

# Sustainability Case Study MetaArchive

Nick Krabbenhoeft

# Overview

---

- Introduction to the MetaArchive
- Sustainability Challenges
  - Member Changes
  - Internal Changes
- Lessons Learned

# Introduction to the MetaArchive

---

# What is MetaArchive

---

A distributed digital disaster recovery service and digital preservation community

19 Members

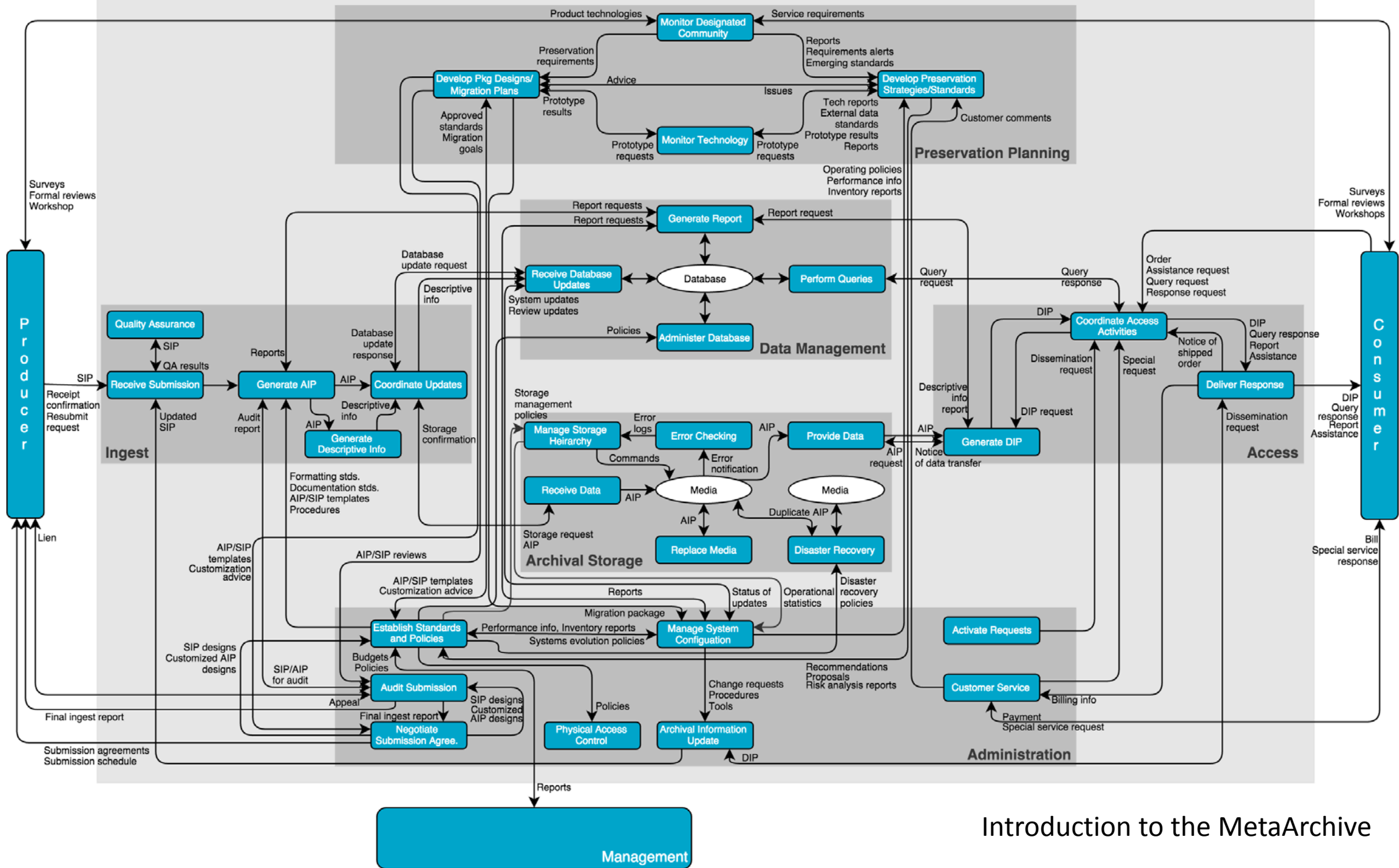
- Mostly academic libraries
- 1 public library

