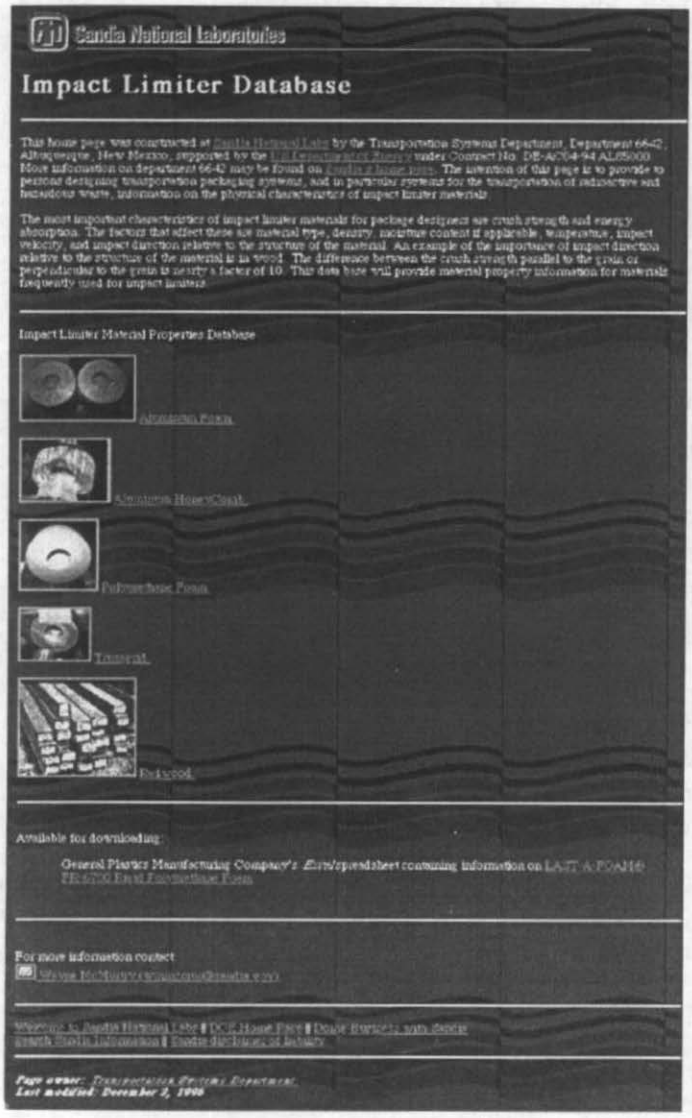


Figure 1 - Impact Limiter Materials Database Introductory Page

MSQLPERL

MsqlPerl is a PERL extension allowing PERL scripts to use the mSQL application programming interface (API). MsqlPerl takes the mSQL API functions and creates functions that can be called by PERL programs. The MsqlPerl extensions allow the mSQL API functions to be compiled into Perl. Once that is done, any PERL script with access to the newly compiled version of PERL can access a mSQL database, using the MsqlPerl function calls. The MsqlPerl module is available at **http://www.perl.com/CPAN-local/CPAN.html.**

NCSA HTTPD Server

The NCSA HTTP server was used as the web server software for the ILMD and SD. The software is in wide use on the World Wide Web. Features of the software are the compactness and speed of the code, server side includes and Common Gateway Interface (CGI) capabilities. The NCSA HTTPd server software is available at **http://hoohoo.ncsa.uiuc.edu/docs/Overview.html/.**

IMPACT LIMITER MATERIALS DATABASE

The ILMD contains structural material characteristics useful for impact limiter designers and analysts. The materials currently available are redwood, aluminum foam, aluminum honeycomb, trussgrid and polyurethane foam. A picture of each material as well as pertinent structural design data is available for each material. The database is on-line and is available for access at this URL:

