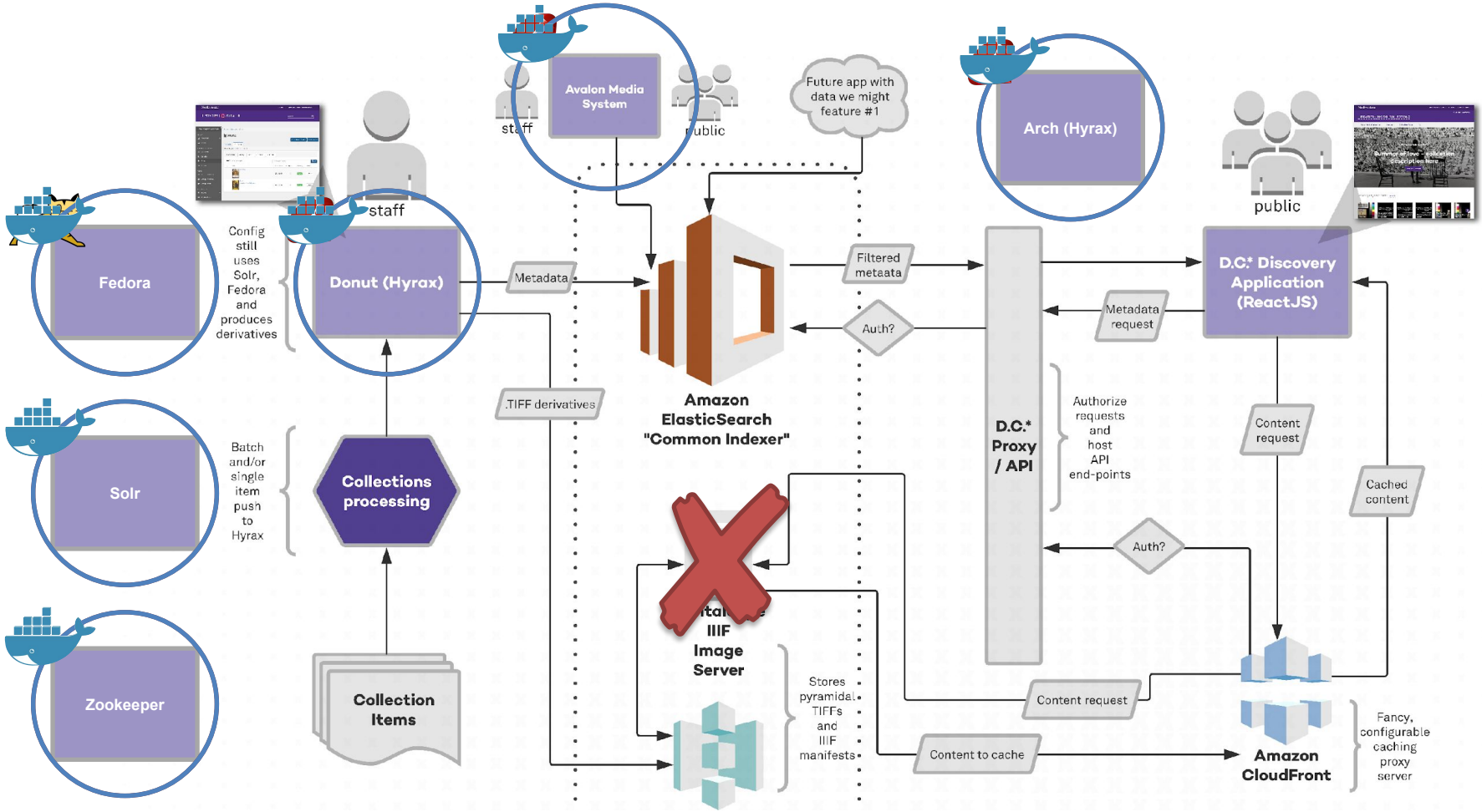
Northwestern
UNIVERSITY LIBRARIES

New Directions for Northwestern
Taking a Cloud-First Approach

Michael B. Klein
Samvera Virtual Connect 2019
April 24, 2019


Avalon Media System
staff
public
Future app with data we might feature #1
Arch (Hyrax)
public
staff
Fedora
Config still uses Solr, Fedora and produces derivatives
Donut (Hyrax)
Metadata
Filtered metaata
Auth?
Amazon ElasticSearch "Common Indexer"
D.C.* Proxy / API
Metadata request
D.C.* Discovery Application (ReactJS)
Authorize requests and host API end-points
Content request
Cached content
Solr
Batch and/or single item push to Hyrax
Collections processing
.TIFF derivatives
IIIF Image Server
Stores pyramidal TIFFs and IIIF manifests
Auth?
Content request
Content to cache
Amazon CloudFront
Fancy, configurable caching proxy server
Zookeeper
Collection Items

