Wanted: a Libmaster - no pay, but glory

[1/22] from: carl::s::rebol::com at: 14-Sep-2002 15:28

Wanted:

We need a Libmaster for the REBOL on-line library. For 5 years, I've done it, and not very well at that (I apologize - not enough time).

I'm looking for someone who's diligent (and loves working on REBOL) to take it over. We can bounce ideas around, but the Libmaster will have full control over the library, how it works, change policies, postings, etc. The server will continue to be at reboltech.com, which has good backbone connections, so it's a good spot.

I tell you I'm worried... because there are a lot of new people (e.g. a lot of young kids) who are interested in REBOL, but they find that a lot of the scripts in the lib don't work (because they are examples, not working scripts -- so they should be in an example section, apart from other scripts.)

Also, I know that many of us, most of us, have dozens of scripts that are useful to others, and we need a reliable place to put them.

So, please contact me if you have an interest! REBOL experience, of course, is required, and there's no pay for the job, but a lot of glory... major historic REBOL glory. ;)

Thanks,

-Carl
01: library stats

- published scripts .................... 795
- messages .............................. 45,644
- articles and docs .................... 75
- active team members ................ 5
  - Maxim Olivier-Adlhoc: repack and future projects
  - Gregg Irwin: leader
  - Sunanda: code droid
  - Brian Tiffin: rookie script ranger
  - Peter Wood: quality assurance

- members ............................... 218
  This number has remained remarkably consistent for around two years. Memberships (especially spam sign-ups) are automatically deleted at regular intervals. The interval varies depending on how spammy / inactive the member appears:
  - if you own a script you are good for life
  - spammy and / or incomplete sign-ups are usually purged within a few weeks
  - memberships are usually dropped 11 months after the last activity; they can usually be resuscitated up to a year after that
  - total expired members since 1-jan-2005 ........................................... 498
  - total incomplete sign-ups since 1-jan-2005 ........................................ 91

- script owners .......................... 98
When we were hit by a troll, we quickly added some troll protection:

- first quick idea was to add some delay. Make them bored and they'd go away. Sadly, even adding 45 seconds to their response times didn't stop them.
- next, we made them a "read only" Library member: if they tried to add scripts (articles, documents, discussions, etc), they got told rather blandly: "Sorry, that feature is not available just now"
- data structure in user record to add troll protection fields:
  - tpf: [0 0 0 0 0 0] ;; only tpf/1 and tpf/2 in use as yet
- code in common header::
  - if member/tpf/1 <> 0 [wait random 90]
- code in half a dozen places where updates happen:
  - if member/tpf/2 <> 0 [do %say-goodbye-troll.r quit]

Several spammers:

- Form spam: pornography and/or links in our feedback form. pol.r is poised to take care of that.
- Sign-up spammers: most are too stupid to complete the process.
  - Those that do leave an obvious signature, and their memberships are auto-expired in a couple of weeks.
  - Many of those that do survive are too stupid to enable the display of their homepage URL, so no one ever sees it.
  - Those that get that right, are too stupid to realise that we do no display members' homepage URLs except to logged on members, so their link is visible only to 200 people at most.
- A sheer waste of their effort!

cgi scripts
totalling 1,800K of commented source code

indexes
including indexes for scripts, mailing list archive and articles.
http://www.rebol.org/cgi-bin/cgiwrap/rebol/site-search.r

html templates

known bugs
Garbage collection crashes every 150,000 or so CGI executions
Library members page shows date of last update as date of last viewing
02: development environment

- live site ............... BSD/apache/rebol/core
- dev site .................... Win/xitami /core
- dev site .................... OSX/apache/core

There are remarkably few cross-platform issues between Live and Dev. Most are documented here:
http://www.rebol.org/cgi-bin/cgiwrap/rebol/art-display-art?article=x60w

There are many differences between releases of REBOL/Core. So many, that we cannot safely move on from Core 2.5.6 until there is a major retesting of the whole code base.

But that also causes problems:
- We cannot load a script with a needs header field greater than 2.5.6 (well, we can but we cheat to do it)
- Similarly, we have to cheat to load a script with some "modern" syntax, such as
  
  field: n: 1

server-specific code. ........... 4 line .htaccess
- ErrorDocument 404 ....
- DirectoryIndex ....
- ReDirect ....
- IndexIgnore ....

databases ........................ none

non-REBOL code ........................ cgiwrap

javascript ........................ empty <script> tag
source library ............................. IOS

Library/IOS

RT

REBOL.org

Local

Uploader

- Add shebang!
- Check checksums
- Error recovery

Live

Dev

Gregg

Peter

Sunanda
03: access format agnostic

- web site
- rss feed
- email notices
- Reb site (updated weekly)
- desktop application

http://www.rebol.org/cgi-bin/cgiwrap/rebol/download-librarian.r
### API (LDS)

