OTS Project Health Report

February 14, 2019

Introduction

Late in 2018 EBSCO sponsored an assessment of FOLIO by Open Tech Strategies (OTS). OTS was asked to answer one central question: “how ready is FOLIO for multi-tenancy hosting production?” They spoke to several members of the Core team and the FOLIO Community, downloaded code, stood up FOLIO instances, reviewed GitHub activity, and evaluated the code, processes and community activity to create their report.

The report is 36 pages including appendices.

OTS has experience with other open source projects and has approached this engagement as technologists as well as observers of open source communities. They applied both of these perspectives to their analysis and findings. FOLIO is a complex project both technically and socially. As a result, the OTS report required some validation, vetting and review so that its findings and recommendations could be put into context. This page represents the Tech Council's evaluation and eventual recommendations based on the OTS report.

The OTS Findings

The report is available here: https://drive.google.com/drive/folders/1WN757TCn0TsgzRC1LkimOIXwW1EelmHq

The executive summary asserts "For all that this document reads like a list of complaints and deficiencies we were impressed at where FOLIO is at this stage in its lifecycle... FOLIO is at a healthy place on the maturity curve.” OTS further states:

- The system is well engineered in many ways, but there are some areas of concern.
- While one could deploy FOLIO in a multi-tenant environment at this time, it would be difficult, expensive and risky to do so.
- In particular, EBSCO could deploy it with significant investment given their knowledge and “privileged access” to FOLIO. [Technical Council note: there is no privileged access to FOLIO. The code itself as well as the development process is the same for all participants.]
- At this point, it would be hard for new entities to justify the significant investment it would take to build a business around standing up and maintaining multi-tenant deployments for FOLIO libraries.
- OTS expects FOLIO to become easier to deploy and maintain in multi-tenant environments.
- Given its stage of development, FOLIO has issues that could be expected and is on par with other projects at similar stages.

- There were issues related to scalability, performance, customization and reliability that fell into a few categories:
  - Difficulty diagnosing problems
  - Performant development with JSON-based storage
  - Complexity and entanglement between front and back end that make it difficult to develop and deploy FOLIO
  - The way FOLIO handles modules creates a major hurdle for efficient customization by tenants
  - Inadequate documentation
- Functionally FOLIO is, at present, only able to meet the needs of a narrow set of libraries. The architecture enables further development but care must be taken to ensure the bar isn't too high for new developers to join so that added functionality can be realized.
- The quality of the Core is, overall, good.
- There were some potential security issues, and OTS recommends a full security audit.

Critical Recommendations:

- Streamline introductory developer documentation
- Provide better support and documentation for custom modules, especially private ones. Document how to add a new module to a FOLIO instance
- Dedicate staff time to seeing out and resolving issues related to the difficulty of setting up development instances and evaluation instances
- Improve diagnostic capabilities by implementing a dashboard
- Start designing and implementing the core changes needed to make full module independence per tenant the norm in FOLIO. This will be a complicated change touching many parts of the code; the sooner it is started the better

OTS made 29 recommendations, which they listed in Appendix C of their report. Each recommendation comes with a criticality/time horizon as well as the section of the report to which it relates.

Tech Council's Review and Commentary

Overall, it was an excellent, thoughtful and helpful report.

Our Process

We reviewed the report then held a meeting where OTS presented the report, and we were able to ask questions and discuss. We created a spreadsheet that listed all recommendations and then asked each Tech Council member to rank each recommendation with a timeframe in which we should recommend addressing the recommendation. The choices we had were:

- ASAP
- Q2-2019
- Q3-2019
- 2020
- 2021 (after 1st round of major deployments)
- Not important - whenever if ever
- Disagree with Recommendation
After each TC member ranked each recommendation we met several times to discuss our responses and agree on a group ranking, which we documented in that spreadsheet.

**Assessment of their recommendations**

The report's recommendations ranged in urgency, scope and specificity. Each recommendation was accompanied by a horizon (near, year, long) as well as a 'theme'. It was well done.

The TC did not initially agree on many of the recommendations - even some of the more concerning recommendations were met with responses ranging from "ASAP" to "Not important". However, after our discussions we have collectively settled on the following assessment of their recommendations. Gene rally, there were three key takeaways from the report:

1. There may be security issues, and we would benefit from a full external security audit. We request that the FOLIO Stakeholders fund this activity as soon as practical. We estimate the cost of this external audit to be in the range of $30,000 to $70,000.
2. There is a question as to how problematic the current method of using schema-level tenant separation at the database level combined with the difficulty in using separate databases per tenant per module is. This needs to be discussed further.
3. Documentation continues to be a challenge that may benefit from the creation of a new project role: Technical Writer

**Critical Recommendations:**

- Streamline introductory developer documentation: **TC: Not urgent but we call for the creation of a Technical Writer role to oversee documentation**
- Provide better support and documentation for custom modules, especially those developed outside of the community process. Document how to add a new module to a FOLIO instance **TC: Address in 2020**
- Dedicate staff time to seeking out and resolving issues related to the difficulty of setting up development instances and evaluation instances **TC: Evaluate in Q3 in conjunction with SysOps SIG**
- Improve diagnostic capabilities by implementing a dashboard **TC: Not urgent - evaluate in 2021 (after 1st round of major deployments)**
- Start designing and implementing the core changes needed to make full module independence per tenant the norm in FOLIO. This will be a complicated change touching many parts of the code; the sooner it is started the better. **TC: It's arguable that this is a gap, but should discuss in Q3**

Other selected recommendations

- Document how to pull FOLIO modules from different or multiple npm repositories **TC: 2021 (after 1st round of major deployments)**
- Pick an approach to improving thoroughness of test coverage **TC: Evaluation may have been done prior to current active efforts lead by Anton; we are using SonarCloud and set 80% coverage as definition of done.**
- Decentralize continuous integration and ease the burden of CI for developers who lack FOLIO insider access. **TC: We should plan tasks to address this in Q2-2019**
- Provide better support for standardize administrative interfaces for customization. **TC: Not important**
- Improve code quality by adding a linter or other automatic checkers to the continuous integration pipeline. **TC: Important to continue to look at the output of the linters and other automatic checks that we have in place - but we do have them in place. Ensure that all teams are reviewing SonarCloud. Anton will send a message to developers.**
- Explore supporting additional database backends. **TC: Something to consider as technologies and FOLIO mature but not urgent**

**Conclusion**

This exercise was valuable. Having external eyes review and critique the project is healthy. Thankfully the assessment shows that FOLIO is in good shape with opportunities to improve (and with the understanding that functionality at this point is limited; teams are working this).

Key takeaways:

- We feel that having a security audit done is an immediate need.
- The project's technical documentation has some inconsistencies and redundancies - to the point where it is a soft blocker for an organization that is new to the project and trying to engage in development or setup a multi-tenant environment. The TC's discussion around this area led to the idea of commissioning a Technical Writer for the project to help manage and streamline our documentation. We feel this role will be more necessary and beneficial over the next year (likely a contract role).
- Partially related to this, there is currently a lack of diagnostic capabilities that will prove daunting in a deployment.
- Lastly, we should step back and review some of the technical decisions/implementations related to multi-tenancy and see if we need to strengthen or add flexibility as a result of that analysis.

We will be adding appropriate actions to the JIRA backlog so that they can be prioritized by the POs and community members.

The FOLIO Technical Council thanks EBSCO for sponsoring the OTS engagement.