APACHE SLING & FRIENDS TECH MEETUP
2 - 4 SEPTEMBER 2019

Sling Content Distribution for the Cloud

Timothée Maret, Adobe

About the speaker

Timothée Maret

https://twitter.com/timaret

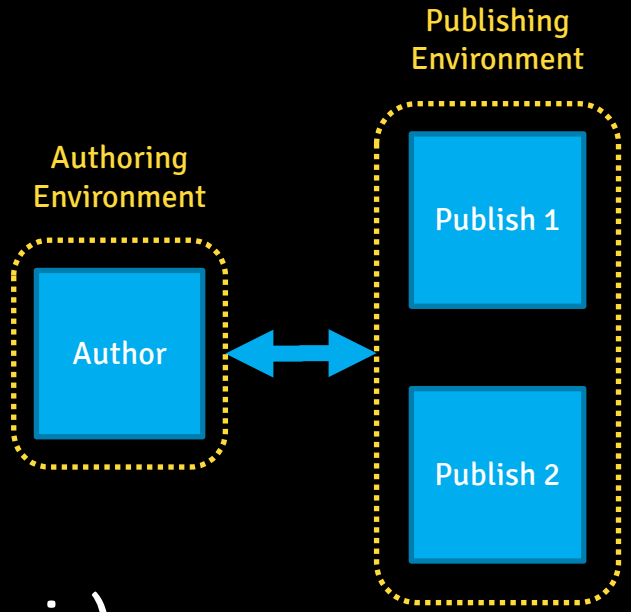
tmaret@apache.org

Sr. Software Developer, Tech Lead @ Adobe R&D

Apache Sling PMC member

Distribution Framework

- Central to AEM architecture
- Use cases
  - Publishing content
  - Syncing user and groups
  - Fetching User Generated Content (basic)

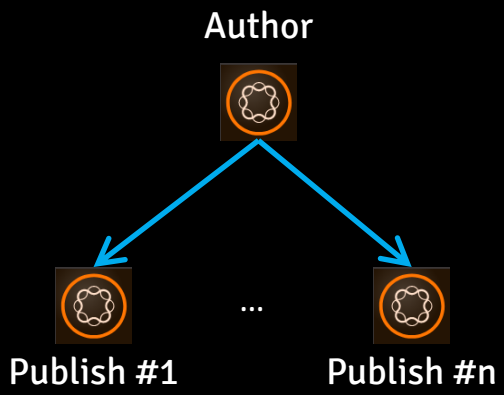


adaptTo()

Distribution Flows

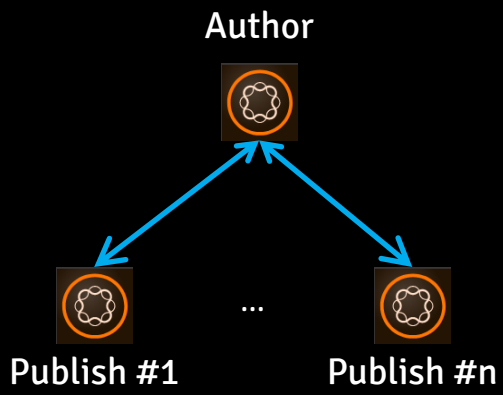
Forward

1 to n



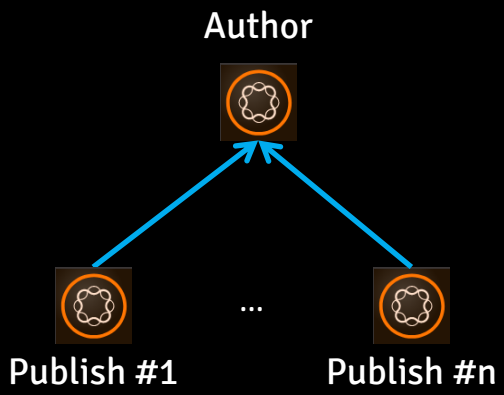
Sync

n to n



Reverse

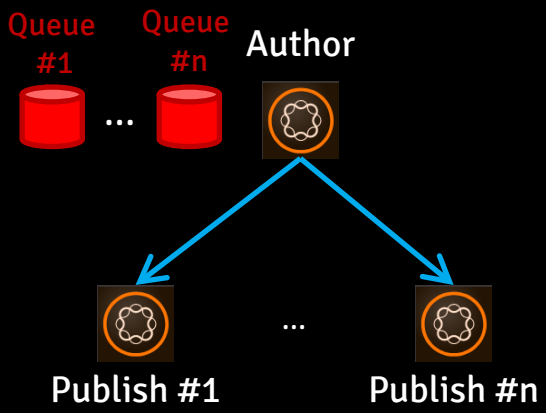
n to 1



Implementation on Sling Jobs (1/2)