Learning As We Go

Services, Not Servers

• Database	RDS Postgres
• Storage	S3
• Transcoding	Elastic Transcoder
• Caching	ElastiCache Redis
• Search Index	ElasticSearch
• Streaming	CloudFront
• Monitoring	CloudWatch Alarms
• Logging	CloudWatch Logs
• Messaging	SQS/SNS/SES

Services, Not Servers

- IIIF as a serverless application (Lambda)
  - Blazingly fast
  - No servers or environment to maintain; only code
  - Scales immediately and intelligently

Services, Not Servers

- ElasticSearch vs. Solr
  - SolrCloud is by far the most problematic piece of our infrastructure
    - Scaling, shards, replicas, nodes
    - Upgrading in lockstep without downtime
    - Backup and Restore
    - Disaster Recovery
    - $$$

Separate Public & Staff Tools

- Break out a public-facing tool that focuses on the needs and habits of patrons
- Ingest & Admin tasks shouldn’t drain resources from Discovery & Access
- Staff & End Users don’t keep the same schedules
  - upgrading one tool shouldn’t impede the other

Twelve Factor Methodology

In the modern era, software is commonly delivered as a service: called web apps, or software-as-a-service. The twelve-factor app is a methodology for building software-as-a-service apps that:

- Use declarative formats for setup automation, to minimize time and cost for new developers joining the project;
- Have a clean contract with the underlying operating system, offering maximum portability between execution environments;
- Are suitable for deployment on modern cloud platforms, obviating the need for servers and systems administration;
- Minimize divergence between development and production, enabling continuous deployment for maximum agility;
- And can scale up without significant changes to tooling, architecture, or development practices.

The twelve-factor methodology can be applied to apps written in any programming language, and which use any combination of backing services (database, queue, memory cache, etc).

Source: https://12factor.net/

The App Runs Everywhere and Nowhere

- Don’t assume tasks share disk storage
- In fact, don’t assume disk storage
- URLs, not Pathnames
- Move data as little as possible
- Retrieve only what’s needed
- Environment config > File-based config
- **Don’t let scaling and concurrency be an afterthought**

Other Concerns

- Scalability: Spawn Fast, Die Fast
- Expect the Unexpected
  - Unpredictable instance lifecycles
  - Duplication of messages
  - Timeouts

Minimize Developer Pain

- Docker (and Docker Compose) to the rescue!
- `devstack` app with configured containers for:
  - Fedora
  - SolrCloud
  - ElasticSearch (AWS!)
  - ElasticProxy
  - IIIF (node-express wrapper for our Lambda)
  - S3 (Minio)
  - ElastiCache (Redis)
  - SQS (ElasticMQ Wrapper)

Last But Not Least: People

- Thinking “cloud-first” *probably* involves a significant culture shift
- Don’t have a single “cloud guru”
- Try to get everyone to be The Expert at something
- Get everyone outside their comfort zone
  - Yes, even when it makes things take longer
- Different Modalities: Training, Sharing, Reading, Doing

Thank you!

Michael B. Klein
michael.klein@northwestern.edu

Brendan Quinn
Brendan-Quinn@northwestern.edu

David Schober
david.schober@northwestern.edu

Karen Shaw
karen.didrickson@northwestern.edu

Adam Arling
adam.arling@northwestern.edu

NUL GitHub Repo
https://github.com/nulib/

Forest Path with light shining through. Photo by Peter Heeling. Public Domain. From https://www.goodfreephotos.com/
Owlbear © 2019 Wizards of the Coast LLC. https://www.dndbeyond.com/monsters/owlbear