LaSSSI meeting June 7, 2006

Attendees:

CCLA: Jeannie Dixon, Gary Johnson
CDL: Patti Martin
CUNY: Pat Young
FCLA: Jim Corey
MnSCU/PALS: Al Rykhus, Becky Bell, Mike Barnett, Maryann Greenwald, Deb Domek
Michiana Academic Library Consortium: Tom Hanstra
ODIN: Tony Stukel
SDLN: Sean Crooks, Patrick Weber
SUNY: Maureen Zajkowski, Carey Hatch
USMAI: Jamie Bush, David Kennedy, Jean Phillips
VCCS: Gene Damon

Ex Libris: Omri Gerson, Carmit Marcus, Susan Stearns, Revital Marck, Dan Trajman, Matti Shem Tov, Einat Zviran

Purpose of the meeting: talk about solutions to the key issues that are impacting shared systems (primary focus on Aleph) and discuss how we can work together to resolved them.

The geometric impact of certain elements of the Ex Libris software. The scope and repetitiveness impact inherent to the LaSSSI architectural models. Carey Hatch expounded on the issue at SUNY. They are comparable to FCLA; the only thing they share is the servers. The repetitiveness of the work; they run 56 upgrades. Partly due to the architecture of the system, but partly due to the upgrade process. They have automated as much as they can. At USMAI they are one institutional model, but they hear it from other LaSSSI members so have concern should they change models in the future. USMAI and CUNY are the only ones with a single ADM. It is safe to say that this issue is a function related to multi-ADM installation. Matti noted that there is a difference in multi-installation and multi-ADM. In respect to upgrade and service pack, once implemented all processes run once. In multi-ADM there would be multiple batches to run; but the software is upgraded once for all, regardless of how many. At SUNY there are multiple installations, so multiple processes.

What is the issue that is causing the problem with repetitiveness? Is it multi-ADM? In conjunction with multiple u-trees? What causes this? Once that is known, then we could identify who could work with Ex Libris to resolve. What is the impact of the different architecture?

Concern expressed by several that their configuration may not have been the best one as no upfront analysis as to whether this was the best choice. There may not be sufficient understanding of the impact of the architecture decision. We need to be clear about what impact each choice has.

Carmit presented PowerPoint for discussion on different architectures currently in use.

SHARED MODELS:
Single bib with multiple ADM – CCLA and PALNI. Only 2 NA consortia with this architecture.

Multi bib with union view and multi ADM – Mankato, ODIN, South Dakota, 5 Colleges.

Separate catalog with own ADM – SUNY and FCLA

A single bib with a single ADM is not a shared model – CUNY and USMAI (from the Ex Libris perspective; the institution may be a shared system but the architecture does not reflect that)

Single bib/ single ADM – USMAI and Novanet – 'flat circulation'. Allows resource sharing easily. The built-in circulation between sub-libraries was not enough for consortia. USMAI is using it for inter-campus borrowing and added the HOLD request function on top of it (where patron walks in and requests an item). In acquisitions and budget control there was a problem and Ex Libris had to introduce new functionality along with functionality written by USMAI to get institutional autonomy.

This is a very simple model.

Single bib title with multi-ADM libraries – CCLA, Palni, SBTS. Had to use ILL until v. 18 for resource sharing/consortia circulation. In v. 18 have ILL and PDQ. There is an agreement about unified catalog and separate institutional control. Consortia office can give control of the tab files, parameters, etc. to the institutions. Single u-tree.

Omri – tension in 3 elements: catalog; acquisitions/administrative elements; circulation/resource sharing. Each model takes these into account. As far as the history, Ex Libris comes along with solutions in all 3 tracks. The multiple u-tree is a preliminary solution to try to give the ADM autonomy in configuration. In v. 16 have a new approach that is called configuration vs. software approach. Took the elements that should be multiplied per ADM and multiplied only them per a single u-tree. The natural structure would lean towards what we see on the board (?single bib/multi-ADM?)

At implementation, sites generally focused on functionality not maintenance/operations. Now want to /need to optimize the configuration for maintenance/operations. We need a balance of functionality and maintenance.

In terms of TCO "this" is the cheapest. One u-tree, one software, one service pack, one upgrade and everything configured in a single u-tree – this model would be best for TCO in the long run. This model did not provide, at that time, the range of functionality that was needed. (esp. in circulation and autonomy for catalogers). From a patron perspective we give a union-view, give autonomy for ADM. And we need this PDQ for circulation/resource sharing.

