

Almost Magical User Account Creation
and Normalized User Data
in Requests to Improve Reporting
(a two-and-a-half-part presentation)

Scott R. Anderson & Krista Higham
Millersville University Library

Atlas Systems, MakerSpace, October 13, 2020
A virtual event, actually held

Preface

- The part before Part the Firsttest
- Ask your questions as you have them
 - Chat your questions
 - Audible questions are also welcome
 - Gasps of glee or horror are also appreciated as feedback

The Blurby thing that brought you here

In this MakerSpace training and innovation event for ILLiad, Scott Anderson, Information Systems Librarian and Krista Higham, Access Services Librarian, share how Millersville University is using single sign-on technology (OpenAthens in particular) to create and update ILLiad user accounts on the fly (no more first time patron registration or patron loads or manual patron updating).

Millersville is also in the early stages of flowing university provided categories of user data (faculty, staff, student; major; minor) to requests. Including user category data at the time of request should improve the accuracy of historical reporting. Category data about a patron remains persistent over time when included with the request and is it not subject to retroactive changes when patron information is updated (Example: requests placed by biology major in 2018, remain “biology” requests; despite the student changing major to chemistry in 2019 and making subsequent requests.)

About Millersville University (MVS)

- Public institution
- Mid-sized masters level + three professional doctorate programs
- ~6,400 FTE (undergrad + grad) / ~300 FTE faculty
- Library houses 280k physical volumes with 335k records in the catalog
 - No longer accessioning physical resources except as last resort
- Uses ILLiad, RapidILL, PALCI's E-ZBorrow (OCLC's D2D; soon to be ReShare)
- Also uses EBSCO Discovery Service

- "Mostly remote" through the Spring 2021 semester.

Part the First

- Also known as the first part of the presentation.

Why use SSO magic?

- Transitioned to “nearly all electronic” library back in 2010
 - Building renovation accelerated approach to e-content and “delivery” in general
- General approach to avoid redundancies and “annoyances”
 - If we know something about a user, don’t ask for it again
 - If we authenticated a user 10 minutes ago, don’t force them to do it again
- Avoid “library centric accounts”
 - Avoid requiring accounts that are not university directory accounts
 - Access is a function of institutional affiliation, leverage that affiliation
 - Access to library then generally determined by the UNIVERSITY
 - Death to the barcode

Single Sign-On (SSO) with OpenAthens

- Identity and access management tool
- Use institutional accounts (AD) or create accounts in OpenAthens
- Supports secure access using SAML 2.0, for transmitting user data
- Entirely cloud-based service (no need for customer to host)
 - With 24/7 customer support
 - MVS works through EBSCO (North America),
 - EBSCO sends improvements / suggestions to OA as necessary

Single Sign-On (SSO) at Millersville (MVS)

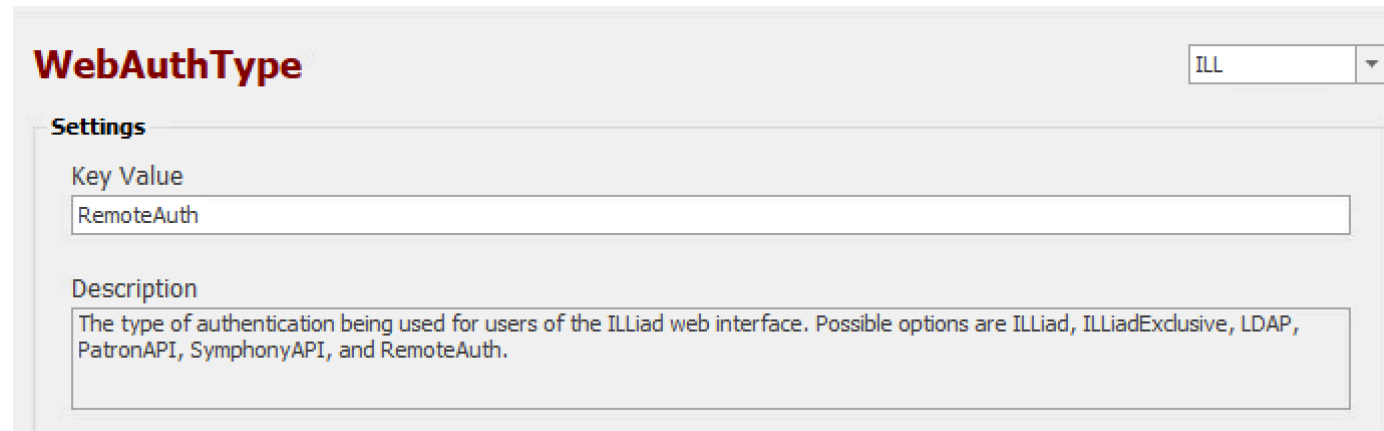
- MVS “go live” with OpenAthens (OA) in August / Sept 2018
 - ILLiad was moved from AD to OA well after our “go live” dates for content
 - 12 months lead time for OA due to dual institution development work (not typical)
- MVS flows data from SIS > AD > Shib > OA
 - University ID to userinfo4; then manual into ILLiad ID field
 - Student “Major” flows into userinfo1
- Consistent *categories* of data
 - User Type (Status in ILLiad)
 - Faculty, staff, student
 - Department (only for faculty and staff)
 - Major (only for students)
 - Minor (only for students)

