Academic SIG
A COSUGI Special Interest Group

Web Services for "My Account" functions

By Mark Witteman
University of Virginia Library

Virgo: the UVA Library discovery platform

- "VIRGO" originally was the branded name for Sirsi's "Webcat" OPAC
- The name was retained when we built our own discovery system
  - Approximately 2005
  - Built on Blacklight
  - Many search sources
- Sirsi source came from catalogdump: MARC, callnum, item data
- Nightly update
- Used direct Oracle query for real-time availability (status of items)

A new version of Virgo

- "Version 4" is a total re-write
  - 2019
  - Ruby
  - Locally written API based back-end
  - Many search sources
  - For Symphony catalog, use catalogdump to provide MARC, callnum, item data

Symphony SaaS migration

- In 2018, before Virgo 4 work
- When local hosted, Virgo made direct connection to Symphony Oracle tables for real-time availability
- Not allowed with the Oracle license offered to SaaS customers
- So Symphony Web Services instead!
  - Needed to make the SWS methods work before SaaS migration

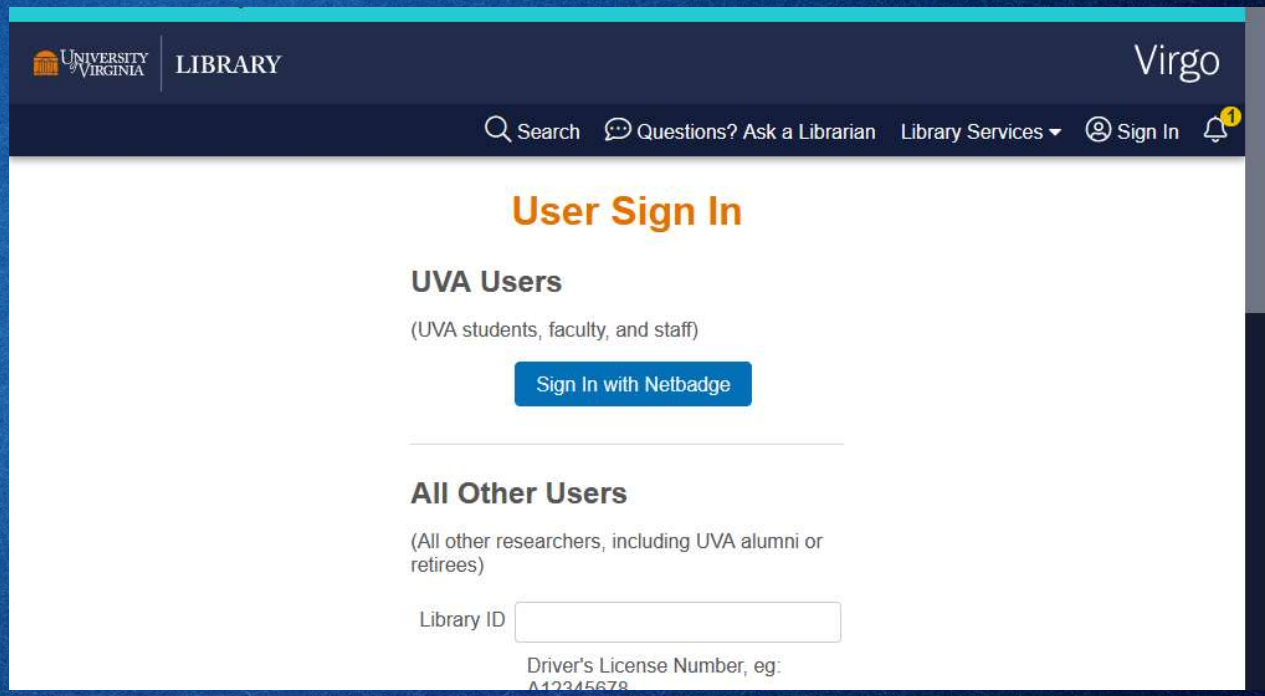
About the SWS requests…

- Question:
  - Which SWS style?
- Answer:
  - ROA, not RESTful
- First generation Symphony Web Services were RESTful
- One day support for RESTful style will end
- So we knew we wanted to use the newer style: ROA
  - ROA = resource-oriented architecture

Initial sign-in

UVA users use UVA's SSO "Netbadge".

Others use Library ID and Password stored in Symphony.