In v. 18 can cross borders between ADM with ILL and PDQ. Re: TCO, look at 2 parameters-batch environment and ADM parameters.

Next model is very similar – different at the bib. library level. Mankato, Five Colleges, VCCS. Catalogers maintain autonomy. Omri – this would be a preferred model (both models; this even more). The sharing solutions in the bib file – have autonomy on the bib record but have software solutions to share the bib instead of configurations. The multi-alephe and multi-installation of configuration solutions for problems. Going down the road developing software solutions on a single configuration, single system to develop software solution for sharing and non-sharing. To balance this tension.

This is a model sep. bib libraries and sep. ADM libraries per institution. Combined union view of the catalog – only way to do is a union catalog -- SUNY, FCLA model.

The union catalog is a physical library with a union view. The UV is there to de-duplicate the bib records.

One of the issues with this model is that you have full autonomy so you can't have different systems or versions of aleph. No system is aware of the existence of the other system and its status. Hard to know how to implement automatic changes- because they might be so different. In the other model at least there is one system, with multiple "stuff"; this model is the only one that allows one system to be up and one down, etc. If you want less TCO there is some dependency in the single bib/multi-adm model. If you have to take the DB down for maintenance all are down. Easier to do a soft STP in the separate model. In the long run gives you some complications.

Single server for multiple aleph instances with multiple u-trees. No one at the meeting has this configuration. Each institution has its own u-tree. There is one a-tree. This is model for service providers. (an IT consortia – no relationships – an asp solution) Only supplying services. Ex Libris does this in the managed services model. Not a central office concept.

In v. 16 no longer support multi-alephe. Mankato and CCLA were the 2 with that.

Next model is the CUNY model – a combination of 2 parts. One bib library, multiple bib records with a union view. Allows for independence of the catalogers. But on the ADM level as if it were a single institution. Can't do a lot at the institutional level. Omri -this is a natural to go to a multi-ADM model. Since the adm records are managed separately, can't do any resource sharing.

Last model is Notre Dame model – multi institution. But different installations of aleph. From TCO point of view have even more burden than others. Have one alephe; but have to do table changes multiple times. Very similar to asp model.

CDL does not have administrative information so not represented in these models. The issue is that the bibliographic library is very, very huge. Until now the issues are how the institutions are reflected in the architecture and what functionality is enabled. Everything is single and therefore very simple. But the size matters.

Many of the issues associated with scale have to do with indexing and discovery tools. Ex Libris doesn't have a problem with managing 10 million or 20 million items; that's not an issue; the issue is connected with the OPAC; exposing things to the patron.

From Ex Libris point of view if we can narrow down the configuration to 1 or 2 and then focus the resources on improving the lives of those who use those configurations, is this an option? Longer term view (not one year) – choose different configuration and how to move to that – what is the impact on hardware; what is the impact on making the decision. Is this an option?

Matti suggested that we go through the process and select 2 or 3 models (extremes) and as a group explore what does it mean. Let's assume we pick 2; we develop whatever is needed to automate the things you had problems with; then pick one of the thinkings and with the brain trust around table explore what it means to move to that – what are the implications; impact on cataloging; impact on the institution; impact on your staff; on the library. How do we assure it works (and who is the 'we')?

Concern about how authority files are used. Not addressed in the models discussed. Carmit noted that there are a lot of variations – in the patron file, the vendor file, etc.

Consensus that reducing the number of configurations would help everyone. As a group let's pick one or two and argue about why and why not. Flexibility of the product for the user; for the patron; there is a trade off. Identify those issues and how to approach resolution. Establish what are the criteria to make the decision. And then Ex Libris will take this homework and say what the advantages are. This would lessen this automating stuff; these are the dependencies we would break. Next step would be researching those dependencies that make sense for us to continue. No one will be forced to move from one model to another. We need to agree to develop and enhance a specific configuration. Those that are on that configuration would test it and see that it does work and have confidence in going that direction.

Omri noted that the single bib library/multi ADM is the preferred way; stuff has been developed to support that. Minnesota, ODIN, SDLN, CCLA, VCCS, 5 Colleges, Palni, SBTS are the institutions with this configuration. This is the majority of those present. This is a core group. Two big criteria for moving will be functionality of essential services ("can we do the things we like in the new model?") and that those on the model will be satisfied with the TCO (functionality comes first).