Single Sign-On (SSO) at Millersville (MVS)

- Will need “actionable” data into AD / SSO service
 - Many institutions populate SAML services “sparsely”
 - Intended to answer “yes or no” access questions
 - Attributes allow for actual “assessment” / “analytics”
 - Informs collection development / services
 - What category of user is requesting – eg. Biology student, Geography Faculty

How to do SSO and ILLiad?

- Match existing accounts? Start all over? Create new leaving the old?
 - MVS username is the same in AD as it is in OA
 - So auth mechanism changed (AD to OA), but not account
- Activate “RemoteAuth” for WebAuthType



WebAuthType ILL

Settings

Key Value
RemoteAuth

Description
The type of authentication being used for users of the ILLiad web interface. Possible options are ILLiad, ILLiadExclusive, LDAP, PatronAPI, SymphonyAPI, and RemoteAuth.

How to do SSO and ILLiad?

- Shibboleth activated on machine hosting your ILLiad instance
- Getbuild file needs to be “mapped” so machine knows what to expect
 - This is the file that lets data into ILLiad the application through the machine
 - Atlas did this for MVS; this is not yet customer facing

MVS Example:

```
<Attribute name="username" id="uid"/>  
<Attribute name="surname" id="lastname"/>  
<Attribute name="forenames" id="firstname"/>  
<Attribute name="emailAddress" id="email"/>  
<Attribute name="urn:oid:MNumber" id="mnumber"/>  
<Attribute name="department" id="department"/>  
<Attribute name="major" id="major"/>  
<Attribute name="minor" id="minor"/>  
<Attribute name="status" id="status"/>  
<Attribute name="LibraryID" id="LibraryID"/>
```

How to do SSO and ILLiad?

- Configuration RemoteAuthValidation table
 - This is the place to tell ILLiad, the application, where to put attributes
 - Attributes we are transferring from OA into ILLiad
- Also configuration “acceptance” or “modification” rules as well

RemoteAuthValidation

ID	NVTGC	ILLiadFieldName	RemoteFieldName	Validation	ValidAction	InvalidAction	ValidDefault	InvalidDefault	Overwrite	LogIfChanged
1	ILL	Username	REMOTE_USER	.+	accept	reject			No	<input type="checkbox"/>
2	ILL	FIRSTNAME	FIRSTNAME	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>
4	ILL	SSN	MNUMBER	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>
5	ILL	LASTNAME	LASTNAME	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>
6	ILL	EMAILADDRESS	EMAIL	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>
8	ILL	DeliveryMethod	UID	.+	substitute	substitute	Hold for Pickup		Yes	<input checked="" type="checkbox"/>
9	ILL	LoanDeliveryMethod	UID	.+	substitute	substitute	Hold for Pickup		Yes	<input checked="" type="checkbox"/>
11	ILL	Department	DEPARTMENT	.+	accept	substitute			Yes	<input checked="" type="checkbox"/>
12	ILL	UserInfo1	major	.+	accept	ignore	ValidDefaultUI1		Yes	<input checked="" type="checkbox"/>
13	ILL	UserInfo2	minor	.+	accept	ignore	ValidDefaultUI2		Yes	<input checked="" type="checkbox"/>
16	ILL	STATUS	status	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>
18	ILL	ElectronicDelivery	UID	.+	substitute	substitute	Yes	Yes	Yes	<input checked="" type="checkbox"/>
19	ILL	UserInfo4	LibraryID	.+	accept	ignore			Yes	<input checked="" type="checkbox"/>

