

SFX Version 4 Collaborative Testing Report

Introduction

The SFX Version 4 release process involves two phases of external testing, collaborative testing on an early product version and beta testing on a version closer to release, in addition to Ex Libris' internal Quality Assurance process. All processes serve the purpose of detecting possible problems in test and production environments and to deliver a high quality release version to customers. This report summarizes the arrangements and findings from the collaborative testing in March 2010.

Preliminary

In January 2010, Mark Dehmlow, SFX Product Working Group Coordinator for IGeLU and Maribeth Manoff, SFX Product Working Group Chair for ELUNA issued a request on the SFX Discussion List for participants in SFX version 4 collaborative testing. The goal was to find four volunteers from the SFX customer base who had a thorough knowledge of the SFX v.3 Admin at both the command line and web interfaces, and who collectively represented different types of SFX installations (single and consortia). Two members from IGeLU and two from ELUNA were sought. The needed composition of the group was one person to test the upgrade process, one person to test SFX workflows, and two people to test the SFX v.4 installation.

The participants

Four people were selected to take part in the collaborative testing:

- Jessica Hartwigsen from California State University (CalState)
- Inga Overkamp from Max Planck Gesellschaft (MPG)
- Holly Thomason from Stanford University
- Bas Vat from Leiden University (Leiden)

The test team represented a good mix from various disciplines, and more importantly, from different types of SFX installations and local implementations.

Reporting

In this report, we will try to elaborate on the following aspects of the collaborative testing:

April 15, 2010

1. The quality of the release.
2. The quality of the testing being performed by Ex Libris, including the need for new or changed test scripts.
3. The utility of the collaborative testing process, including answering the question as to whether it should be repeated for the next product release.
4. Provide suggestions for Ex Libris as to how future testing can be improved.
5. Provide suggestions for the user groups and Ex Libris about how to make future collaborative testing more effective.

Each of the four participants tested different processes and parts of the system. We decided to report back in this single document, which also includes our individual impressions. Our report focuses on the testing procedure, and it only gives a short description of the new functionalities of SFX v.4. More information about the release is available from Ex Libris.

Brief description of SFX Version 4

The text below is taken from Ex Libris' document "What_is_New_in_SFX_4_-_Highlights.pdf." Additional information on the contents of the release can be found in that document, and it will be supplied in other forms.

SFX v.4 includes the following key features:

- A database redesign to enable leaner, faster SFX KnowledgeBase update processes
- Weekly KnowledgeBase updates, supplemented by modifications between routine updates as necessary ("quick injections")
- New look and feel of the SFX Admin interface and functional enhancements, all resulting from customer feedback, including:
 - Easier navigation that is more workflow oriented
 - New KBManager functions, such as adding an object via the user interface and setting institution-specific thresholds
- Enhanced e-book support with the addition of author names

In SFX v.4, the database has been changed to support much leaner and faster KnowledgeBase update processes. SFX v.4 clearly separates local and global fields. Previously, in SFX v.3, all data—local and global—was stored in every local instance, as well as in the global environment (sfxglb3) within one installation, so it was therefore replicated several times. In SFX v.4, global data is held only in the global instance in each installation, and the local instances point to the data instead of storing the data themselves.

A major change was made in the data model of SFX and the SFX Admin. This provides a new structure for further developments, but it also underscores the need for thorough testing.

Testing before the collaborative testing

Before the collaborative testing started, extensive testing by Ex Libris had been done. This included massive comparison tests and thorough testing of the KBManager by the Ex Libris KB team. At the start of testing, SFX release v.4 was already in use by the KB team.

Preparations

A month before the collaborative testing, Christine Stohn (SFX Product Manager) told each of us what our individual focus would be, and she asked us to bring test scenarios, as well as lists to upload with DataLoader. Jessica was chosen to test the upgrade process on the CalState consortia environment and to check afterwards that everything worked as expected. Inga was assigned the task of starting with an upgraded MPG environment, running tests to see if everything worked as expected, and checking her most commonly used workflows. Holly and Bas were asked to work on a new SFX v.4 environment, testing their most commonly used workflows and focusing on the new SFX Admin interface.

Two participating institutions (CalState and MPG) had been asked to share a backup of their SFX installations to ensure that parts of the test could be executed in a localized environment. The data was loaded by Ex Libris staff into two SFX installations before the testing started.