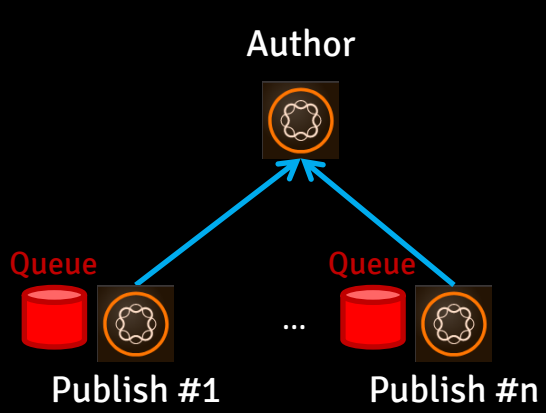
Forward

1 to n



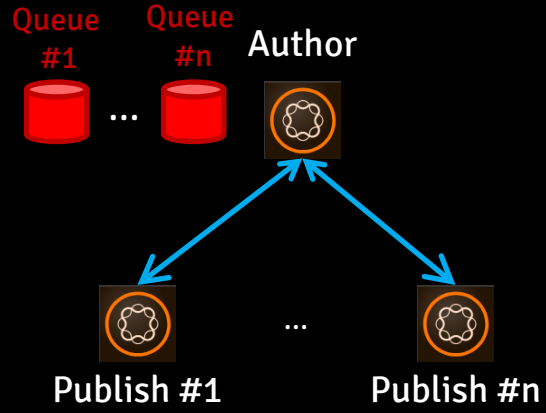
Reverse

n to 1



Sync

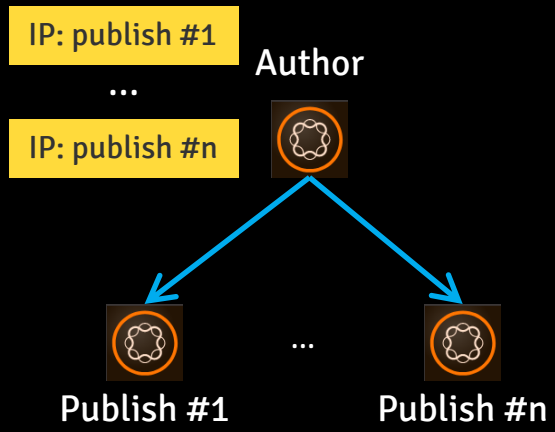
n to n



Implementation on Sling Jobs (2/2)