Academic SIG
A COSUGI Special Interest Group

Initial sign-in: the SWS request

- We use UVA's single sign-on platform for students, faculty, staff
- For community (non-UVA) patrons, we use SWS

URI =

```
/user/patron/authenticate
```

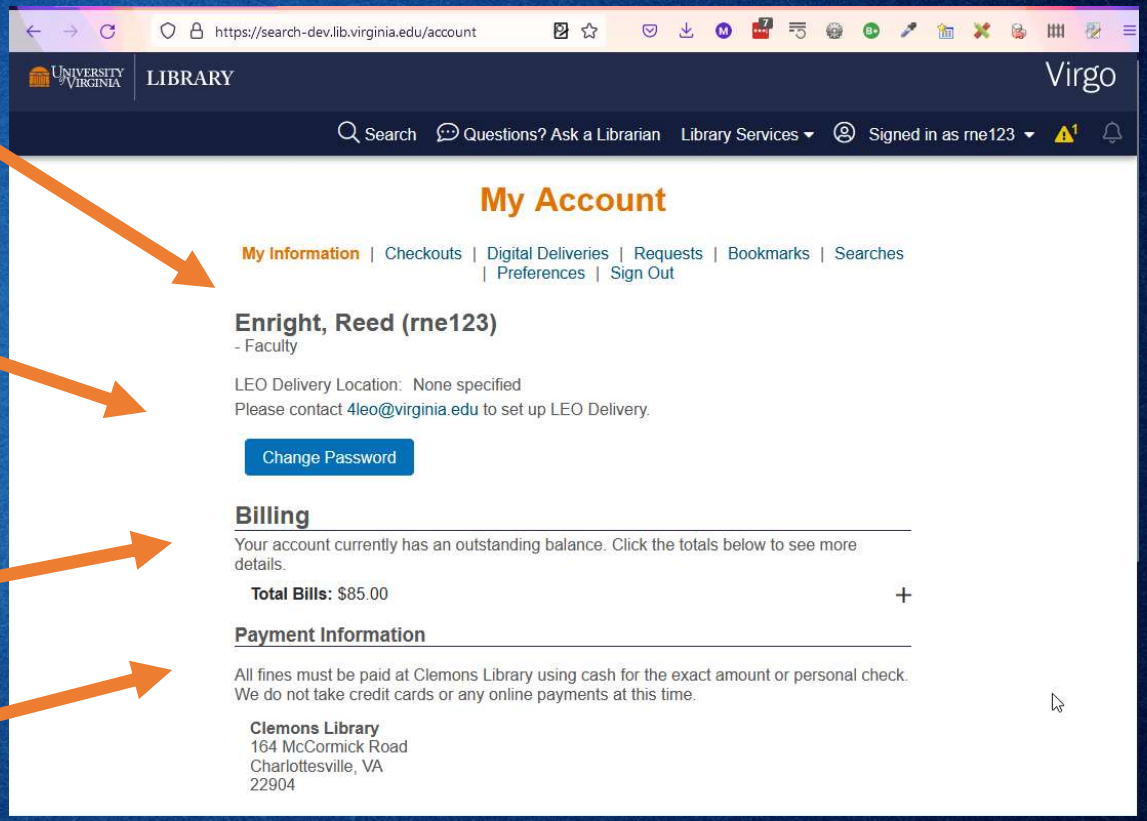
Request payload =

```
{"alternateID":"RNE123","password":"<* REDACTED *>"}
```

Academic SIG
A COSUGI Special Interest Group

"My Information"

- Name & address from SWS for community users; from NetBadge for UVA users
- Password change (PIN in Symphony) for community borrowers only
  - UVA users change password in UVA's "Netbadge system"
- Bills on main My Information page.
- Different advice on paying bills for UVA vs. community users



Academic SIG
A COSUGI Special Interest Group

My Information: the SWS request

```
/user/patron/search?
q=ALT_ID:$uva_computing_service_id
&includeFields=barcode,
   primaryAddress{emailAddress},
   displayName,
   profile{description},
   patronStatusInfo{standing,amountOwed},
   library
```