Benefits of SSO for user and staff

- No more user "account creation" process
 - Stop asking users for information we already know
- Updates ILLiad account with fresh information with each user login
 - ILLiad PATRON information is always up to date for staff
- Institution determines when users are ineligible for service
 - Library is not arbitrating between user and the institution at large
- Consistent university directory data
 - Very confident that we'll get the data that we need / want for operations
 - Example: we ONLY use the university e-mail address

Part the First and a half

- What's the user experience?
 - It's hip
 - It's cool
 - It's pretty simple
 - It's actually kinda boring

General approach to login / authentication

- Users need to login / authenticate before using
 - Any resource or service
 - Regardless of location – it's the same on or off campus
 - Regardless of weather or season of the year
 - ... or even major world events

Sign-on / account creation process

Millersville University FRANCINE G. MCNAIRY LIBRARY & LEARNING FORUM





Library Search Our Website

Our collections & more: Keyword Title Author

[Advanced Library Search](#)

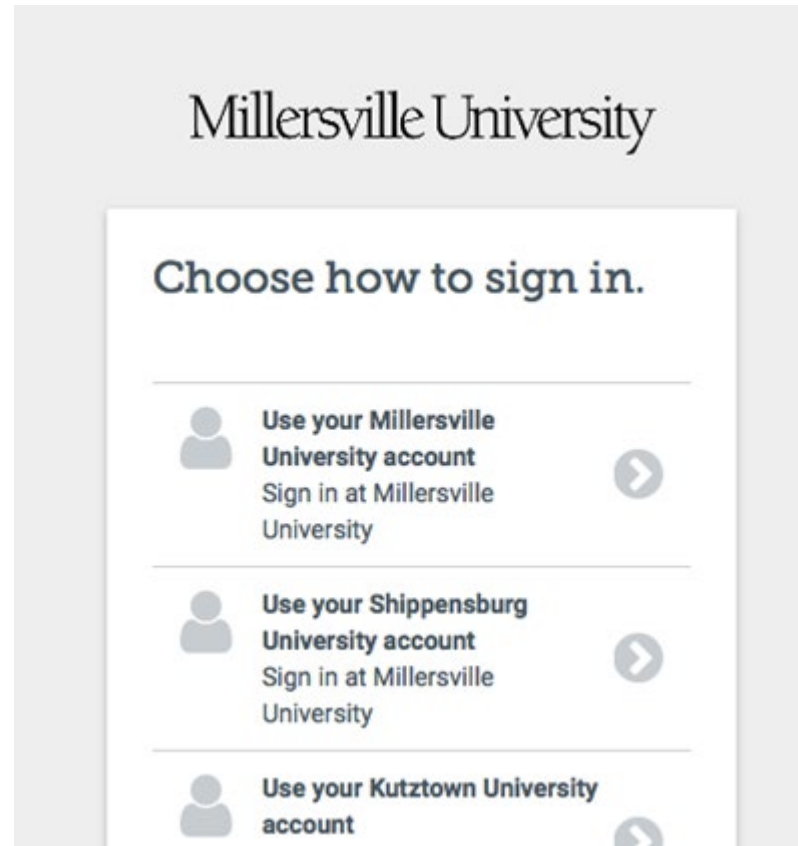
MORE RESEARCH TOOLS: [Browse By Subject](#) | [All Databases A to Z](#)

