LStools: Critical Needs in Folio

Gary Branch, Cornell University Library

March 2019

With the accompanying spreadsheet, I have tried to outline what is required based on the current LStools functionality in order for Folio to be fully operational for day one. We need to be able to load many vendor files and in order to do that the files need modification prior to implementing Data Import in Folio. On an average week the approval print files along near 100 and without MARC scripting the time for a staff member to do this manually would be around 5 minutes per file. That’s one full day to do what can be done in an hour or two. Without an LStools app it would require the staff member to connect to the vendor’s FTP site, download the files to be loaded, use MarcEdit or some other editing program to make necessary changes to each file, then drag and drop them into the Folio Data Import file upload space. Usually we cannot combine files, even for the same vendor, as they are paid on various selector funds for each plan with that vendor. It would require staff training for those Acquisitions staff with the skills needed to work in MarcEdit. Folio’s Data Import is not planning to develop an FTP configuration in version 1. The advantage of LStools allows us to see the files on the vendor’s ftp site, download the file, run the script, and queue it for import. We need that kind of functionality in a Folio app or have a significant cost in staff time.

 While the number of files for electronic are not as many per week, often the scripts required to modify the MARC records are more complex. Using MarcEdit is possible but shifts all the work into Batch Processing and will require either additional staffing or limit the staff’s ability to do other work. Many of the staff are currently split between batch work and other LTS functions.

 A further cost in print processing comes from the fact that we current use the OCLC Search API to identify better copy and replace the vendor record. This allows students and staff to process these approval books without the need to research them in OCLC and merge if they identify better copy. For many of our vendors that can be up to a third or more of the records in a file. We also use the API to improve existing records in Voyager. We need that functionality in order to continue to update lower quality records with better metadata. We can wait a little while for this, but it needs to be available relatively soon after Folio implementation as without it we would have a further drag on production and impede discovery.

 I think it’s an absolute necessity to be a part of Folio that apps can make use of. Besides the downloading of vendor files we do MARC exports to OCLC, Coutts Oasis, and Rapid for InterLibrary borrowing, plus more to come. For us not to have this functionality that Folio apps can call upon seems short sighted and moreover another waste of staff time to do this manually. On a given day there are at least 3 data exports requiring FTP. There is also the issue with uploading purchase orders to vendors which currently is scripted and there can be 4-7 scheduled connections and deliveries made in a couple minutes. It would also be useful to download EDIFACT invoices, but that is less a priority.

 This leads to perhaps one of the more important functions of LStools that needs to be supported somewhere in Folio. The ability to schedule both import and export jobs to run without staff intervention. Every day there are 6-10 scheduled jobs that run that require little staff time. It only requires a staff member to be sure they did run. The ability to query the data warehouse to identify erroneous data, use Data Export to pull that data, a script to fix the errors, and then Data Import to reload the corrected file. Some of this will need to be scheduled while others will be required to run live manually during the day. Data correction scripts while not a day one necessity need to be in the queue for as soon as reasonably able. It should continue the existing philosophy of fixing what is identifiable and correctable via automation and reporting out what is not safe to automatically correct.

 Further there needs to be interoperability between Data Import and LStools, LStools and Data Export, and wherever FTP support is placed and the apps. Whether this is in the planned Workflows app or elsewhere is a design decision.

 I estimate that if LStools is not available to use on day one, that the major functions alone will require the work of at least 2.5-3 level D FTE with Batch Processing training. This also means that new vendors will need to be manually processed with the remaining Batch staff greatly diminishing the unit’s output and shifting or delaying the other job responsibilities of that staff. I would advise strongly that as much as possible LStools functionality as complete as possible is a day one necessity. From gauging the speed of development of Data Import, then Data Export will be either rushed or not as complete as needed, if we move forward on July 1, 2020.