1 museum

# How the MetaArchive differs from CLOCKSS

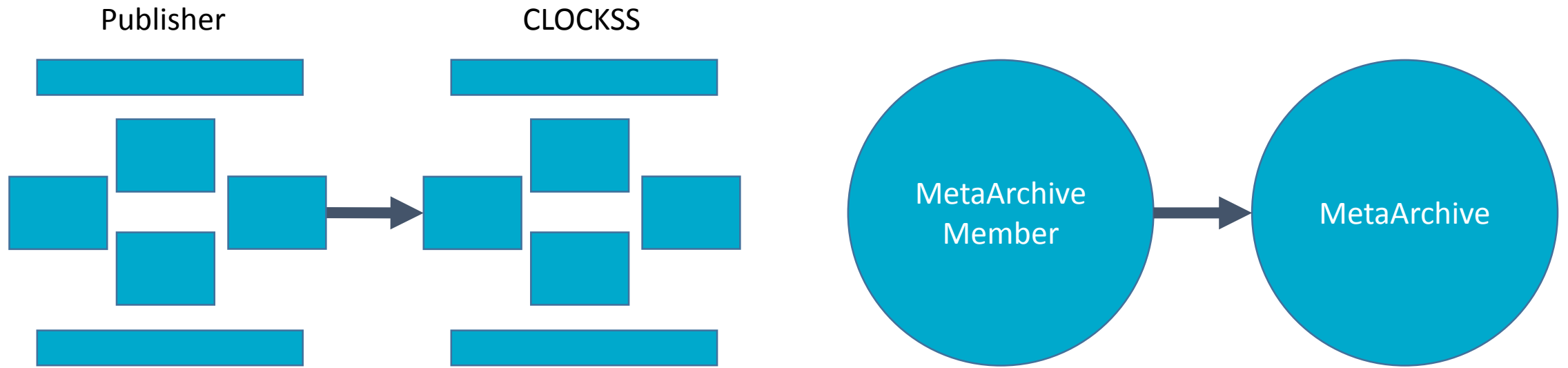
---



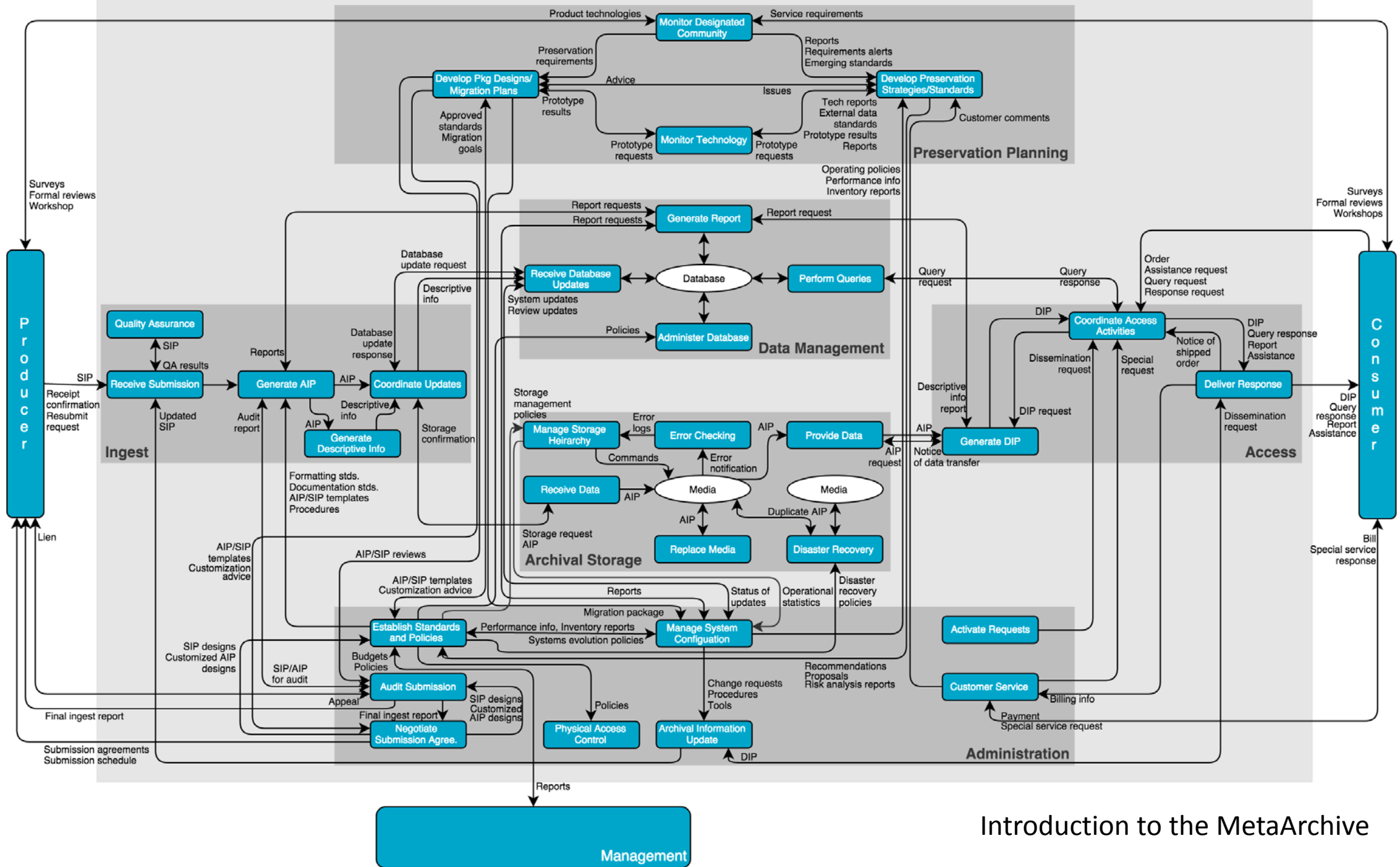


# How we're different from CLOCKSS

---

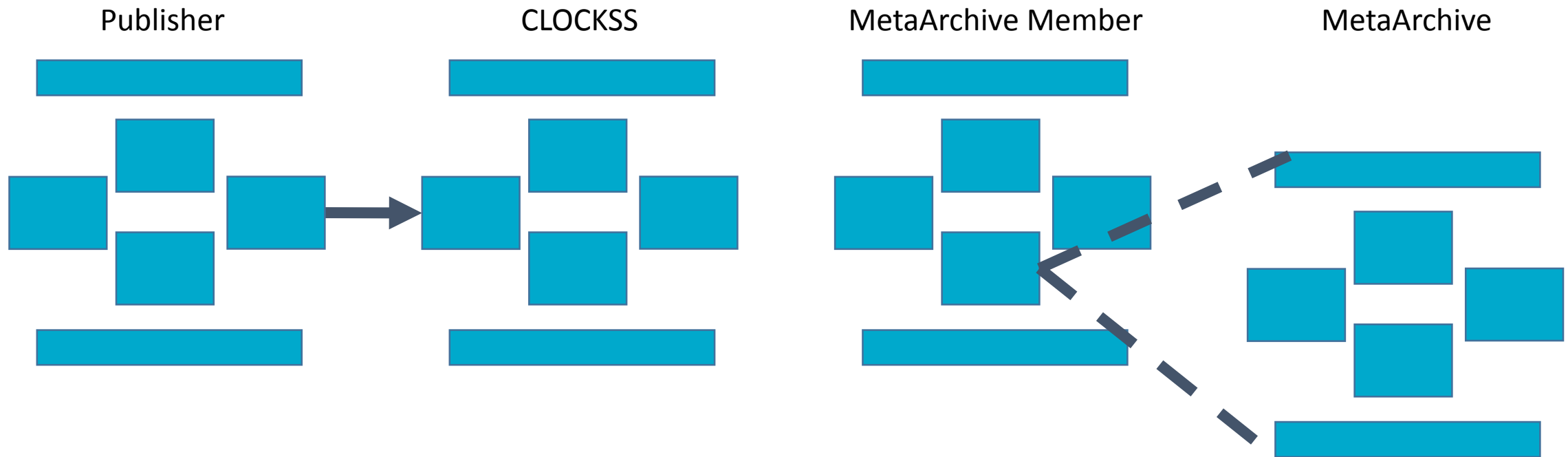


CLOCKSS harvests produced journal volumes





# How we're different from CLOCKSS



CLOCKSS harvests produced journal volumes (DIP)

MetaArchive harvests collection source objects (AIP)

# Sustainability Challenges

---

# Sustainability Challenges

---

Our cooperative is comprised of

- a diverse group of libraries with
- a variety of repository systems

We keep costs down by working with

- a small central staff
- outsourced tools where possible

# Changes

---

- Member Changes
  - Membership
  - Repositories
- Internal Changes
  - Staffing
  - Technology

# Member Changes

---

# Changing Repositories

---

MetaArchives members currently use a number of different systems.

- Dspace
- bepress
- HydraDL Consulting
- ContentDM
- Custom
- ...

# Changing Repositories

---