ASK A LIBRARIAN

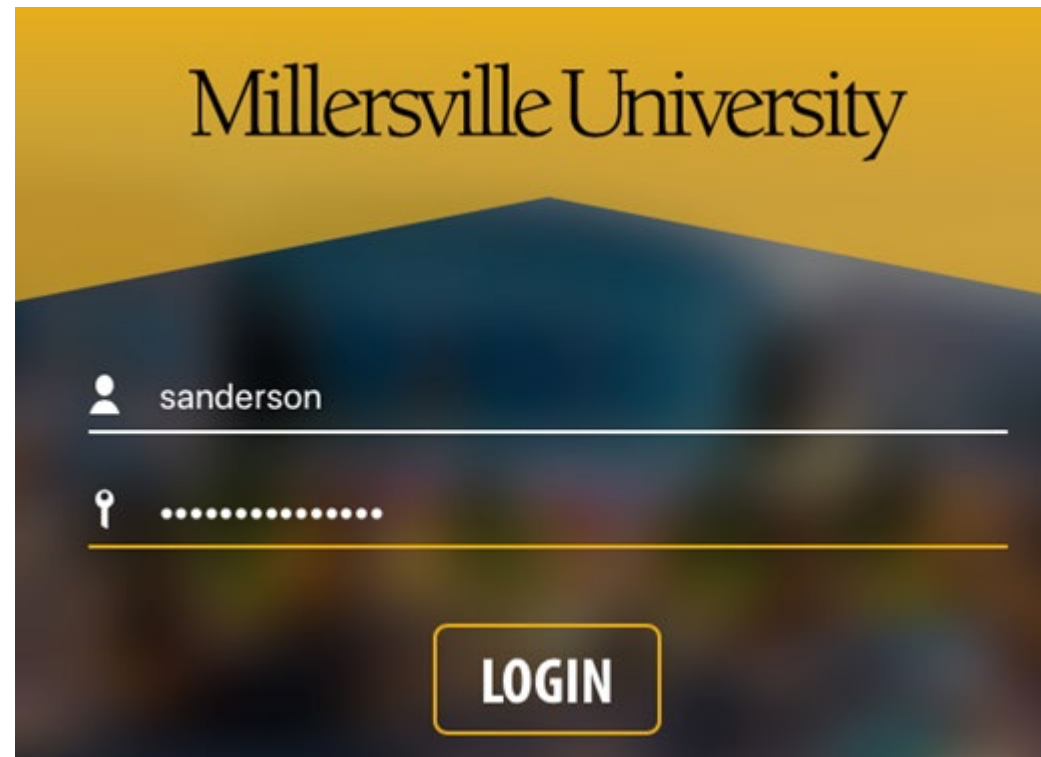
 CHAT OFFLINE But you can still get help	 CALL 717-871-7110	 APPOINTMENT Consult with us	 E-MAIL Contact form
---	---	---	---

Sunday: 6pm-9pm, Monday-Thursday: 9am-5pm, 7pm-9pm, Friday: 9am-12pm (hours vary during university breaks)

Choose how to sign in



Millersville account chosen



Millersville University

LOGIN

The image shows a login interface for Millersville University. At the top, the university's name is displayed in a serif font against a yellow background. Below this, the background transitions to a dark blue and black gradient. There are two input fields: the first is for the username, containing the text 'sanderson', and the second is for the password, represented by a series of dots. A yellow 'LOGIN' button is positioned at the bottom center of the form.

Regular ole' citation (about trains!)

1. Monitoring and Analysis of Dynamic Behaviour of **High-Speed Trains** in **Long-Term Service** using an Unattended Test System.



Academic
Journal

By: Xin Ding; Xingqiao Ai; Yongsheng Li; Jintian Wang; Zhendong Liu. *International Journal of COMADEM* , Jul2019, Vol. 22 Issue 3, p45-50, 6p, Database: [Applied Science & Technology Source](#)

Dynamic behaviour and its evolution are concerns for those operating **high-speed trains**. In this paper, a cloud server is applied to establish a connection between the vehicle-mounted equipment an...

Subjects: **High speed trains**; Data transmission systems; Intelligent transportation systems; China; Test systems; Behavior; Telemetry



[Request this item](#)

Request It

[Main Menu](#)[New Requests](#) ▾[History](#) ▾[Profile](#) ▾[Logoff](#)

Enter information below and press the Submit Request button to send.

Article Request

Journal Title *(required)*

Please do not abbreviate the journal title.

Article Title *(required)*

Article Author

Questions thus far?

- Now is the time for questions!
 - Call an audible or submit a question via chat
 - Only on part of the first and half!
- If we don't have answer, we'll fabricate something
 - It will be reasonable, plausible, and perhaps deniable later
- Humorous point #2 here
 - Snarky response here

Part the Second

- Also known as the second part of the presentation
 - Typically follows, Part the First or Part the First and a half
- Krista wanted the second part first, but was vetoed
 - Presentation committee voted down the proposal

Why flow user categories into transactions?

- Because we could
- We wanted categorical user information at time of submission
 - Not user information at time of reporting
- Categorical USER data does NOT change in transactions table
 - Requests by CHEM major in Fall 2019, remain CHEM requests forever
 - even if student becomes BIO major in Spring 2020
 - Improves “historical” reporting opportunities
 - Simplifies reporting to a single table
 - Takeaway : “at time of request” vs. “at time of report”

How to flow categories into transactions?

