



# THE EVERGREEN HTTP INTERFACE

---

*Generating New Item Lists Elegantly and Automatically*



# OTHER OPTIONS

---

- ❖ LibraryThing Widget
- ❖ Goodreads Widget
- ❖ LibraryThing for Libraries Widget (Subscription)
- ❖ Manual entry and retrieval



# POTENTIAL ISSUES

---

- ❖ Add additional steps for technical services and other staff.
- ❖ Limited control over display and layout.
- ❖ Inconsistent behavior of some third-party widgets.



**BUT THERE'S A  
BETTER WAY!**



# EVERGREEN HTTP CALLS

---

- ❖ Up-to-date information!
- ❖ Automatically generated!
- ❖ You have control over how the data is handled!
- ❖ Compatible with any system!



# ISSUES WITH EVERGREEN

---


- ❖ Higher learning curve!
- ❖ You are responsible for integration!
- ❖ Higher learning curve!
- ❖ Calls can be slow, especially when they're big!
- ❖ **Higher Learning Curve!**



*Retrieving data in a standardized XML format*

---

THE ATOMIC CALL



[http://gapines.org/opac/extras  
/browse/atom-full/item-age  
/<PINES Name>/1  
/<Number of Items To Retrieve>](http://gapines.org/opac/extras/browse/atom-full/item-age/<PINES Name>/1/<Number of Items To Retrieve>)



# WHAT IT DOES

- ❖ <http://gapines.org/opac/extras/browse>
  - The PINES Evergreen server OPAC, extras, browsing service
- ❖ [/atom-full/](#)
  - Return the results as an Atom-standard XML document
- ❖ [/item-age/](#)
  - Sort the results by age.