Academic SIG
A COSUGI Special Interest Group

Getting outstanding bills: SWS request

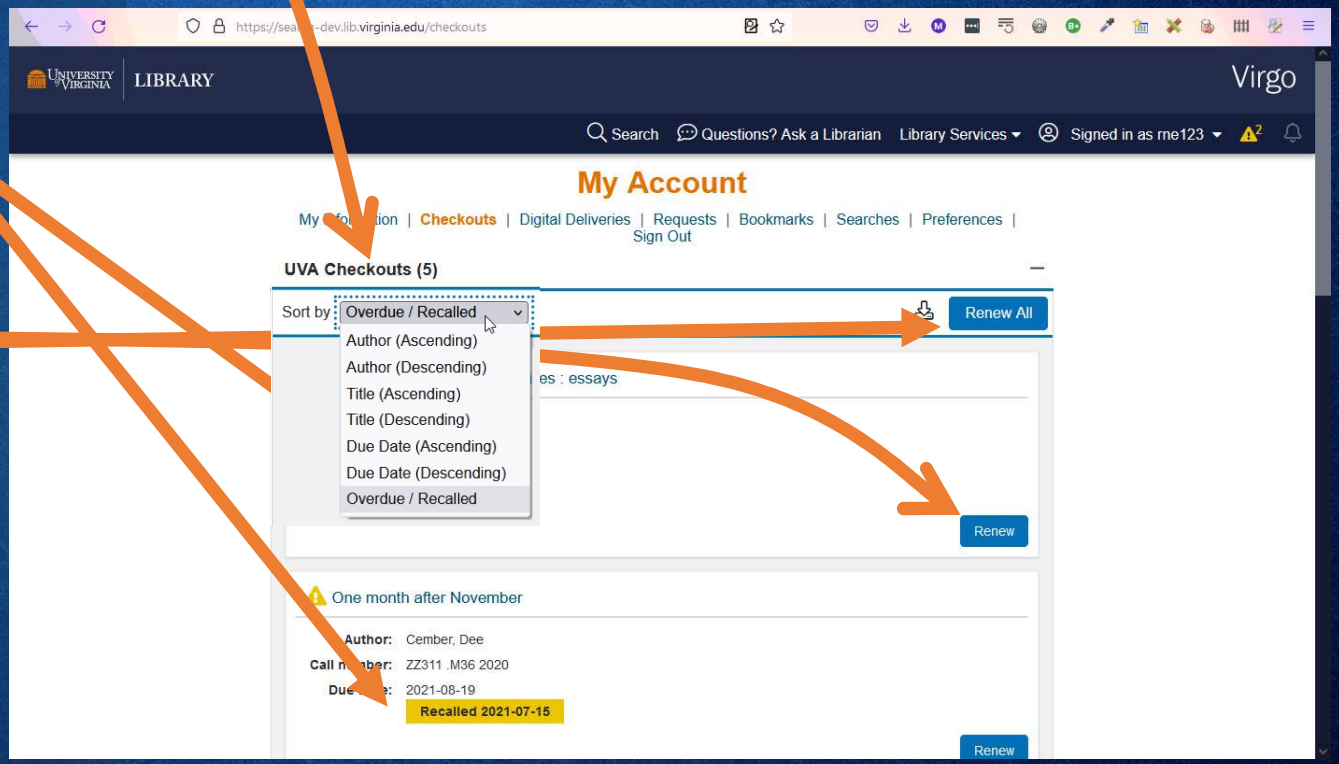
```
/rest/patron/lookupPatronInfo
    ?alternateID=$uva_computing_service_id
    &json=true
    &includeFeeInfo=UNPAID_FEES
```

This is actually a RESTful request. Will change to ROA one day … soonish?

Academic SIG
A COSUGI Special Interest Group

Checkouts

- Sort options
- Overdues and recalled checkouts are highlighted
- Renew one or renew all (“Renew All” button at top of list)
- Failed renewal displays error from SWS response



Academic SIG
A COSUGI Special Interest Group

Getting checkouts: the SWS request

```
/user/patron/search?
   q=ALT_ID:$uva_computing_service_id
   &json=true
   &includeFields=circRecordList{dueDate,
         overdue,estimatedOverdueAmount,recalledDate,
         renewalDate,library{description},
         item{barcode,call{dispCallNumber},
         bib{key,author,title}}}
```