- Adjusting HTML at GitHub for MVS
 - /RemoteAuth directory (cause we're using SSO, which is "RemoteAuth")
 - Create new "include_user_attributes.html" file in /RemoteAuth
 - So you can manage all the data flows in one place
 - Insert "include" into appropriate user forms just above "form state"
 - FIVE forms in Millersville's use case

New "include_user_attributes.html" file

```
<div id="additional_user_attributes">  
  <input type="hidden" name="ItemInfo1" value="<#USER field='UserInfo1'>">  
  <input type="hidden" name="ItemInfo2" value="<#USER field='UserInfo2'>">  
  <input type="hidden" name="ItemInfo4" value="<#USER field='UserInfo4'>">  
</div>
```

It's hidden, so the user doesn't see anything

Moves "UserInfoN" (user table) into "ItemInfoN" (transaction table)

Code snippet to insert to include the include

```
<input type="hidden" name="NotWantedAfter" value="<#PARAM name='NotWantedAfter'>">
```

```
<!-- 2020 09 08 to pull in user attributes into the item request, a new include file. -->
```

```
<#INCLUDE filename="include_user_attributes.html">
```

```
<#FORMSTATE>
```

ArticleRequest.html

BookChapterRequest.html

ConferencePaperRequest.html

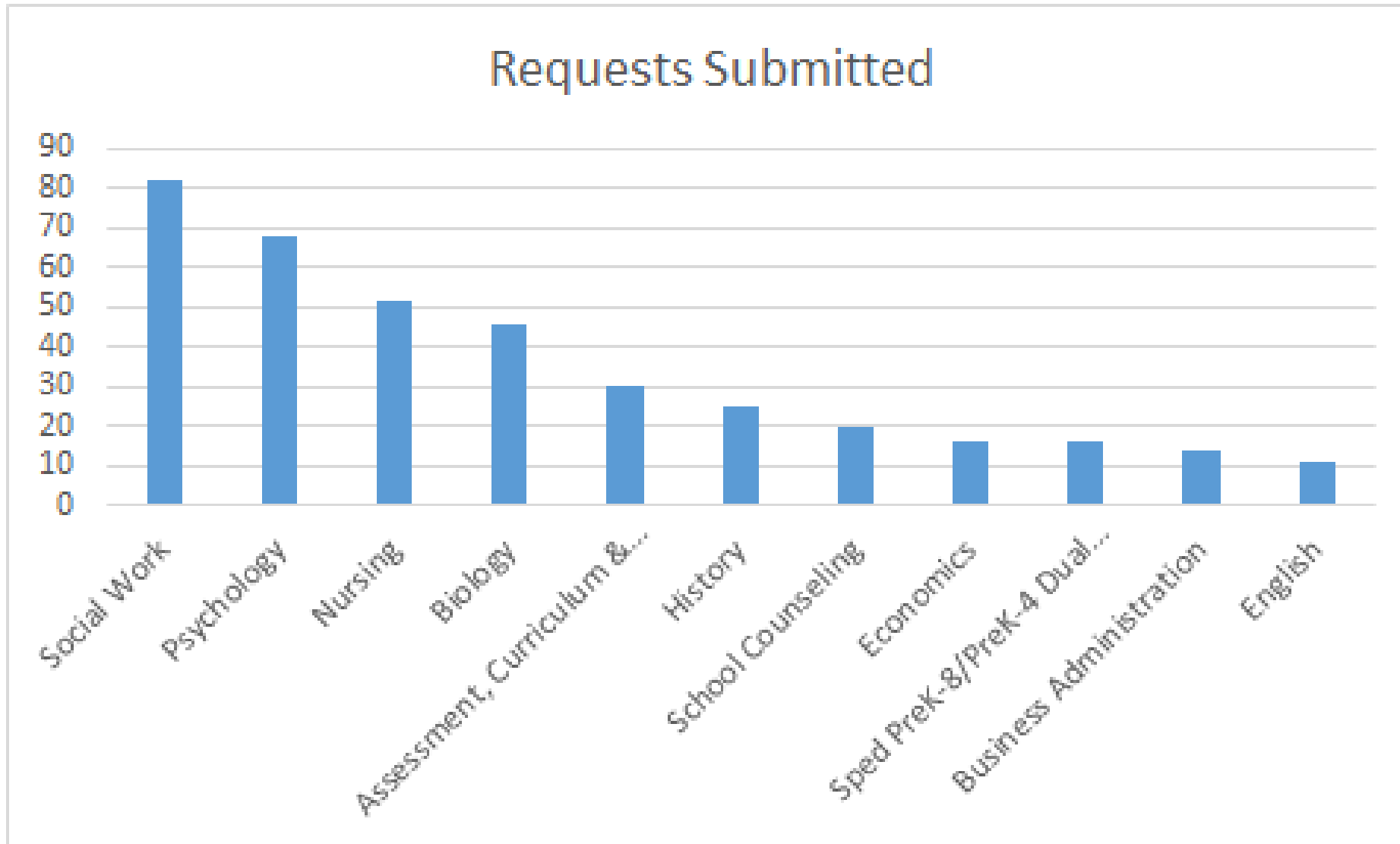
LoanRequest.html

MultimediaRequest.html

Categorical user information in transactions

- ItemInfo1 contains student “Major(s)”
- ItemInfo2 contains student “Minor(s)”
- ItemInfo3 [currently empty]
- ItemInfo4 contains SIX digit Library ID (barcode) number
 - Manually transformed into 14 digit number and placed into “ID” field
- ItemInfo5 contains manually determined grad/undergrad attribute