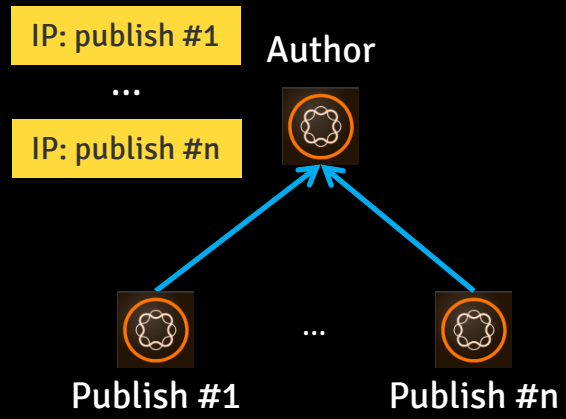
Forward

1 to n



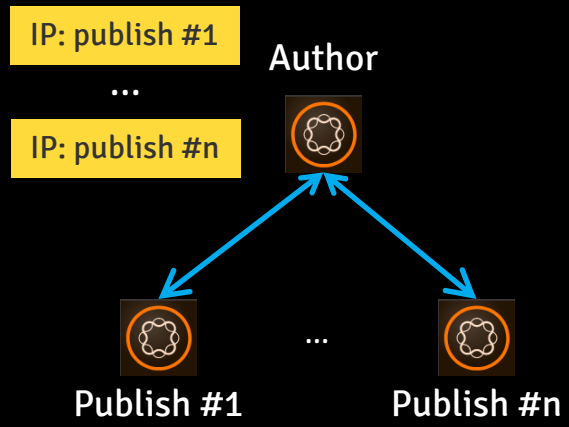
Reverse

n to 1

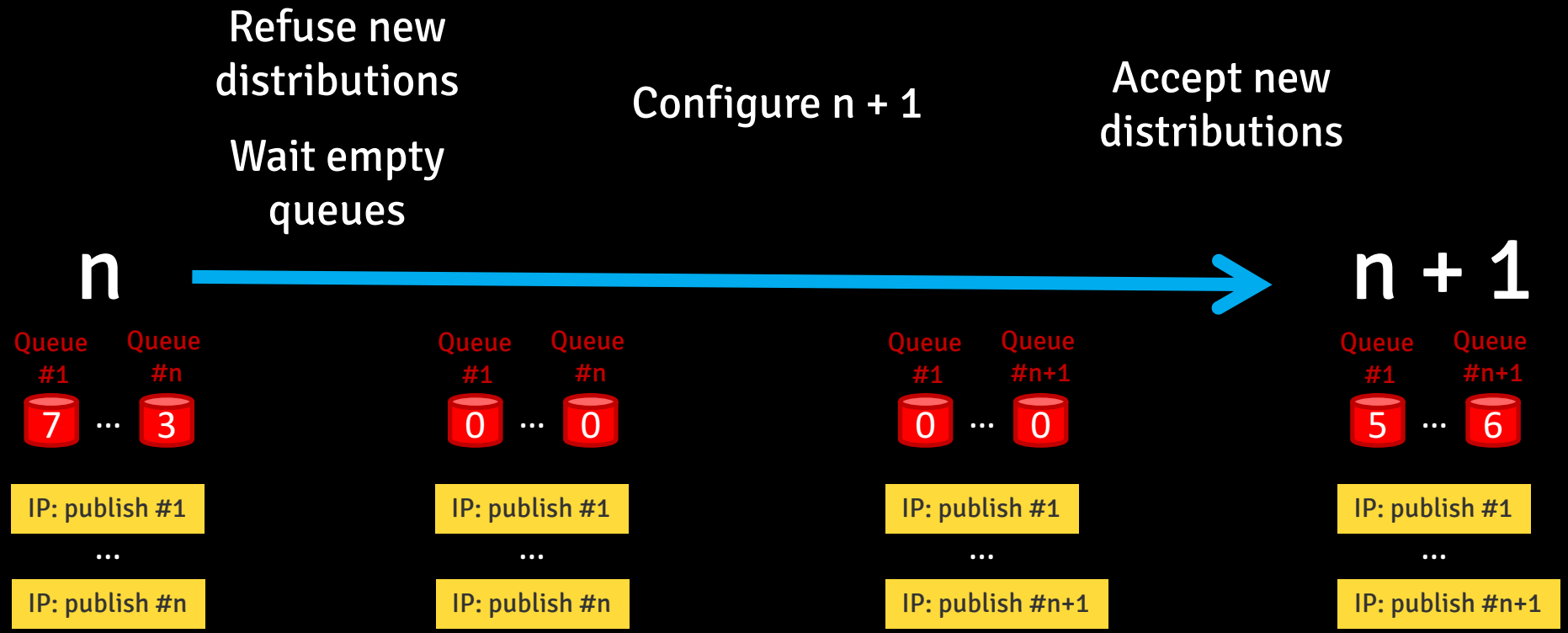


Sync

n to n



Scaling operations on Sling Jobs



Moving to the Cloud

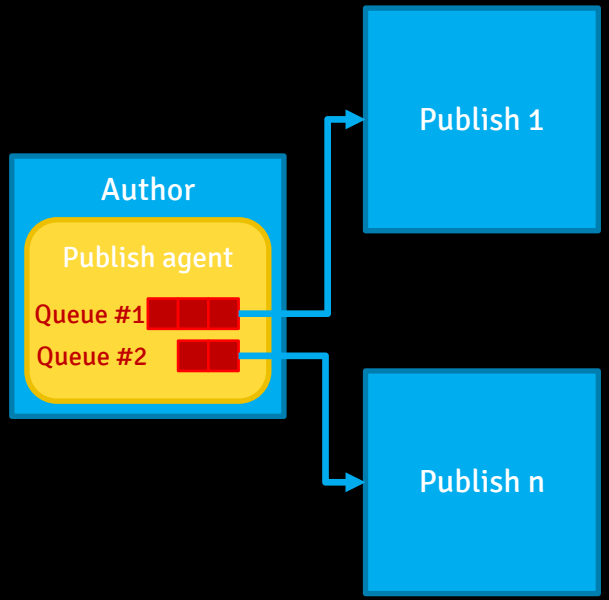
adaptTo()