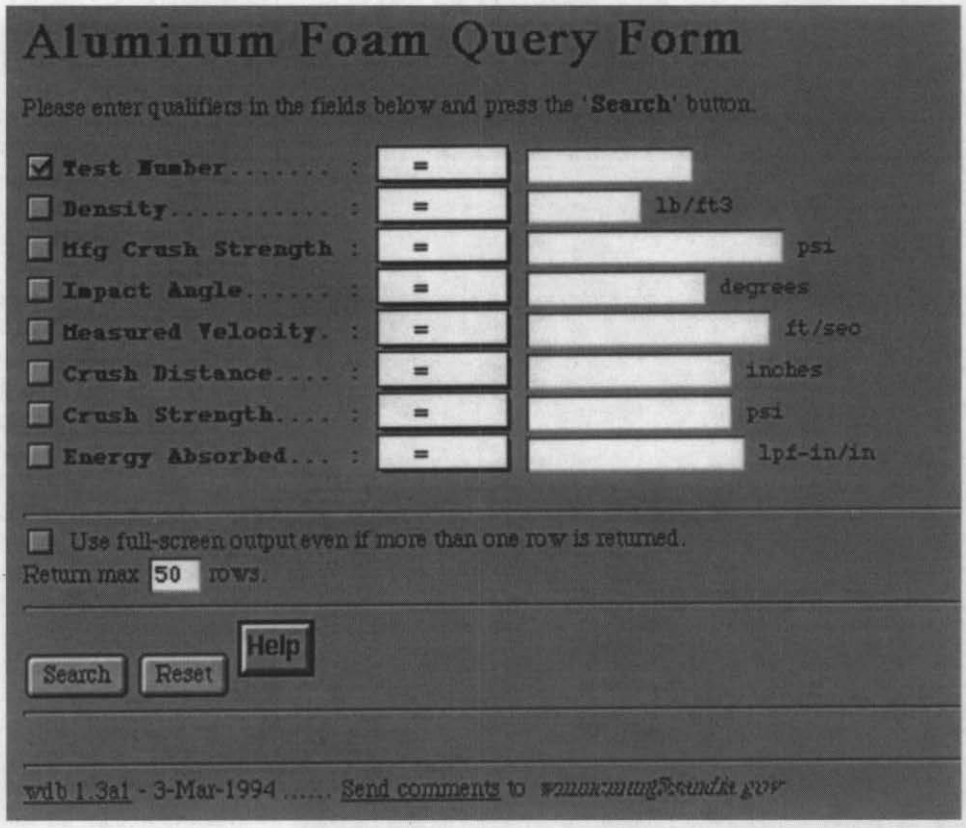
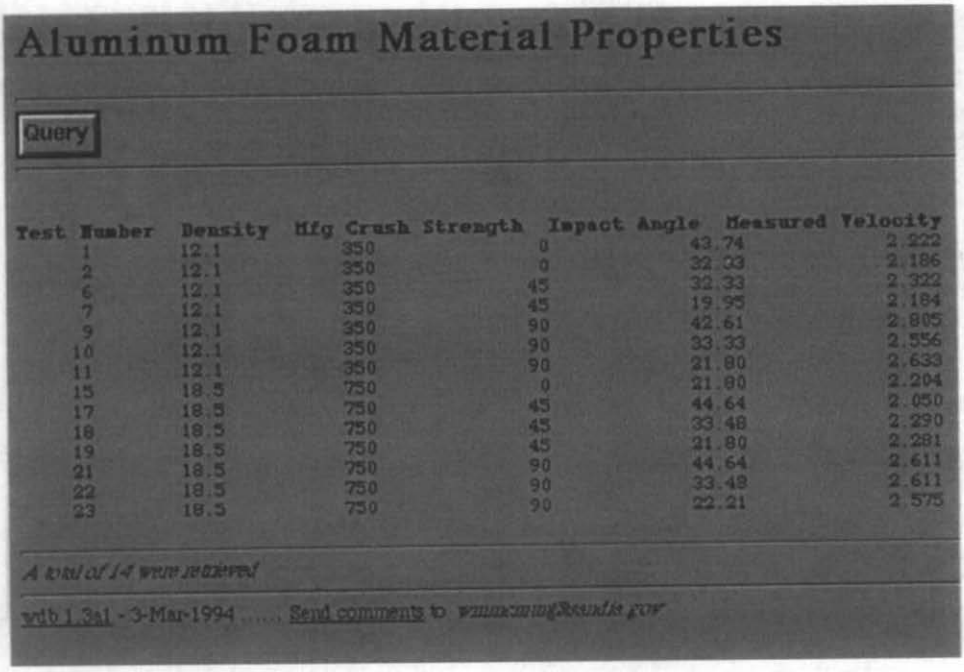


Figure 2 - Impact Limiter Materials Database Query Form for Aluminum Foam
http://ttd.sandia.gov/impact.limiter/impactlimiter.html

Figure 1 presents the introductory screen to the ILMD. The introductory screen presents a thumbnail picture of each material type, links to each material database, links to other material information and contacts for more information.

Figure 2 is the query page for the material database. The aluminum foam database was used for an example. The user first determines the properties of interest to query, such as test number, by checking the box next to each property of interest. Then the user can narrow the search by inputting values in the text box next to the property of interest. A pull down menu allows the user to select operators such as greater than, less than, and equal to. The user then presses the submit button and the query is submitted to the ILMD database.

Figure 3 presents the results page for the aluminum foam database. The query button takes the user back to the query page for a new search. Figure 4 presents an alternative method of viewing the data returned by a query. The method is invoked by pushing the full screen output button at the bottom of the query screen.