Ex Libris also wants to know "why not" a particular model. If we identify why not, and come back to Ex Libris and say this is critical, then Ex Libris will come back with information on how difficult or easy it would be to make that work in the suggested model. In some current architecture our libraries have autonomy and consortia office doesn't have to be involved in a lot of things for them like privileges. In v17 we don't have the capability to prevent people with password-creation authority to not assign a permission that is not wanted. These are the 'why not' that Ex Libris needs to hear about. (the loss of multiple alephe creates this).

What do you really like about your environment and what are the things you really have to have?

Novanet is configured like USMAI because they need to have patron placed holds. They will be encouraged to consider switching models.

Every group should go home, talk to staff, create a matrix listing what is good, what is killing them to help get a handle on the issues. It should be basic: from the Oracle level, support level, from the sys admin support, the alephe, job list, batch processing, functionality. Need a conversation at the technical level to get a handle on the different environments. It is so huge, maybe pick out a few – table maintenance, oracle, permissions.

(Caution with using logical synonyms instead of path_convert. ) Don't ask the question "can I work with….", but explain what the problem is and how can it be solved?

How can we work with each individual and try to figure out the issues? What is needed is a gap analysis to identify what is missing from the preferred model and the other models.

Ex Libris will propose a preferred configuration. Each institution will review it and perform the gap analysis (with help from each other and Ex Libris). At the next meeting have an understanding and discussion of the gap and prioritize within each gap what are the issues that are must have and review it and see how Ex Libris can improve the preferred configuration to accommodate the multi-ADM. There still need to be enhancements to the preferred configuration. Define the preferred and the benefits as of today and define future development for the preferred model. Also do the gap analysis; what would it take for X institution to move to the preferred model.

There is still a difference of magnitude/scale within each configuration.

There is not one model, but there are flavors of models. Ex Libris has to have one common bond; it is one system with connections and maintenance. Can have multi batch, but running on one database. One system – one database is the key. FCLA noted that if the functionality is present the configuration is open to discussion. (SUNY agreed)

Ex Libris will come up with a preferred configuration with description and details. By August 15 the document will be circulated. Around Sept. 15, after discussion and clarification back and forth, have a day or 2 day Boston office get together with the technical staff to identify the key issues or propose some compromise. When the document is received, if there is something not clear should contact Carmit as soon as possible so the meeting won't be spent doing clarification. Should do a conference call to discuss clarification. In mid-Sept. do an Interwise with the Boston office meeting in October. Dan and Carey will work on the particulars for these meetings. Each institution should draw up our configuration and identify the advantages we thought we were getting from this configuration (and the disadvantages) so that everyone comes with documentation.

Question from SUNY: How does the shared bib file work (with some autonomy for each institution)? How does shared holdings (xxx60) work?

It would be good that one side of the group most familiar with the configuration share their knowledge of that configuration with the rest of the group.

Will the preferred shared system be the one that works best with Primo? Because of the decoupling of the system, there is no impact of the architecture configuration.

Is there a possibility of adding a shared system as a development partner for Primo? Could be a charter member, but too late to be a development partner. Concern was expressed that if no shared system involved, Primo might end up not working for shared systems. Ex Libris should know in 4-6 weeks if the European consortia will be a development partner. If not, then it might be possible to add a US shared system.

In our response to the preferred architecture document we have an obligation to state the reasons why we chose the architecture we did. In many cases what was done was because that was the way it was done in our systems at the time.

For agenda items 2 (batch processes), 3 (tables management), 4 (service packs/new versions), 6 (effective management of ongoing processes and cleanup scripts/start & stop scripts) Ex Libris is fully aware of and are either working on or have thoughts on these. 2 and 3 are big issues and solutions will be forthcoming, probably in incremental ways. Service Packs (SP) and Cleanup are smaller issues but have big impact.

Shared system testing information – testing is the role of the company, not the customer. The test plan needs to be transparent to the customers and goes a long way to building trust with the customers. What is the test bed; what is the test plan?

Approach with SP starting with v. 16--- SP include all changes to the version so that no customer has a version with specific fixes that Ex Libris cannot control; Ex Libris knows exactly what a customer has and doesn't have. The downside to this – we all want it as stable as possible and on the other hand we want every bit of functionality now – in v. 16 (in 2 years), 1600 changes – 25% of them are development (responding to the customer base needs). In terms of SP Ex Libris started out with a procedure with 4 cycles – 1) fixing – unit testing and in some cases (Ex Libris doesn't have test bed of configuration of every customer) testing done on server of customer with that configuration; do unit testing ; 2) integrators – every code change does not go by the programmer, but to a team of 3-4 senior programmers who have to verify each change and integrate into the software. After SP closed take 1 week to do sanity testing and unit testing (which is manual).