New paradigm

- Requirements
  - Publish farm horizontal auto scaling (no point to point connections)
  - Reduce repository operations
  - API backward compatibility
- Maintaining queues outside of JCR is a sensible option

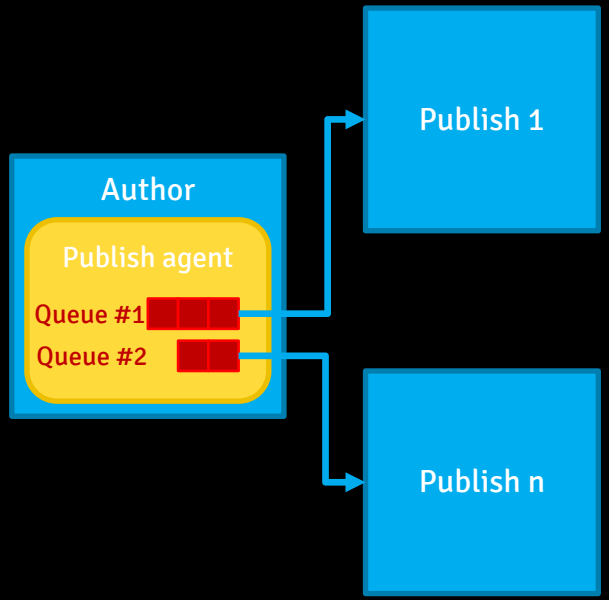
Distribution Model - On premises

- One queue per publish service
  - Events (queued, distributed)
  - Iterate queued items
  - Remove queued items randomly
- Published content eventually consistent



Distribution Model - Cloud

- One queue per publish service
  - Events (queued, distributed)
  - Iterate queued items
  - Remove ~~random~~ consecutive queued items from the head
- Published content eventually consistent



Journal based Forward Flow

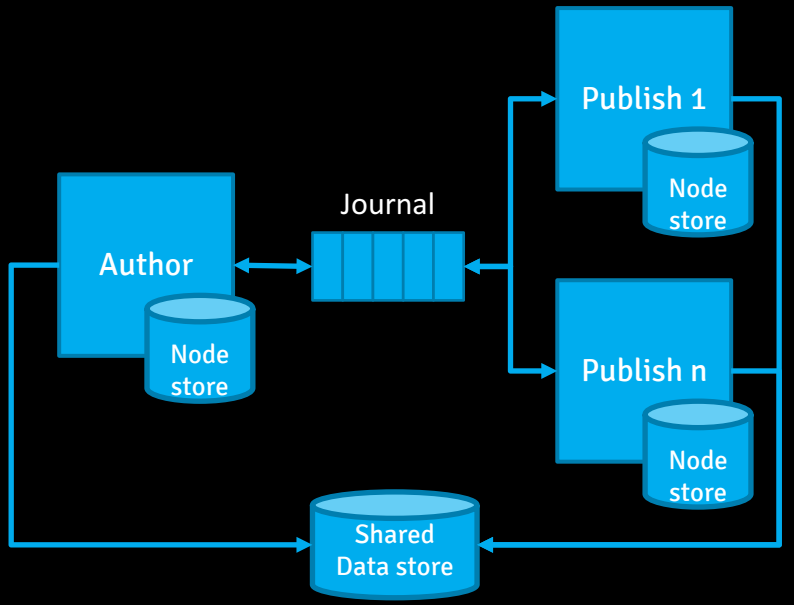
Journal

- Append-only
- Messages
  - Totally ordered
  - Positions orderable
  - Can be replayed from historical position
  - Time based retention
- Pub/Sub messaging model
- Subscriber manage positions

