

gitweb-aggregator

Claude Heiland-Allen

2015–2016

Contents

1	gitweb-aggregator.conf.sample	2
2	gitweb-aggregator.sh	2
3	gitweb-aggregator.xsl	4
4	README.md	5

1 gitweb-aggregator.conf.sample

```
# example configuration file
#
# protocol for web addresses (either http or https)
proto="https"
5 # prune entries before this time
  since="$(date --date '1 week ago' --utc '+%FT%TZ')"
```

```
# site hostname
site="code.mathr.co.uk"
# prefix to strip from feed title when forming category term
10 prefix="${site} - "
# suffix to strip from feed title when forming category term
suffix="/atom log"
# gitweb text index query string
index="?a=project_index"
15 # output file for aggregated feed
out="${HOME}/var/www/${site}/www/commits.atom"
# directory for cache
cache="${HOME}/var/cache/gitweb-aggregator/${site}"
# content for aggregated feed header
20 header="$(cat << EOF
<header xmlns="http://www.w3.org/2005/Atom">
<title>${site} commits</title>
<link rel="alternate" type="text/html" href="${proto}://${site}/" />
<link rel="self" type="application/atom+xml" href="${proto}://${site}/commits.
  ↵ atom" />
25 <id>${proto}://${site}/commits.atom</id>
<icon>${proto}://${site}/static/git-favicon.png</icon>
<logo>${proto}://${site}/static/git-logo.png</logo>
</header>
EOF
30 )"
```

2 gitweb-aggregator.sh

```
#!/bin/sh
self="$(dirname "$(readlink -e "${0}")")"
```

```

conf="$(readlink -e "${1}")"

5  # read configuration
  if [ "x" = "x${1}" ]
  then
    echo >&2 "usage: ${0} config-file"
    exit 1
10  fi
  . "${conf}" || exit 1

# check configuration
[ "x" = "x${proto}" ] && echo >&2 "error: 'proto' not set by '${conf}'" && ↵
  ↵ exit 1
15 [ "x" = "x${cache}" ] && echo >&2 "error: 'cache' not set by '${conf}'" && ↵
  ↵ exit 1
[ "x" = "x${since}" ] && echo >&2 "error: 'since' not set by '${conf}'" && ↵
  ↵ exit 1
[ "x" = "x${site}" ] && echo >&2 "error: 'site' not set by '${conf}'" && ↵
  ↵ exit 1
[ "x" = "x${index}" ] && echo >&2 "error: 'index' not set by '${conf}'" && ↵
  ↵ exit 1
# [ "x" = "x${prefix}" ] && echo >&2 "error: 'prefix' not set by '${conf}'" && ↵
  ↵ exit 1
20 # [ "x" = "x${suffix}" ] && echo >&2 "error: 'suffix' not set by '${conf}'" && ↵
  ↵ exit 1
[ "x" = "x${header}" ] && echo >&2 "error: 'header' not set by '${conf}'" && ↵
  ↵ exit 1
[ "x" = "x${out}" ] && echo >&2 "error: 'out' not set by '${conf}'" && ↵
  ↵ exit 1

# download feeds
25 wget -nv -P "${cache}" -x -nH "${proto}://${site}/${index}"
  cat "${cache}/index.html${index}" |
  while read repository author
  do
    echo "'${proto}://${site}/${repository}/atom'"
30 done |
  xargs wget -nv -N -P "${cache}" -x -nH

# aggregate feeds
(
35   echo '<?xml version="1.0" encoding="utf-8"?>'
   echo '<feeds>'
   echo "${header}"
   cat "${cache}/index.html${index}" |
   while read repository author
40   do
     # strip xml header line
     tail -n+2 < "${cache}/${repository}/atom"
     done
     echo '</feeds>'
45 ) |
  xsltproc \
    --stringparam SINCE "${since}" \
    --stringparam PREFIX "${prefix}" \
    --stringparam SUFFIX "${suffix}" \
50   "${self}/gitweb-aggregator.xsl" -> "${cache}/commits.atom"

```

```
# update target only if changed
diff >/dev/null 2>/dev/null -q "${cache}/commits.atom" "${out}" ||
cp -af "${cache}/commits.atom" "${out}"
```