Most repositories are built for user access not HTTP crawls of the backend. We sneak in however we can:

- OAI-PMH crawls
  - Allow flexible AU definition
  - Require OAI-PMH support and stable URIs
- BagIt exports
  - Allow more control of AU contents and URIs
  - Require double storage or hands-on management of recrawls

# Changing Repositories

---

While we would like to create stable ingest pathways, we are not a high priority user for repositories. In discussion, some have even said “We already do preservation.”

Custom pathways for every member create greater administrative overhead, especially as hacks have to be maintained over time.



# Changing Membership

---

What constitutes an AU or a collection?

2 Case Studies

- The Museum
- The Consortium

# Changing Membership – The Museum

---

In 2015, an art museum joined the MetaArchive.

It will preserve:

- Collections from the archives
  - 1 AU = the Papers of Artist A
- Images and metadata from the art collections
  - 1 AU = the Art of Movement X?  
the Acquisitions of Year Y?  
the Products of Digitization Project Z?

# Changing Membership – The Museum

---

One member uses the MetaArchive for its ETD (electronic thesis and dissertations) collection.

The entire collection is divided by departments.

Multiple departments are used to fill an AU until it reaches 20gb.

## AU 1

- Agriculture (10gb)
- Chemistry (10gb)

## AU 2

- History (9gb)
- Mathematics (11gb)

## AU 3

- Performing Arts (19gb)

# Changing Membership – The Museum

---

What happens when the departments change?

For example, adding an art history program.

## AU 1

- Agriculture (10gb)
- +Art History (1gb)
- -Chemistry (10gb)

## AU 2

- +Chemistry (10gb)
- History (9gb)
- -Mathematics (11gb)

## AU 3

- +Mathematics (11gb)
- -Performing Arts (19gb)
- +AU 4
- +Performing Arts (19gb)

# Internal Changes

---

# Staffing

---



# Staffing - 2012

---

- Program Manager – Matt Schultz
- Programmer – Monika Mevenkamp
- Contractors
  - Sys Admin – Chris Helms

# Staffing - 2013

---

- Program Manager – Matt Schultz
- Contractors
  - Sys Admin – Chris Helms
  - Cache Admin – Clay Miller



