

Theme: Shanghai library FOLIO project

Time: January 26, 2021 07:00pm (EST) / January 27, 2020 08:00am (GMT+8)

Attendees:

Vincent Bateau (Enterprise Architect, EBSCO)

Gang Zhou (Project manager, Shanghai library)

Sha Jiang (Technical Director, Jiata)

Lucy Liu (Product Owner, Folio China)

Notes:

1. Librarians can access Port 9130 without going through the front-end of FOLIO.

How can we add control to this?

Vince: By design access to Okapi APIs is required because Folio is a Single Page Application (SPA) and the front-end (Stripes) runs in the user's browser. Therefore there needs to be accessible from individual users' browsers wherever they may be. So it is not possible, for example, to place Okapi on a firewalled subnet and limit access to the hosted web server - as is done in traditional web applications. We should also discourage direct integration via Okapi by external systems - this is why we have Edge APIs. The coverage of Edge APIs is not complete and if some are integrating to internal Folio APIs via Okapi, it should be captured as requirements to create additional Edge APIs. In terms of restricting access to Okapi 9130 it is a challenging problem because of the need of SPAs. Folio does not offer a solution for this. My recommendation to this problem is to consider placing a Web Application Firewall (WAF) in front of Okapi. WAFs can be programmed with sophisticated rules that can attempt to infer whether requests are legitimately from Stripes in a browser.

2. Is there a circuit breaker in Okapi? If not, what alternative solutions would you recommend?

Vince: As a whole, Folio has not implemented this pattern. In its current form Folio is still struggling to deal with dependencies and distributed transactions. I think that circuit breakers will be part of the conversation when we get to discussing dependencies and distributed transactions solutions - which may be in the very near future. In the meantime, I do not have much of a solution to offer. The approach taken by many is to provide hosting in a High Availability (HA) mode, where redundancy can reduce, but not completely eliminate, the risk of failures.

3. There is information about database connection in Okapi discovery. What security methods would you recommend in the production environment?

Vince: It is a known issue that Okapi must maintain superuser credentials for database access. This is because it will be responsible for initiating database schema and user creation for a new tenant. It is possible to avoid Okapi storing these credentials, by providing them as runtime parameters where they are then only in memory.

Environment variables can be used on the system running Okapi. This is not ideal but removes the security issue from Okapi itself and moves it to the operating system level where it might be better protected through best practices (such as removing ssh access). This issue has been raised as a security concern, but no action has been taken so far by the core platform team. There also exists a proposal for a possible solution ([FOLIO-1935](#)), but that also has not been evaluated nor implemented.

4. Database Issues

- **inventory_storage.item - It takes 46 minutes to build index for 35,000,000 lines of items.**
- **Some search will trigger Recheck Cond: (permanentlocationid = '89951591-1adb-4f63-8ce5-98e3134808c2':::uuid)
The first step is to locate data blocks using Bitmap Index Scan. But there are still a lot of IO scans in the second step. So the whole search depends on the IO performance.**

Vince:

RMB is responsible for creating indexes according to what sends to a Json file. For a long time, performance problems in folio are addressed by creating specific indexes in that file. Somebody reports there are really bad performances in the circulation or inventory, so they create an index and a tuner index to try to make it work better. Then with the next release someone has tried adding other modules or changed the schema, or somebody has now got more data than before, and it doesn't work. So now they tweet the index again and it was constantly about tweaking and adjusting the index. The result is kind of a mess. And it's entirely possible that there are many indexes that are not useful, or fields that are being indexed don't matter anymore. I don't know if anybody has really gone in and done clean up. But I'm not sure anybody ever really removes them if they're useless. So this is the pattern that folio has done for many releases. We want elasticsearch to do searching for inventory because we reached the limit and can't keep turning these indexes.

None of the indexes have been tuned for 35 million items. And we know from past experience that when we target larger and larger datasets, we have to change the indexes. So I'm not surprised if there's issues with these indexes. I think it's fine if you would go in tune these indexes according to what you are seeing with your 35 million records. If you need to remove some, go ahead and try that. You can always put it back if you need to. But you're going to need to tune it. Since you're using elasticsearch, maybe you should just focus on changing that query path to go using elasticsearch rather than using the index that is built into the PostgreSQL.

Regarding the syntax of specific index creations, if you want to validate some changes or you want to discuss the things further, I would suggest you contact people (in this case inventory) who would have been responsible for these indexes in the first place.

Gang Zhou: In some cases, we will need some native indexes in the table?

Vince: Then I suggest that you engage with members of the communities who are working on these indexes in the past. I think basically members of the Core-platform and probably other ones who have worked on this is probably a good place to start to get their opinion on how to improve this one, if it can be improved, or if there are other approaches to query for barcodes.

Followup: Lucy approached people in the Core-platform team and was told that Julian Ladisch would be the right person. Lucy has introduced Gang Zhou to Julian and started the conversation on Slack. It was also suggested that we post directly to #development or #raml-module-builder if it's something others would benefit from.

5. Do we have other news to share?

Gang Zhou: We plan to deploy the circulation function on folio with a closed stack at Shanghai Library in March.