Academic SIG
A COSUGI Special Interest Group

Renewals

For each renewal:

```
/circulation/circRecord/renew
```

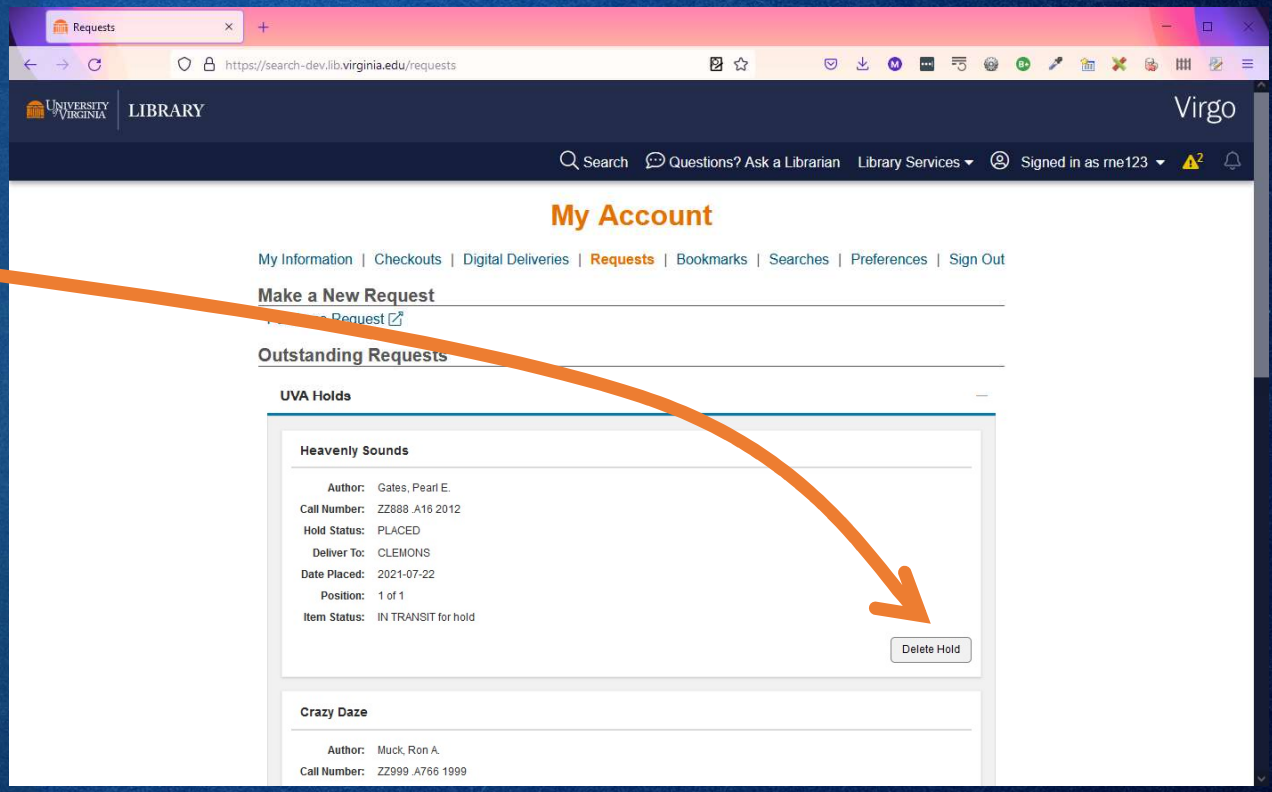
With this message body:

```
{ "itemBarcode": “$item_barcode" }
```

Academic SIG
A COSUGI Special Interest Group

Holds

- "Requests" tab shows Symphony holds
- List and delete holds
- Below UVA Holds are "ILL requests"
  - Virgo also uses ILLiad API!
  - Virgo also uses Aeon API (Special Collections requests)



Academic SIG
A COSUGI Special Interest Group

Holds: the SWS request

```
/user/patron/search?
   q=ALT_ID:$uva_computing_service_id
   &json=true
   &includeFields=holdRecordList
         {*,bib{title,author},
         item{barcode,currentLocation,library,
         transit{transitReason},
         call{dispCallNumber}}}
```

Academic SIG
A COSUGI Special Interest Group

Delete hold

- A simple ROA request:

```
/circulation/holdRecord/key/#{hold_key}
```

In theory, getting patron info could all be done in one SWS request:

- Get all components in one SWS request:
  - User information (name, address, user library, user profile)
  - List of checkouts, list of outstanding bills, list of active holds
- I think it would be more efficient
- Currently discussing with the UVA Library "Virgo Team"

```
/user/patron/search?
    q=ALT_ID:$uva_computing_service_id
    &json=true
    &includeFields=
        barcode,primaryAddress{emailAddress},
        displayName,profile{description},
        patronStatusInfo{standing,amountOwed},library,
        circRecordList{dueDate,
                overdue,estimatedOverdueAmount,recalledDate,
                renewalDate,library{description},
                item{barcode,call{dispCallNumber},
                bib{key,author,title}}},
        holdRecordList
                {pickupLibrary,placedDate,status,
                bib{title,author},
                item{barcode,currentLocation,library,transit
                  {transitReason,transitDate,destinationLibrary,
                   holdRecord},
                call{dispCallNumber}}},
        blocklist{createDate,amount,owed,
                block,library,billDate,
                callNumber,title}
```

Initial user info

Checkouts

Holds

Outstanding bills

Academic SIG
A COSUGI Special Interest Group

In conclusion

- Learning to use SWS has been challenging (for me) but rewarding
- I am slowly gaining facility with SWS

Academic SIG
A COSUGI Special Interest Group