**Library Data Services** - complete API to the data held on the server.

```rebol
do http://www.rebol.org/library/public/lds-local.r
probe lds/send-server "help" []
probe lds/send-server "search-rml" ["rugby"]
probe lds/send-server "search-rml" ["rugby 2006"]
probe lds/send-server "list-scripts-owned-by" ["anton"]
probe lds/send-server "get-script-stats" ["anamonitor.r"]
probe lds-send-server "contribute-script" [
  use-mode "check"
  note "just a demo, not really updating"
  script {rebol [file: %test.r] print "hi" quit}
]
```

(this is a somewhat idealised schema: many cgi scripts had their own embedded/entangled code for data access before the parallel LDS API command was created. So some logic is repeated).
04: **idealised cgi script**

```rebol
#!shebang – inserted by uploader according to the target environment
REBOL [... standard header ...] 
do %preferences.r ;; settings for this 

;; env (eg paths to files; url prefixes)
;; debug mode

do cgi-debug | ;; see note 1
;; rest of script is contained within the execution block

do %librarian-utilities.r ;; lots of useful things:
;; includes automatic RID check

do %xxx-utilities.r ;; special to a section of the Library, eg

;; mailing-list-utilities.r

do %safe-cgi-input.r ;; note 2

cgi-data: safe-cgi-data-read

page-template: read preferences/templates/search.html

;; validate cgi data here

;; access data here

;; replace used place holders in html here

library-utilities/print-page page-template

] ;; end of cgi-debug – includes the ‘quit
```

**note 1:** Basically, this script with some extra goodies:
http://www.rebol.org/cgi-bin/cgiwrap/rebol/view-script.r?script=cgi-debug.r

**note 2:** in the ideal world, this script:
http://www.rebol.org/cgi-bin/cgiwrap/rebol/view-script.r?script=safe-cgi-data-read.r

**note 3:** see this Community discussion:
http://www.rebol.org/cgi-bin/cgiwrap/rebol/ml-display-thread.r?m=rmlGPGQ

In practice few scripts are this tidy! This is how we’d do it if we were starting over.
05: highly visible on the web

- third in world in Google for [script library]
- high rankings for many REBOL-related search terms
- a script in the Library is highly visible
06: members pages

- personal page for all members
- use whatever text-to-html formatter you prefer

Gregg is using MakeDoc

Philippe goes for Make Doc 2

Hallvard keeps it simple

Brian creates his own styles

http://www.rebol.org/cgi-bin/cgiwrap/rebol/lmp-index.r
07: **cgi mode**

- library does not use fast-cgi
- each cgi request fires up a new copy of rebol/core:
  - fewer run-time garbage collection errors
  - each cgi could have a tailored executable
  - no dependence on a specific webserver
  - no twiddly installation settings 😊

REBOL fast cgi overview:
08: **script owners stats**

### Detailed stats on your scripts

<table>
<thead>
<tr>
<th>Scripts owned stats: sunanda</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>script</strong></td>
</tr>
<tr>
<td>-------------</td>
</tr>
<tr>
<td>google-r</td>
</tr>
<tr>
<td>alexandra-r</td>
</tr>
<tr>
<td>google-r</td>
</tr>
<tr>
<td>base-r</td>
</tr>
<tr>
<td>rsi-panel</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>extend-object</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>make-mail-list</td>
</tr>
<tr>
<td>project-r</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>project-r</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
<tr>
<td>srs-list</td>
</tr>
</tbody>
</table>

- **Click the number in the stats column to drill down to access stats and version details**
09: **script owners resources**

- compare previous versions of a script

  ![Compare versions of skimp.r](image)

  **Settings**
  - Compare:
    - 1 -- 5-Apr/8:40
    - 3 -- 3-May/15:46
  - Options:
    - Ignore blank lines
    - Ignore case
    - Ignore all whitespace
    - Ignore leading whitespace
    - Shows moves as ins/del
    - Display changes only
  - Help for comparing versions