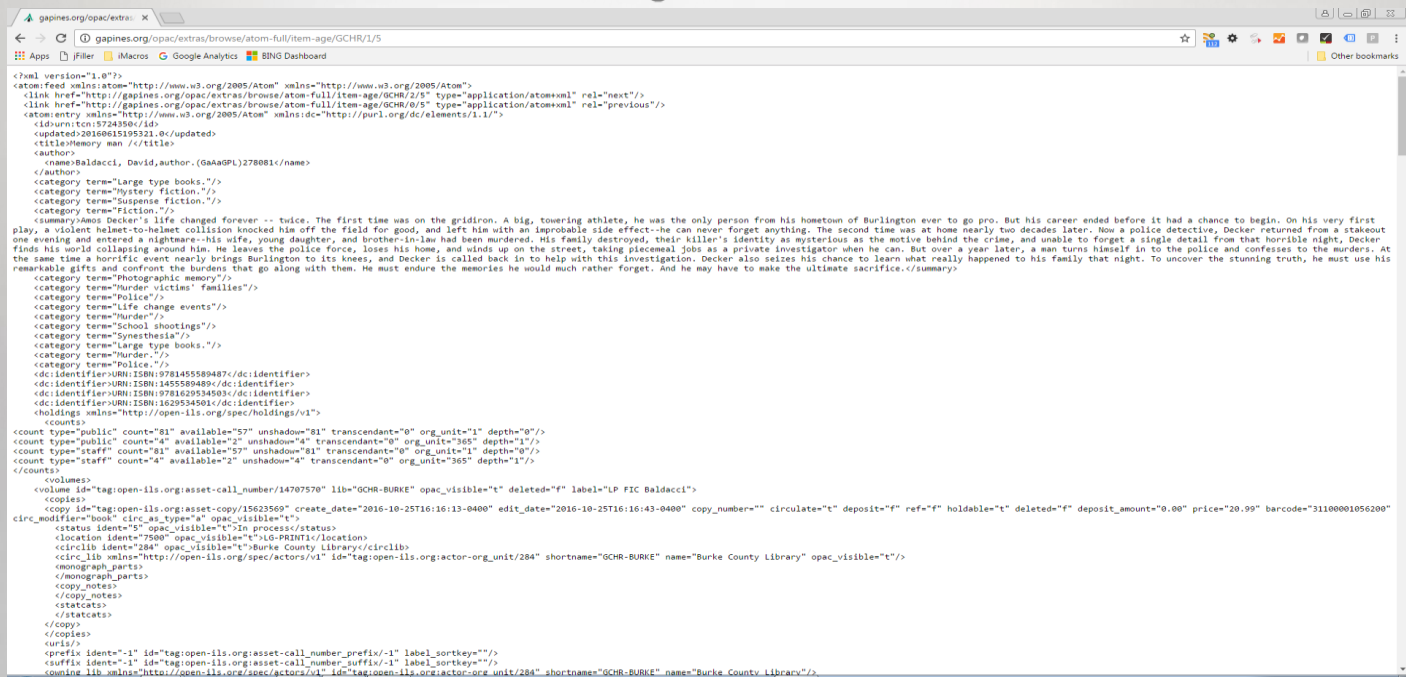
# WHAT IT DOES

- ❖ /<PINES NAME>/
  - Variable, specifies the location to retrieve items for. Examples – GCHR, ARL-ATH, etc.
- ❖ /1/
  - First page of results
- ❖ /<Number of Items to Retrieve>/
  - Variable...the number of items to retrieve per page.



[http://gapines.org/opac/extras  
/browse/atom-full/item-age  
/GCHR/1/5](http://gapines.org/opac/extras/browse/atom-full/item-age/GCHR/1/5)

# WHICH GIVES US...



```
<?xml version="1.0"?>
<atom:feed xmlns:atom="http://www.w3.org/2005/Atom" xmlns="http://www.w3.org/2005/Atom">
  <link href="http://gapsines.org/opac/extras/browse/atom-full/item-age/GCHR/2/5" type="application/atom+xml" rel="next"/>
  <link href="http://gapsines.org/opac/extras/browse/atom-full/item-age/GCHR/0/5" type="application/atom+xml" rel="previous"/>
  <atom:entry xmlns="http://www.w3.org/2005/Atom" xmlns:dc="http://purl.org/dc/elements/1.1/">
    <id>urn:tcn:5724350:/id
    <updated>20160815195321.0/updated
    <title>Memory man </title>
    <author>
      <name>Baldacci, David, author. (QaaA9L)278081</name>
    </author>
    <category term="Large type books."/>
    <category term="Mystery fiction."/>
    <category term="Suspense fiction."/>
    <category term="Fiction."/>
    <summary>Amos Decker's life changed forever -- twice. The first time was on the gridiron. A big, towering athlete, he was the only person from his hometown of Burlington ever to go pro. But his career ended before it had a chance to begin. On his very first play, a violent helmet-to-helmet collision knocked him off the field for good, and left him with an improbable side effect--he can never forget anything. The second time was at home nearly two decades later. Now a police detective, Decker returned from a stakeout one evening and entered a nightmare--his wife, young daughter, and brother-in-law had been murdered. His family destroyed, their killer's identity as mysterious as the motive behind the crime, and unable to forget a single detail from that horrible night, Decker finds his world collapsing around him. He leaves the police force, loses his home, and winds up on the street, taking piecemeal jobs as a private investigator when he can. But over a year later, a man turns himself in to the police and confesses to the murders. At the same time a horrific event nearly brings Burlington to its knees, and Decker is called back in to help with this investigation. Decker also seizes his chance to learn what really happened to his family that night. To uncover the stunning truth, he must use his remarkable gifts and confront the burdens that go along with them. He must endure the memories he would much rather forget. And he may have to make the ultimate sacrifice.</summary>
    <category term="Photographic memory"/>
    <category term="Murder victims' families"/>
    <category term="Police"/>
    <category term="Life change events"/>
    <category term="Murder"/>
    <category term="School shootings"/>
    <category term="Synesthesia"/>
    <category term="Large type books."/>
    <category term="Murder."/>
    <category term="Police."/>
    <dc:identifier>URN:ISBN:9781455580487</dc:identifier>
    <dc:identifier>URN:ISBN:1455580489</dc:identifier>
    <dc:identifier>URN:ISBN:9781629534503</dc:identifier>
    <dc:identifier>URN:ISBN:1629534503</dc:identifier>
    <holdings xmlns="http://open-ils.org/spec/holdings/v1">
      <counts>
        <count type="public" count="81" available="57" unshadow="81" transcendent="0" org_unit="1" depth="0"/>
        <count type="public" count="4" available="2" unshadow="4" transcendent="0" org_unit="365" depth="1"/>
        <count type="staff" count="81" available="57" unshadow="81" transcendent="0" org_unit="1" depth="0"/>
        <count type="staff" count="4" available="2" unshadow="4" transcendent="0" org_unit="365" depth="1"/>
      </counts>
    </volumes>
    <copy id="tag:open-ils.org:asset-call_number/14707570" lib="GCHR-BURKE" opac_visible="t" deleted="f" label="LP FIC Baldacci">
      <copies>
        <copy id="tag:open-ils.org:asset-copy/15623560" create_date="2016-10-25T16:16:13-0400" edit_date="2016-10-25T16:16:13-0400" copy_number="" circulate="t" deposit="f" ref="f" holdable="t" deleted="f" deposit_amount="0.00" price="20.99" barcode="31100001056200"
          <status id="5" opac_visible="t">In process</status>
          <location id="7500" opac_visible="t">10-PRH11/location
          <circlib id="284" opac_visible="t">Burke County Library</circlib
          <circ_lib xmlns="http://open-ils.org/spec/actors/v1" id="tag:open-ils.org:actor-org_unit/284" shortname="GCHR-BURKE" name="Burke County Library" opac_visible="t"/>
          <monograph_parts>
            <copy_notes>
              </copy_notes>
            <statcats>
              </statcats>
            </copy>
            </copies>
          </uri>
        <prefix id="1" id="tag:open-ils.org:asset-call_number_prefix/-1" label_sortkey=""/>
        <suffix id="1" id="tag:open-ils.org:asset-call_number_suffix/-1" label_sortkey=""/>
        <online lib xmlns="http://open-ils.org/spec/actors/v1" id="tag:open-ils.org:actor-org_unit/284" shortname="GCHR-BURKE" name="Burke County Library"/>
      </copy>
    </volumes>
  </atom:entry>
</atom:feed>
```



