

## ***David Muir Sharnoff***

resume addendum: Bepress details

<http://dave.sharnoff.org/resume>

### *Details of my work at Bepress*

**Persistent Object Store:** A transaction-safe object store for perl objects. Perl.

**Web Template Language:** I designed a web template language that was used to generate all of the Bepress web pages. In addition to the normal loops and conditionals, the template language provided access to the underlying perl objects that were being presented without giving the template writer any ability to bypass security. Editors of journals (ie: outsiders) were allowed to edit the templates for their own content. The template auto-generated cryptographic checksums for forms so that we could trust some of the information coming back with POSTs. The template automatically sized, and renamed images so that images would be cached whenever possible and load more quickly. Perl.

**Template editor:** Internal staff and editors could edit templates using the split-screen template editor. New templates could be saved so that they override very specific templates. For example: you could have special template just for the 1st article in the issues in the 2007 volume. Perl.

**Mailing list manager:** The academic community seems to have a high tolerance from spam from within their own community. We took advantage of this, but I wrote a mailing list manager that allowed people to reliably subscribe, unsubscribe and object to the email. There were few objections. Bounces were reliably detected and email responses were always matched up with the sent-to address. Perl. One part of this, Net::SMTP::Receive was released to CPAN.

**Site structure editor:** Not all journals had the same structure. Some had "tiers". Some had volumes and issues. Some did not. I built a site structure editor that allowed each journal to have its own layout. Perl.

**Academic Publishing Workflow:** Academic papers are sent out for review before they are accepted into a journal. I built the web application to manage this process, including various status pages and notifications. Perl.

**Site Logic:** The Bepress web site needed a lot of glue: user signup; user authentication; authorization for different site capabilities; control over downloads; security support libraries to prevent injections; confirmations; etc. I built them all. Perl.