# Staffing - 2014

---

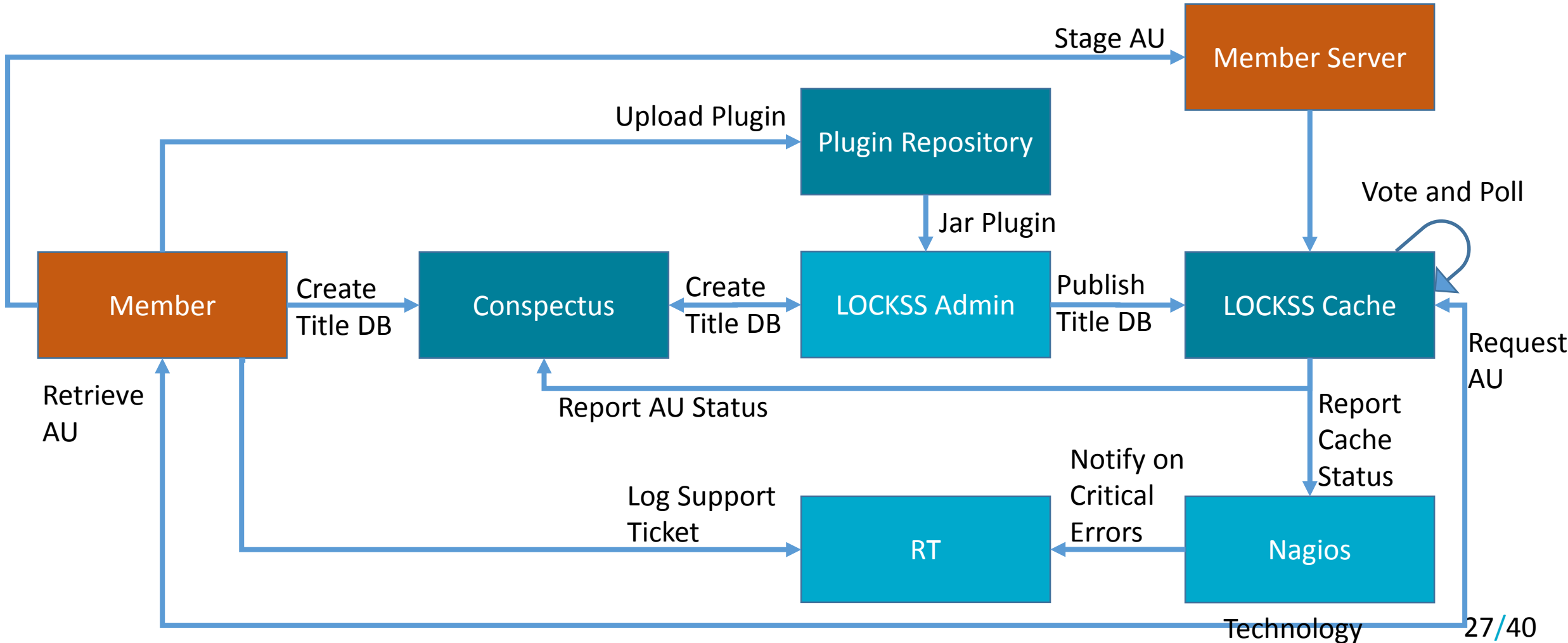
- Program Manager – Matt Schultz
- Contractors
  - Sys Admin – Chris Helms
  - Cache Admin – Clay Miller
  - Programmer – Stephen Eisenhauer