m_3	m_2	m_1	m_0	messages
27	9	2	0	position

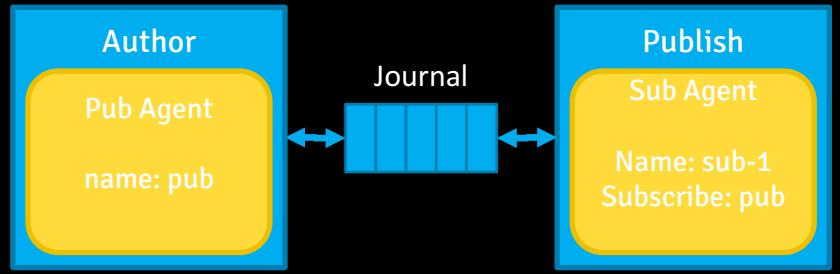
System view

- Author & Publish communicate via the Journal
- Shared Data Store
- Binary-less content packages



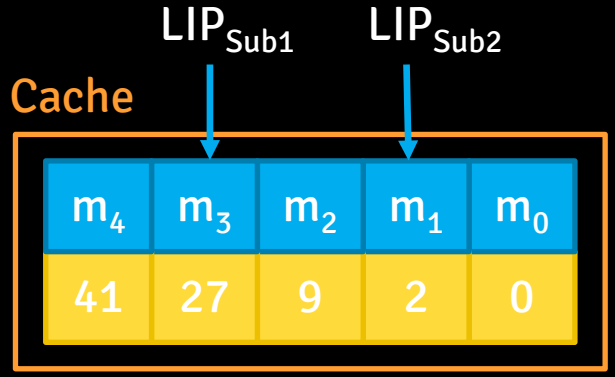
Pub and Sub agents

- Pub/Sub model
- Author and Publish are decoupled
- Sub can subscribe to 1 or many Pub agents



Pub agent functions

- Build & append content packages to a single topic in the Journal
- Discover Sub agents
- Compute queues from Sub agents Last Imported Position on demand
- Cache content package metadata on demand



$Queue_{Sub1}$: (m_4)

$Queue_{Sub2}$: (m_4, m_3, m_2)

Sub agent functions

- Import content packages exactly once
- Announce itself periodically
- Report import statuses
  - Implicitly (LIP in Ping message)
  - Explicitly (Status message)

Ping

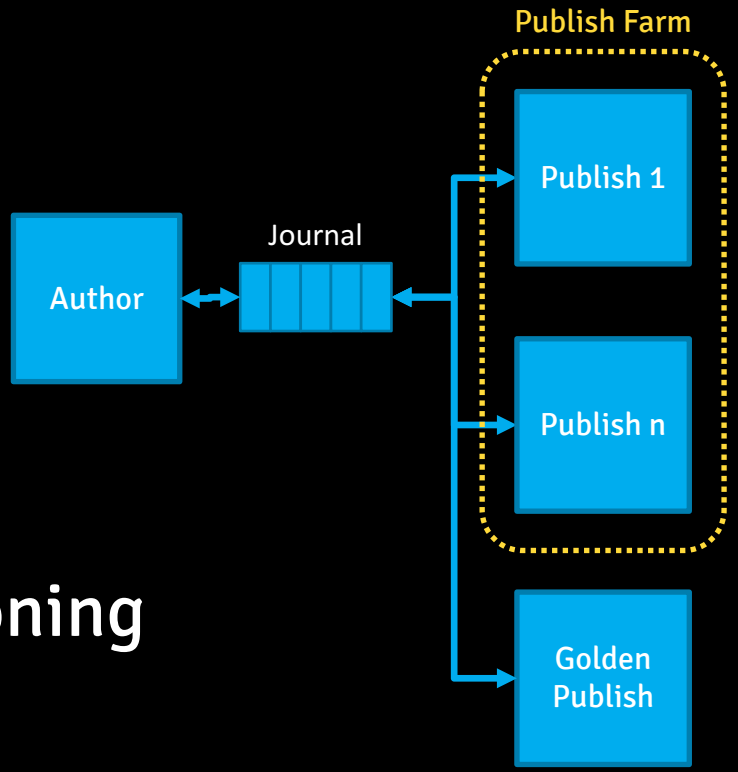
subSlingId: uuid
subName: Sub1
pubName: Pub
lip: 1492
retries: 0