3 gitweb-aggregator.xsl

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
5  xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns="http://www.w3.org/2005/Atom"
>
<xsl:output method="xml" indent="yes" encoding="UTF-8" />
<xsl:param name="SINCE" />
10 <xsl:param name="PREFIX" />
<xsl:param name="SUFFIX" />
<xsl:template match="/">
<feed>
<xsl:copy-of select="feeds/atom:header/*" />
15 <generator uri="https://code.mathr.co.uk/gitweb-aggregator">gitweb-aggregator</
  ↳ generator>
<!-- updated -->
<xsl:for-each select="feeds/atom:feed">
  <!-- sort by date descending -->
  <xsl:sort select="substring(atom:updated, 1, 4)" order="descending" />
20 <xsl:sort select="substring(atom:updated, 6, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 9, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 12, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 15, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 18, 2)" order="descending" />
25 <!-- take the first item -->
  <xsl:if test="position() = 1">
    <xsl:copy-of select="atom:updated" />
  </xsl:if>
</xsl:for-each>
30 <!-- entry -->
<xsl:for-each select="feeds/atom:feed/atom:entry">
  <!-- sort by date descending -->
  <xsl:sort select="substring(atom:updated, 1, 4)" order="descending" />
  <xsl:sort select="substring(atom:updated, 6, 2)" order="descending" />
35 <xsl:sort select="substring(atom:updated, 9, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 12, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 15, 2)" order="descending" />
  <xsl:sort select="substring(atom:updated, 18, 2)" order="descending" />
  <!-- filter by date >= SINCE -->
40 <xsl:if test="
    substring(atom:updated, 1, 4) > substring($SINCE, 1, 4) or (
    substring(atom:updated, 1, 4) = substring($SINCE, 1, 4) and (
    substring(atom:updated, 6, 2) > substring($SINCE, 6, 2) or (
    substring(atom:updated, 6, 2) = substring($SINCE, 6, 2) and (
45 substring(atom:updated, 9, 2) > substring($SINCE, 9, 2) or (
    substring(atom:updated, 9, 2) = substring($SINCE, 9, 2) and (
    substring(atom:updated, 12, 2) > substring($SINCE, 12, 2) or (
    substring(atom:updated, 12, 2) = substring($SINCE, 12, 2) and (
    substring(atom:updated, 15, 2) > substring($SINCE, 15, 2) or (
```

```

50     substring(atom:updated, 15, 2) = substring($SINCE, 15, 2) and (
        substring(atom:updated, 18, 2) >= substring($SINCE, 18, 2) )))))))))))
">
    <entry>
        <!-- include feed identifier in title -->
55     <title type="html">[<xsl:value-of select="substring-before(substring-
        ↵ after(../atom:title, $PREFIX), $SUFFIX)" />] <xsl:value-of select=
        ↵ ="atom:title" /></title>
        <!-- include feed identifier as category -->
        <category term="{substring-before(substring-after(../atom:title, $PREFIX
        ↵ ), $SUFFIX)}" />
        <xsl:copy-of select="atom:updated" />
        <xsl:copy-of select="atom:published" />
60     <!-- ignore author email -->
        <author><xsl:copy-of select="atom:author/atom:name" /></author>
        <xsl:copy-of select="atom:link" />
        <xsl:copy-of select="atom:id" />
        <xsl:copy-of select="atom:content" />
65     <source>
        <xsl:copy-of select="../atom:id" />
        <xsl:copy-of select="../atom:link" />
        <xsl:copy-of select="../atom:title" />
        <xsl:copy-of select="../atom:updated" />
70     </source>
    </entry>
</xsl:if>
</xsl:for-each>
</feed>
75 </xsl:template>
</xsl:stylesheet>

```

4 README.md

gitweb-aggregator

Aggregate atom feeds from gitweb. Useful if you have a gitweb instance,
5 and want to generate a feed for everything.

Usage

10 Read the source. Some things assume the URL layout presented by
code.mathr.co.uk, so you will probably need to customize things.

15 --
claude@mathr.co.uk
http://mathr.co.uk