# Staffing - 2015

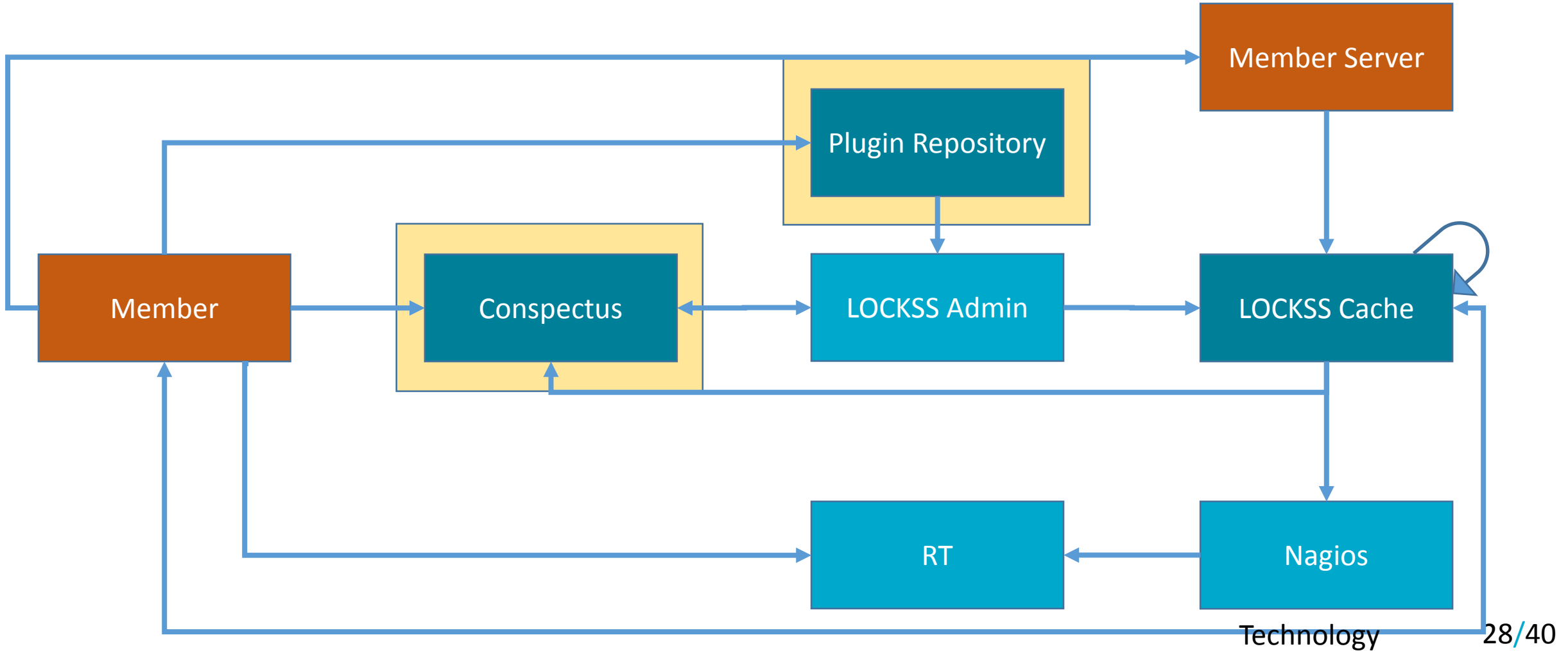
---

- Community Manager – Sam Meister
- Project Manager and Systems Analyst – Nick Krabbenhoeft
- Contractors
  - Sys Admin – Chris Helms
  - Cache Admin – Clay Miller
  - Programmer – Kurt Nordstrom

# Technology



# Technology Changes



# Conspectus Migration

---

- We replaced several Ruby applications with a single Conspectus written in Django since our contractors were more familiar with Python.
- However, we have not migrated all functionality and continue to run the older Ruby applications for network administration views.

# Repository Migration

---

- GoogleCode announced that it would turn read-only in August 2015
- MetaArchive uses Subversion for
  - a public repository of plugins
  - a local repository of network configuration
- Plugin repository migrated to GitHub in August 2015
- Local repository remains in subversion for now

# Lessons Learned

---

- We aren't able to demand direct access to the source data, so initial conversations are the most important when it comes to the long-term disposition of content.
- Documentation is everything.
  - Formal documentation – wikis, manuals
  - Informal documentation – version control, screencasts