Status

subSlingId: uuid
subName: Sub1
pubName: Pub
position: 123
Status: imported

Golden Publish

- Not exposed to traffic
- Source of truth
- Reference when scaling by cloning



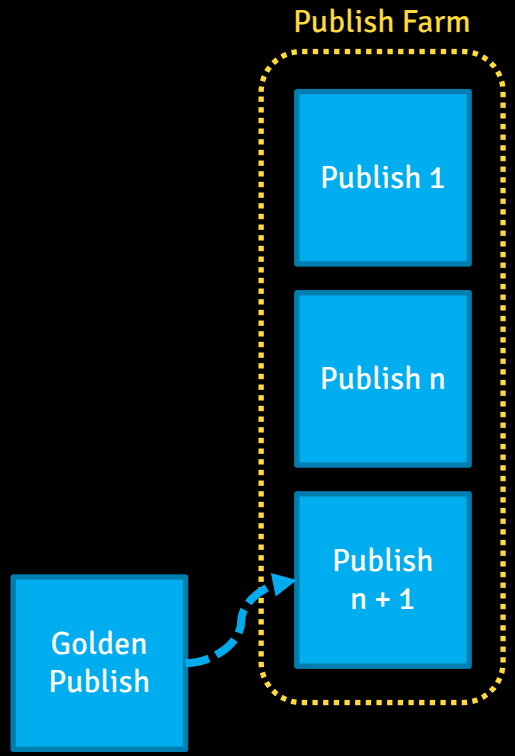
Scaling operations (1/2)

Deploy a recent* Golden Publish clone

* Younger than Journal retention policy (typ. 7 days)

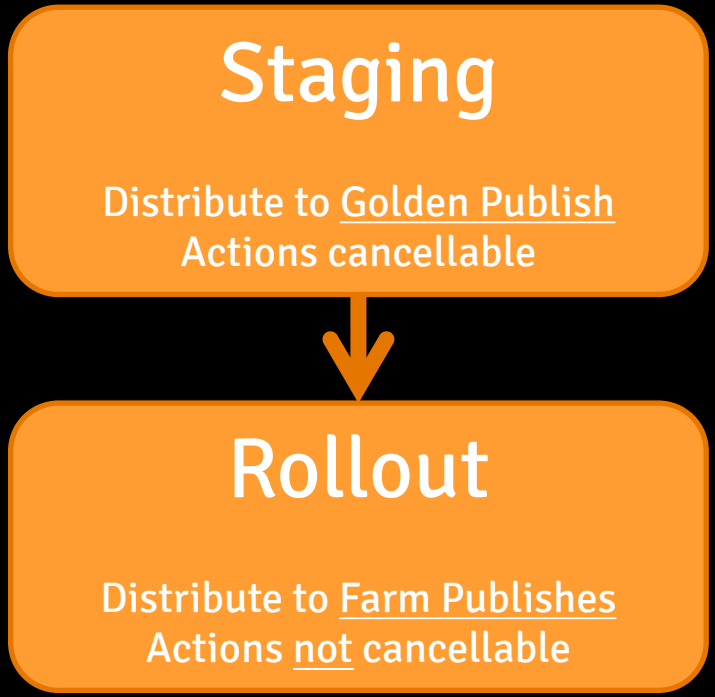
Scaling operations (2/2)

- Publish n + 1
  - Import from LIP carried in the cloned node store
  - Announce itself ready if nothing to import for > X sec
- Orchestration container
  - expose Publish n + 1 when ready
- Distribution actions accepted when scaling
- No coordination



