

Theme: Shanghai library FOLIO project

Time: August 17, 2021 07:00pm (EST) / August 18, 2021 07:00am (GMT+8)

Attendees:

Vincent Bateau (Enterprise Architect, EBSCO)

Gang Zhou (Project manager, Shanghai library)

Sha Jiang (Technical Director, Jiayu)

Lucy Liu (Product Owner, Folio China)

Notes:

1. Updates from Shanghai

Gang Zhou: SHL went live with a foreign collection (English, Russian, German, etc.) in one division yesterday.

2. Okapi clusters

Gang Zhou: We are moving the Folio server to other virtual machines. How do we expand the cluster?

Vince:

- At EBSCO, we run Okapi in the cluster mode. Texas A&M runs Okapi in the cluster mode, with as many as nine instances in the cluster, if I recall correctly.
- You can install/deploy multiple Okapis and configure them to work together in a cluster so that they will use the internal work of Okapi and specifically Vert.x underneath. I think they are using Hazelcast in this case. They will coordinate and keep each other up to date. If you register with one Okapi, the other Okapis will eventually catch up in the same cluster and have the same information. The point of this is to have some extra bandwidth and more requests you can handle because the request can be sent to any of the Okapis in the service. You will have extra capacity. There is redundancy in the system because if you have one single Okapi and that Okapi instance crashes or fails, then your entire system is broken. But if you have a cluster of three Okapis and one of the Okapis fails, you still have the other two that can keep running. Then you can rebuild or restart and put another instance into service. And that instance will automatically catch up with the other two and be fully available to replace the other two. That's assuming that you have enough clusters in quorum - three servers, if one goes down, two can restore the third one.

Gang Zhou: If we have three or more servers to deploy Okapi and the apps, can we use this solution?

Vince:

- Assuming you put three Okapis, each one on one server, that makes sense because you have redundancy. Then you need to configure them in a cluster. They can

essentially behave as one Okapi, especially if you put a load balancer in front of them and make it redirect the traffic/requests. The Okapis have the same information and equally serve.

- Assume you have multiple instances of apps. From the point of view of apps, the three Okapis are treated as one. You deploy the apps on one or more servers and register them into Okapis. You can register them individually as separate instances of the apps and let Okapi decide which one to send traffic to because Okapi essentially is a router and can choose which one to use. Or, you can put each app behind a load balancer and register the balancer with Okapi. So Okapi doesn't know if there are three instances of the same app. Okapi just knows the url/the destination of the load balancer that serves up each of the three apps. So if you put multiple instances in an app, you can either let Okapi decide which one to use, or let the load balancer decide which one to use and Okapi doesn't know. My personal preference is to use load balancers and remove the choice from Okapi. Then you will have the ability to add or remove the instances of apps without having to go through the process of registration in Okapi every time.

Gang Zhou: Which way is easy to expand if we will have more apps in the future?

Vince: The load balancer.

Gang Zhou: We deploy Okapis and apps on one server. Do you suggest that we separate them to different servers?

Vince:

- Not necessarily. You can configure them that way. You can deploy Okapis on one server and deploy apps on the other two servers. Or, you can distribute them across the three servers. It doesn't really matter. If the three servers are not equivalent in terms of CPU or memory, you might want to put certain things on certain servers. If they are all the same, my suggestion is to keep them the same so each server has both Okapi and a set of apps.
- Okapi doesn't know where the apps are in terms of the server. If you put a load balancer, all it knows is how to get to the load balancer. And the load balancer will redirect requests from Okapi. You need to consider your network in terms of latency - how quickly you can communicate between servers. But if they are on the same subnet, I don't think there's a significant amount.

3. How much of the load mod-inventory can handle? (Sha Jiang)

Vince: There were some tests done by the Performance Task Force. You need to consider the CPU and the memory you have in the module server, in Okapi server and also in the database server. Any one of them can become a bottleneck at some point.

Lucy: <https://wiki.folio.org/display/FOLIJET/Performance+Task+Force+Team>

4. [Folio Architectural Blueprint Strategic Changes](#)

Lucy: Can you give us an update of the architectural blueprint changes?

Vince: The new PC and TC might revisit the blueprint and the tech debt in a month. New items might be added to the list. Some of them are short-term, some long-term. They are things that have to be done.

ABI-002	Cross-tenancy support	One of the prerequisites to support the consortium. Will be discussed next year.
ABI-003	system/tenant level users	There is a discussion group from TC and a little bit of work to classify users according to types. There will be work to use these types to create institutional users.
ABI-004	Database connectivity	Might be some work in the near future. Not sure how much consent we will get on it.
ABI-005	AuthN/Z Refactoring	Open topic
ABI-019	Distributed Transactions	Work going on now. How to coordinate changes and data synchronization between different modules?
ABI-006/009	Refactor Okapi	Not sure when will happen, probably not soon
ABI-010	Multi-Tenancy	No progress. There's significant resistance to change this. May connect to ABI-03.
ABI-011	Search Engine	Almost done. Elasticsearch is available as a search engine with indexing and has the ability to synchronize data. It has not been made the default search interface for folio inventory yet. There is a prototype right now. The last thing to do is inventory itself needs to change its search interface to make use of the ES.
ABI-012	Camunda	Nothing
ABI-013	formalized Roles	Not discussed
ABI-014	GraphQL	Not high priority, not a must have in folio
ABI-015	Application packaging	There's a work group. Discussing, but slow. Might make progress next year.
ABI-016	SAML SSO strategy	Depends on if we can get a group to work on it
ABI-017	Codex	Not actively worked on at this time

ABI-018	Okapi/modules	Discussions and designs, but not much has been done yet
---------	---------------	---

Gang Zhou: Some of the items are important to us. Would like to know more updates in the future.

Lucy: We will add this to the October agenda.