  **Results**
  - Summary: [equ: 0 ins: 41 del: 12 mov: 109]

<table>
<thead>
<tr>
<th>disp</th>
<th>v1 line</th>
<th>v2 line</th>
<th>source</th>
</tr>
</thead>
<tbody>
<tr>
<td>del</td>
<td>4</td>
<td></td>
<td>date: 7-apr-2009</td>
</tr>
<tr>
<td>ins</td>
<td>4</td>
<td></td>
<td>date: 13-apr-2009</td>
</tr>
<tr>
<td>del</td>
<td>6</td>
<td></td>
<td>version: 0.0.1</td>
</tr>
<tr>
<td>ins</td>
<td>6</td>
<td></td>
<td>version: 0.0.2</td>
</tr>
<tr>
<td>ins</td>
<td>12</td>
<td></td>
<td>(0.0.1 21-apr-2009 “add disk-cache and disk-cache-all“)</td>
</tr>
<tr>
<td>ins</td>
<td>222</td>
<td></td>
<td>disk-cache: memo []</td>
</tr>
<tr>
<td>ins</td>
<td>224</td>
<td></td>
<td>/disk</td>
</tr>
<tr>
<td>mov</td>
<td>226</td>
<td>227</td>
<td>1</td>
</tr>
<tr>
<td>ins</td>
<td>228</td>
<td></td>
<td>/ there we've got it now</td>
</tr>
<tr>
<td>mov</td>
<td>699</td>
<td>229</td>
<td>1</td>
</tr>
<tr>
<td>mov</td>
<td>700</td>
<td>230</td>
<td>getindex: find index-cache index-name</td>
</tr>
<tr>
<td>ins</td>
<td>231</td>
<td></td>
<td>if not found then failure: bad index-name</td>
</tr>
</tbody>
</table>

- mark versions as public or not
- get stats via LDS interface
- notifications if discussion started on owned script
- get stats via LDS API
010: development methodology

- fully worked example of documentation and decision making process (from Library Team private group on Altme REBOL3 world):

**Gregg:** Yeah, I always forget about that myself. Maybe we need a site map page. Could we do a single, shared page that worked like a member page (i.e. just a make-doc page you edit)? Sort of a pseudo-wiki.

**Sunanda:** Gregg: <=<Could we do a single, shared page that worked like a member page (i.e. just a make-doc page you edit)? Sort of a pseudo-wiki.>

Easy.

Simplest way would be to allow Articles tagged as (say) [wiki/public] to be edited by any Library member, rather than just the owner(s).

**Gregg:** Worth doing an experiment maybe, if it isn’t hard.

**Sunanda:** Will do --- once we have the IOS server back :-)

**Gregg:** I’m sure that will be after DevCon at this point.

**Sunanda:** Thanks ...... That’s rather what I suspected.

*If* I get time in the next week, it would be fun to add a sort of instant-wiki to the Library and unveil it at DevCon. I’ll keep checking my watch for an extra hour or two in the day :-)

**Sunanda:** Gregg <=<Worth doing an experiment maybe, if it isn’t hard.>>>

I just done it -- it took all of 20 minutes.

Not yet live. I’ll leave it a few days in case I think of any annoying subtleties .... Unless anyone is desperate to try it out now.

Basic spec is:

- article owner can add a [wiki/public] tag
- any logged on user can edit/update any article with that tag
- only the owner can remove that tag (so others can’t stop the article being publicly edited)
- only the owner can unpublish the article (so others can’t make it vanish)

***

- no checks if the article has changed since the member started editing, so parallel typing could lose a version.
- We don’t have a history/archive on articles so vandalism reversion is not so easy (we can retrieve older versions from the activity logs, but that’s a bit fiddly). So if the feature takes off, we’d need to add some version control.

**Gregg:** Great!
011: new feature

- "miki" wiki -- released: today: articles editable by a library member
012: data storage

- all flat files
- creates occasional access blip
- very fast as no database start up overhead
- crafted solutions for different storage problems
- highly portable – works on all dev and live platforms

Data design case study: mailing list physical storage

need to support fast random and serial access to between 35,000 (design date) and 200,000 (assumed design life) mailing list posts

posts have three distinct parts:
* the message text (“hello, I have a problem with REBOL”)
* post header info (date sent, sender, subject line)
* meta data (tags, thread information)

solution for message text information:
* store 100 messages per physical record as a block
* physical records saved as write/binary compress
* store 100 physical records per folder
* create folders as needed when physical record is created

so message number 12,345,678:
* folder: msg-data/12/34
* file msg-data-56
* record: pick entry 78

similar solution for index records

attributes:
* supports potentially 100 million messages
* messages in a thread are usually in the same or adjacent physical record
* small space (25meg for all Mailing list related data and indexes)
* no database start-up overheads
* very fast in practice
013: team goodies

Debug team: learn the worst

Plugin posse: front-line anti-spammers

Script ranger: can update any script

ML tag team: tag any post

ML commentator: keep info fresh
014: mistakes

- no surrogate id for scripts
- no common header for cgi scripts
- dependence on one version of rebol
- script meta data in rebol header
- entangled code and data
015: not even mentioned

- personalised css
- anti-bot protection (RID)
- POL (captcha monkey protection)
- packages and repack
- indexing (skimp)
- structure vs caching
- favorites / popularity
- background tasking (using bots as clock)
- spin-ins
- spin-offs
- notices / RSS / front page new
016: **help and future**

- **What's next?**