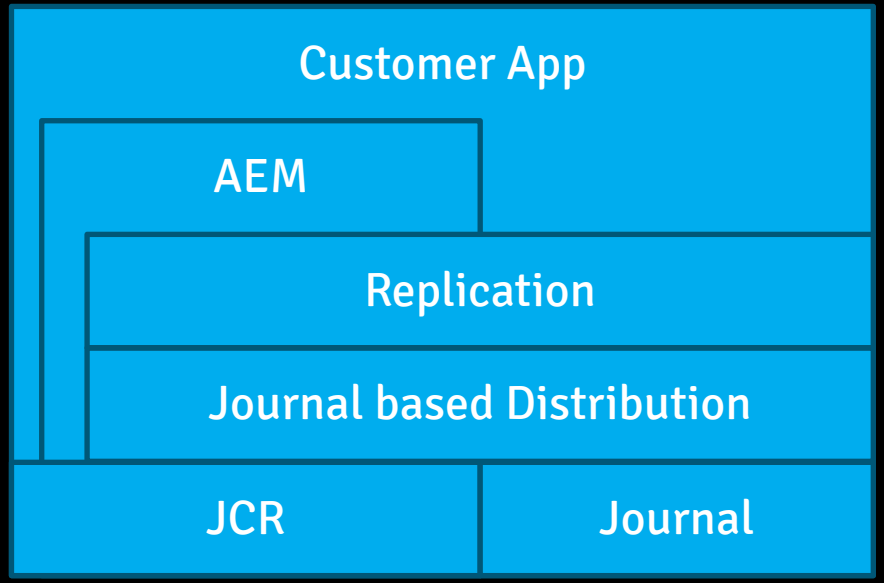
Staged distribution

- Distribution in two phases
- Maintain consistency upon
  - Error (e.g. blocked queue)
  - Clear/remove queue operations
- Error handling
  - Staging (manual, error queue)
  - Rollout (automated detection and replacement of Publish)



Integration with AEM

- AEM Publish Replication agent configured to use Journal Pub agent as transport
- One Replication Agent supports n Publish services
- Distribution UIs for clear/remove operations

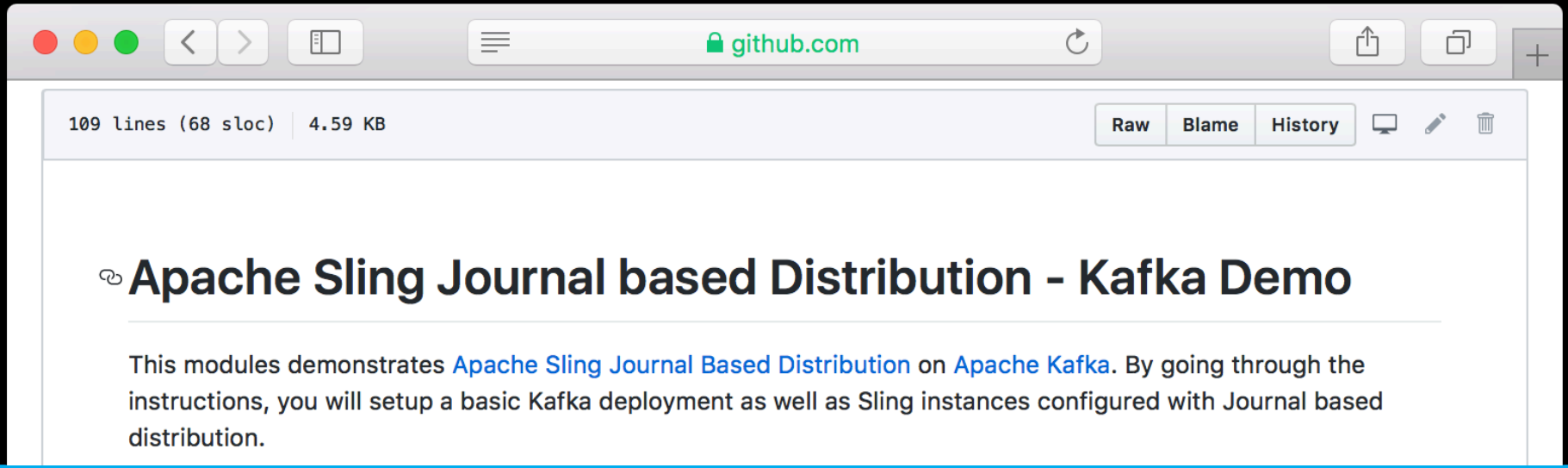


Design Portability

- Distribution Messaging API
- Journal requirements identified
- Kafka, RDBs, MongoDB, etc.

Check it out

https://github.com/tmaret/distribution.kafka.demo



Q & A

adaptTo()

References

- Journal based Distribution
  - https://github.com/apache/sling-org-apache-sling-distribution-journal
  - https://github.com/apache/sling-org-apache-sling-distribution-journal-kafka
- Content Distribution
  - https://sling.apache.org/documentation/bundles/distribution.html
  - https://sling.apache.org/documentation/bundles/content-distribution.html
- Demo
  - https://github.com/tmaret/distribution.kafka.demo