Ex Libris does regression testing of system 1 week before SP is packaged.

Ex Libris is moving to an approach of assurance instead of control.

The role of integrators is not to test fix but to test software. Ex Libris requires a "double signature" – programmer cannot put changes into software without an integrator. The Integrator is not testing the fix but how it works in the software.

The question is not the number of faults, but what they are. Ex Libris puts in an internal document to note in certain places 2 integrators are required because of the complexity of the software.

When it comes to a new version testing goes in a different direction – do extensive testing with 3000 scenarios.

Ex Libris will provide their 'test plan' (big Excel spreadsheet) to LaSSSI. There might be areas of duplication or areas that are missing. LaSSSI will give feedback on what might be missing in the test plan.

If a preferred model is established it will be established as test bed too.

Ex Libris currently tests on 1 bib/ 3 ADM – where 2 are shared and one not shared.

Ex Libris does scalability and performance testing and functional testing. Functional testing done on data of about 200,000 records. The scalability/performance testing done on about 10 million records.

If SP implementation notes could be grouped by function/module it would be helpful in working with it. The approach to the Aleph SP is that all fixes are optional to implement; but you have to read the entire document to determine if you need to implement. Matti – "you should be able to install a SP and without doing anything else, the software should not work worse than before. If this is not the case, let Ex Libris know." 2 policies of SP: you need to activate fixes and they won't do any harm if you don't; and Ex Libris doesn't touch the customer side data and parameters.

Carmit issued a call asking for good test data (especially from the common architectural model); want a lot of data; will scrub patron data; (version 16 and up). If anyone willing to share, contact Carmit to arrange to send a copy.

It would be helpful if someone in customer support would review the Implementation Notes before they are issued in order to clarify what is presented. It isn't always clear what is meant. Susan Stearns will follow up with this.

Oracle issues: concern with compound keys, lack of normalized data which effect ability to use SQL, etc. Sizing of table spaces, which module implemented, makes sizing difficult. Sizing and monitoring of space an issue. In the history of the system, Ex Libris made decisions which might not be the right decision today. Big question everyone is facing – to fix (especially the key structure) would require an upgrade. The cost will be high for some of the fixes.

In some cases a solution is known, but it is not known if the customer will be willing to bear the cost to implement. (do an upgrade, change local scripts, etc.).

Next step – get an Interwise/conference call scheduled with DBAs, etc. to discuss and understand the trade-offs are of whatever options are presented. Is the cost of making the change greater

than the benefit? Is this a DBA or a business discussion? Issues documented by NAAUG Ad hoc reporting group (4 years ago) and by this group (Oracle issues). Jeannie will distribute the NAAUG Ad Hoc reporting group to LaSSSI. CCLA will pull together a webinar of the LaSSSI group to discuss at the technical level.

Ex Libris noted that they are stable on Oracle and don't see a reason for moving from it.

Carey will schedule a conference call in a week or 2 for a LaSSSI member debriefing.

Meeting adjourned at 11:50.

ACTION ITEMS:

1. By August 15 Ex Libris will distribute a document on the preferred configuration with description and details. By mid-September an Interwise session will be held for discussion and clarification of the document. If something is not clear in the document, Carmit should be contacted right away and not wait for the Interwise session. In October a one or two day session in the Boston office will be held to discuss the technical issues of the configuration. Dan and Carey will work on the particulars for the meetings.

2. Ex Libris will provide their 'test plan' (big Excel spreadsheet) to LaSSSI. There might be areas of duplication or areas that are missing. LaSSSI will give feedback on what might be missing in the test plan.

3. Carmit issued a call asking for good test data (especially from the common architectural model); want a lot of data; will scrub patron data; (version 16 and up). If anyone willing to share, contact Carmit to arrange to send a copy.

4. It would be helpful if someone in customer support would review the Implementation Notes before they are issued in order to clarify what is presented. It isn't always clear what is meant. Susan Stearns will follow up with this.

5. Schedule an Interwise/conference call scheduled with DBAs, etc. to discuss and understand the trade-offs are of whatever options are presented. CCLA will pull together a webinar of the LaSSSI group to discuss at the technical level.

6. Jeannie will distribute the NAAUG Ad Hoc reporting group to LaSSSI.

7. Carey will schedule a conference call in a week or 2 for a LaSSSI member debriefing.