Reports from ILLiad transactions table

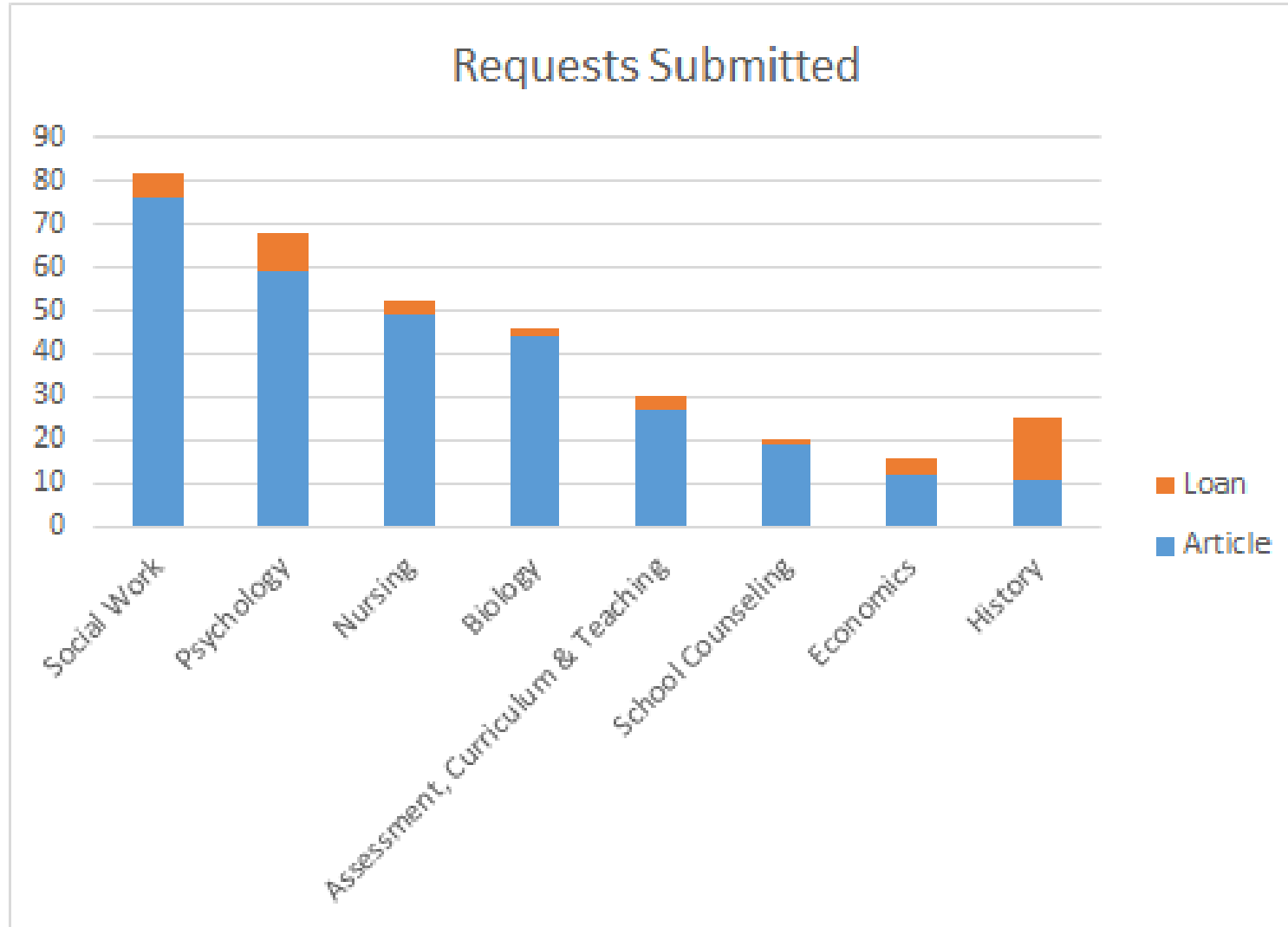


September 10, 2020 –
October 6, 2020

Students

Majors with 10 or more
requests

Article and Loan Requests

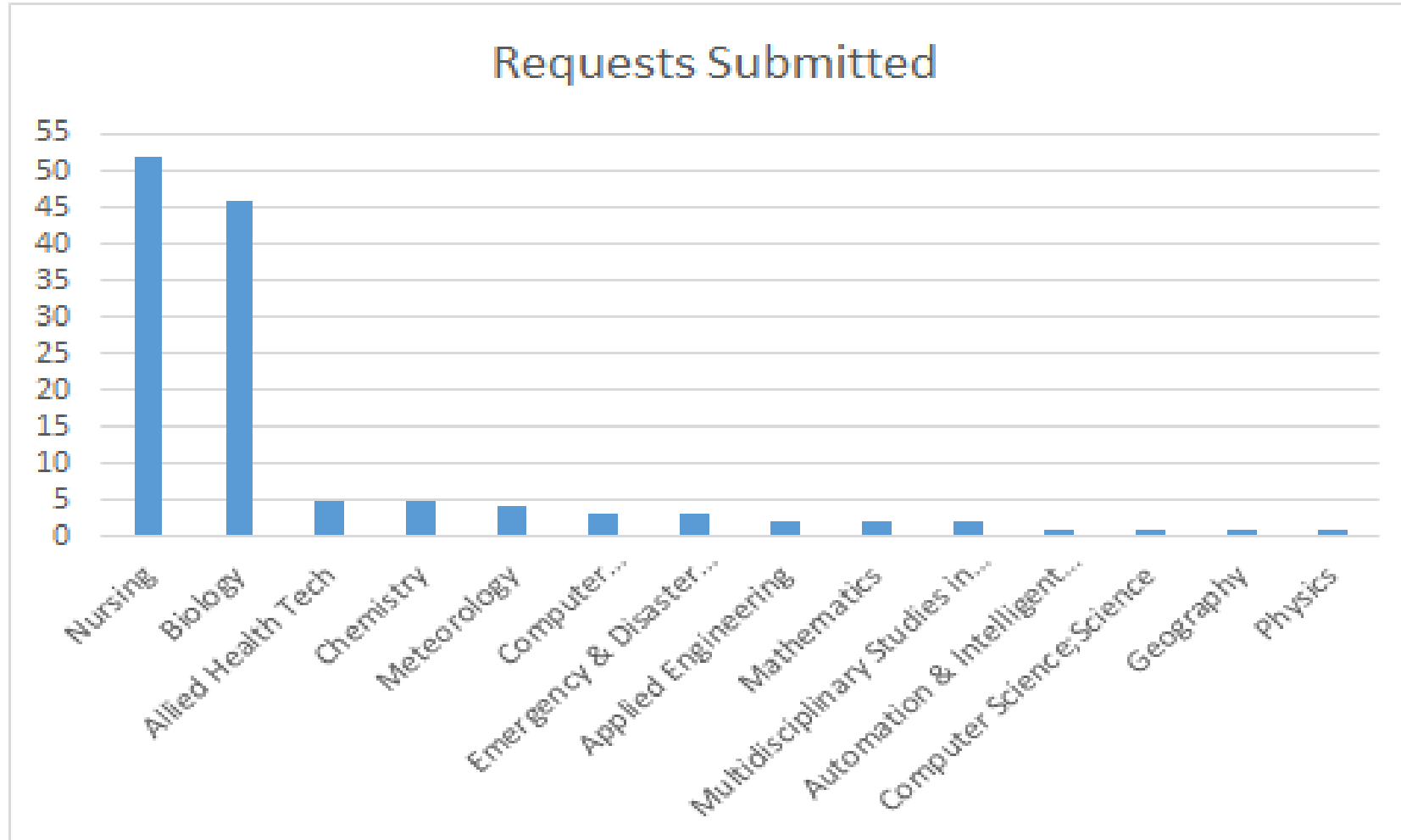


September 10, 2020
– October 6, 2020

Students

Top 8 majors
with 10 or more requests

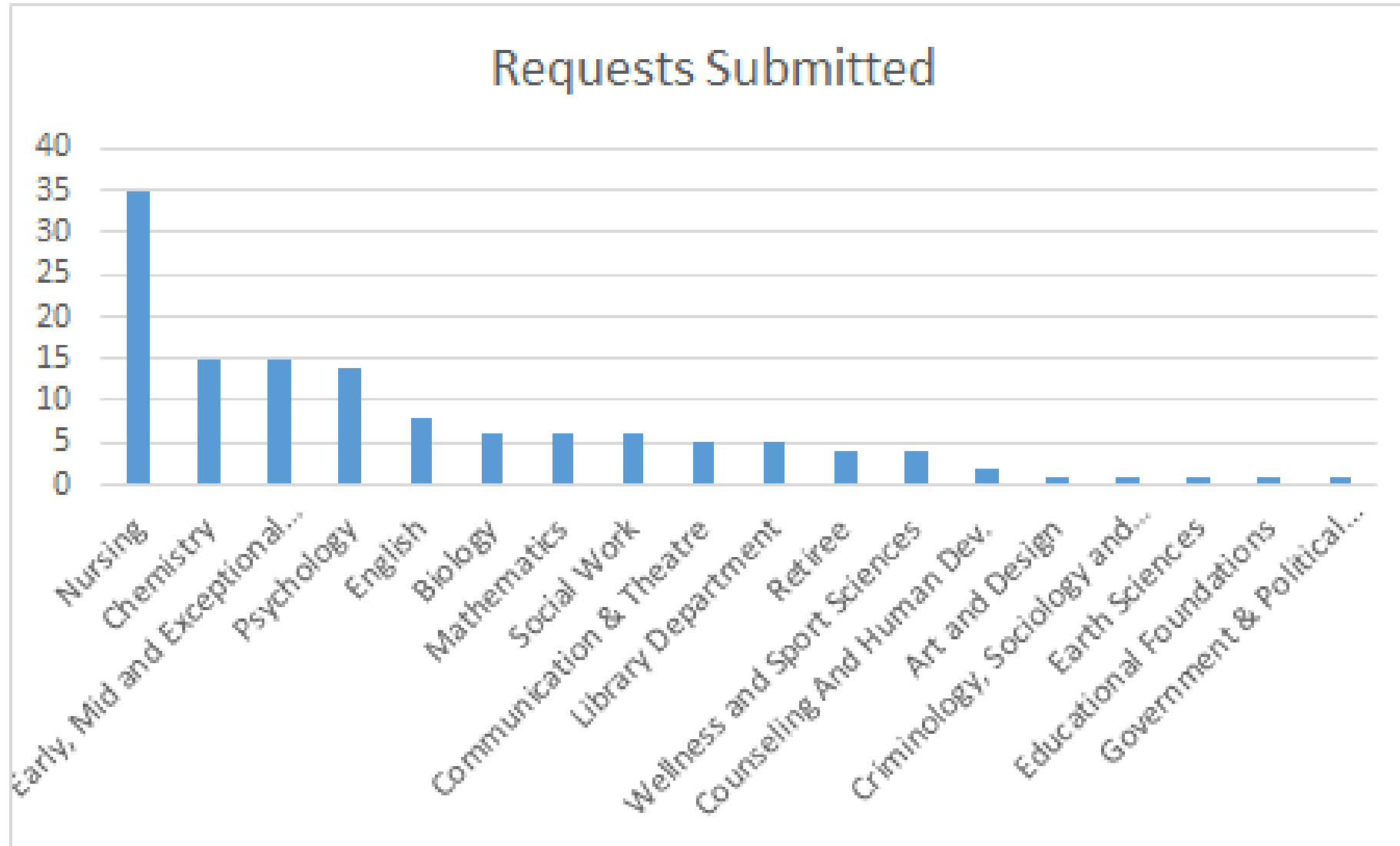
School of Science & Math Requests



September 10, 2020
– October 6, 2020

Student requests by
Department

Faculty Requests by Department



September 10, 2020

– October 6, 2020

Faculty

Derived statements from transactions report

- In the last N days, Psych students have placed N requests for articles
- In the last 30 days, faculty have placed N requests for returnables
- In the Spring semester, our top requesting majors are
 - Eventually, compared to two Springs ago, that represents an X% change
- How requests filled? What % from what service?
- We can very readily make these determinations on category
 - Vs. previously teasing apart user data

Some “Part of the Last” thoughts

- How else might this data be useful ?
 - Targeted instruction for identified majors
 - Administrators are increasingly data driven
 - Perhaps additional funding
 - Collection Development for high requesting majors / disciplines
 - Improved identification of resources instead of ILL
 - Coupled with OpenAthens login data?
 - Assessment ?
 - Excellent trivia opportunities ...

Any questions?

- Now is the time!
- Feel free to contact us at
 - Scott.Anderson@millersville.edu
 - Krista.Higham@millersville.edu