*Using server scripts to only retrieve what you want to use!*

---

THAT'S A LOT OF  
INFORMATION!

- ❖ Not supported on all hosting services.
- ❖ Low learning curve in general, higher curve for exporting data.
- ❖ Might not play well with your CMS.

## **PYTHON**

Using feedparser to convert the XML output of Evergreen into a JSON file which can be manipulated easily with Javascript.

- ❖ Near universal support
- ❖ Higher learning curve than Python, but incorporates web requests more intuitively
- ❖ More likely to play well with your CMS.

## **PHP**

Using `simplexml` and `json_encode` to create an easily manipulated JSON file.



# OTHER OPTIONS

---

## ❖ Ruby on Rails

- self-contained MVC framework
- similar compatibility issues to Python

## ❖ Perl

- Ubiquitous
- XML requires some extra libraries and steps to deal with

## ❖ C#

- Requires an ASP.net server



# TWO METHODS

## ❖ MVC Based

- Translate XML into relevant content
- Save content in a database model
- Retrieve information from the database at will
- Example at <https://gchrl.org>

## ❖ Live Feed

- Retrieve XML content
- Extract relevant information
- Translate into an easily used format (JSON, as an example)
- Example at [http://gchrlphp.dreamhosters.com/json\\_test.php](http://gchrlphp.dreamhosters.com/json_test.php)

# HOW TO GET THE DATA - PHP

- ❖ Using `$xml = simplexml_load_file($url)`
- ❖ Important! Use `xpath()` to navigate the atom structure!
  - `foreach ($xml->xpath("//atom:entry") as $entry);`
  - `$entry->registerXPathNamespace('atom', 'http://www.w3.org/2005/atom');`

# HOW TO GET THE DATA - PYTHON

---

## ❖ Using feedparser

- First, install feedparser using PIP - `pip install feedparser`

❖ `item_json = feedparser.parse('<insert url here>')`