Aluminum Foam Material Properties

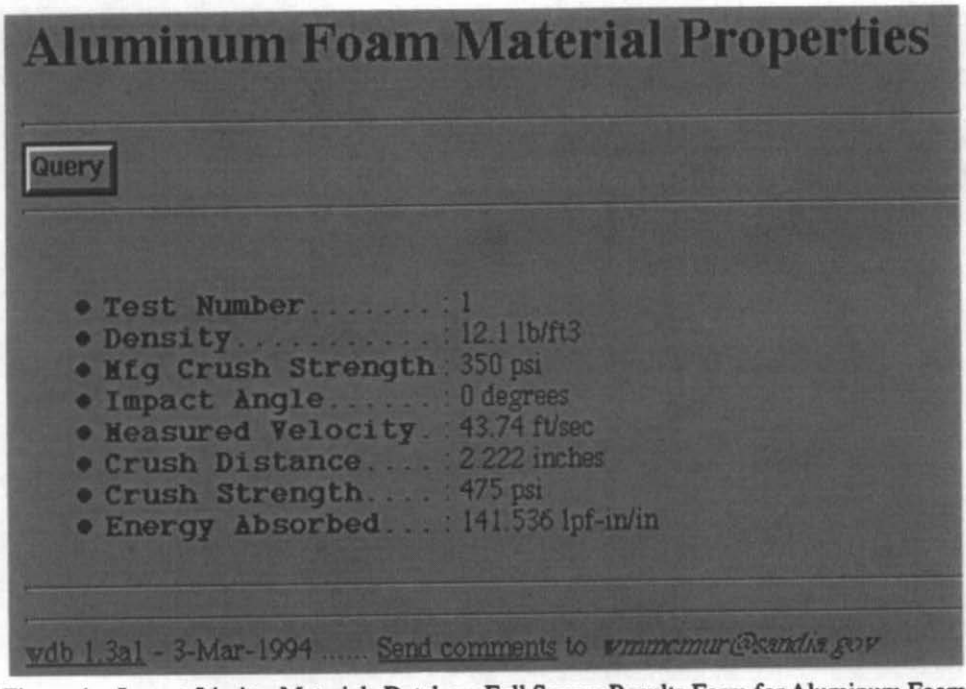
Query

Test Number	Density	Mfg Crush Strength	Impact Angle	Measured Velocity
1	12.1	350	0	43.74	2.222
2	12.1	350	0	32.03	2.186
6	12.1	350	45	32.33	2.322
7	12.1	350	45	19.95	2.184
9	12.1	350	90	42.61	2.805
10	12.1	350	90	33.33	2.556
11	12.1	350	90	21.80	2.633
15	18.5	750	0	21.80	2.204
17	18.5	750	45	44.64	2.050
18	18.5	750	45	33.48	2.290
19	18.5	750	45	21.80	2.281
21	18.5	750	90	44.64	2.611
22	18.5	750	90	33.48	2.611
23	18.5	750	90	22.21	2.575

A total of 14 were selected

vdb 1.3a1 - 3-Mar-1994 ..... Send comments to [illegible]

Figure 3 - Impact Limiter Materials Database Results Form for Aluminum Foam



Aluminum Foam Material Properties

Query

- Test Number . . . . . : 1
- Density . . . . . : 12.1 lb/ft3
- Mfg Crush Strength : 350 psi
- Impact Angle . . . . . : 0 degrees
- Measured Velocity . : 43.74 ft/sec
- Crush Distance . . . . : 2.222 inches
- Crush Strength . . . . : 475 psi
- Energy Absorbed . . . : 141.536 lpf-in/in

vdb 1.3a1 - 3-Mar-1994 ..... Send comments to [illegible]

Figure 4 - Impact Limiter Materials Database Full Screen Results Form for Aluminum Foam

SEALS DATABASE

The SD contains mechanical characteristic data and application information for a wide variety of seals. Some of the types of seals included in the database are elastomeric, metal welded and brazed. Pictures of the seals are also available. The database is on-line and is at this URL:

http://ttd.sandia.gov/seals/seals.html

The screens and method of performing searches on the seals database is similar to the ILMD. Future plans for the databases include adding other impact limiter materials and seals, and incorporation of advanced Internet technologies such as JAVA applets. Third party user data will also be included as the capabilities of the databases are enhanced.

REFERENCES

Mini SQL User's Guide, Release 1.0.11, Hughes Technologies Pty Ltd., Jan 1996

WDB - A WWW Interface to Databases Installation and Users Guide, Bo Frese Rasmussen March 10, 1996

WDB - A WWW Interface to Databases Installation and Users Guide with WDB Enhancements, Jeff Rowe, April, 1997

Perl On-Line Documentation, http://language.perl.com/info/documentation.html, Tom Christiansen, 1996

Comprehensive Perl Archive Network, http://language.perl.com/info/cpan_modules.html, 1997

NCSA HTTPd On-Line Documentation, http://hoohoo.ncsa.uiuc.edu/docs/Overview.html, NCSA HTTPd Development Team, 1996