The concrete preparation phase started two weeks before the collaborative testing. Christine provided us with a list of required documents and settings, which was very helpful in our preparations. She asked for:

1. A set of DataLoader files (about 10 load files) that we used in recent months in our v.3 production instance. They would be used to retest this loading in the v.4 instance.
2. A list of export files that are regularly performed, with a list of the options on the export tool we use to run them.
3. One archive file of SFX v.3 statistics data, created via the `server_admin_util` option to archive statistics. The file was to be as large as possible, to test importing this archival data in SFX v.4.
4. A list of the top statistics queries that we use regularly in v.3. We would use this to check if they work the same on v.4.
5. A list of the top scenarios we use with the Advanced Collection Tool. We would use this to check if this works the same on v.4.
6. A list of the 10 most frequently performed tasks during regular SFX work.

7. Information about any special setup that we might have and want to recreate on SFX v.4, for example, our institute setup, etc.
8. Information about any menu setup that we are using, such as DirectLink, DirectLink with the banner, display logic rules, etc.
9. A summary of our current SFX setup.

In addition, Ex Libris scheduled two webinars that introduced SFX Version 4, including the new KnowledgeBase structure, the new SFX Admin structure, and the new update process. Also, in a conference call, both Product Working Group leaders gave us an overview of ELUNA 's and IGeLU's expectations for our testing and reports.

Test environment and procedures

Testing took place in the Ex Libris offices in Jerusalem from March 7 to 11, 2010. Each tester had access to a separate SFX installation and was asked to focus on the specific aspects mentioned above. Ex Libris gave us detailed testing plans that included example OpenURLs and scenario descriptions.

Elaborate presentations on the new release, the upgrade process, the new data model, and even the internal procedures of the Ex Libris KB team (who were already working with the new version) were provided by several Ex Libris staff members. An overview of known issues was supplied, as well as documentation for the new release.

The tests were attended by Lieve Rottiers (SFX analyst) and various colleagues from Ex Libris' Quality Assurance (QA) team. They were extremely helpful, answering emerging questions, analyzing problems, and documenting some of our discoveries. In addition, Christine Stohn (SFX Product Manager) was around for the complete week and joined us in the wrap-up meetings.

We kept individual Excel spreadsheets of the defects we found, as well as our suggestions for possible enhancements. For each item on our spreadsheets, we assigned an initial priority of Critical, High, Medium, or Low. Each day ended with a brief wrap-up meeting with Ex Libris to discuss the highlights of our day's work. After the first day, we started each morning with a detailed discussion of our spreadsheets from the previous day. Attending the meetings were representatives from various Ex Libris groups, including QA, development, support, and product management. During our discussions, points were clarified, and issues were loaded into an internal bug reporting tool for further handling.

At the end of the collaborative testing, we were asked to fill out a questionnaire about SFX and the testing process. A closing meeting was held to exchange opinions and discuss some of the statements given in the questionnaire.

Informalities

The hospitality we received from all Ex Libris staff (and especially from product management, QA, and development) felt very good. The organization of our stay, travel, facilities, and the possibilities for social interaction were of a very high level.

Individual test results: Remarks from Jessica

- The upgrade process is much easier in SFX v.4 compared to v.3. The upgrade tool runs much faster than it did in v.3.
- The upgrade documentation does need to be little clearer in which steps can be skipped and which are required.
- There were two instances that did not upgrade. Needs to be some note made of steps to take if upgrade fails. This bug was rated “high” in the reporting to SFX.
- Received email when upgrade is finished. Email contains list of warnings and errors that occurred during upgrade. If you don't get the email, then finding the appropriate log files can be quite difficult. There should be a note in the document where to find the upgrade log file. The log contains information on any errors that occurred during the upgrade.
- Was glad to see shared instances appearing in local SFX menus with no problems after upgrade. There were issues in v.3 upgrade.
- Would like to see monthly update process sometime before general release of v.4.
- Seemed like development team was surprised, at first, at the number of defects we found at first but then seemed appreciative that the testers were finding the defects.
- The layout of v.4 looks better and the navigation is a huge improvement over v.3.
- Nice to see long standing consortia defects in v.3 fixed in v.4.
- Would have been nice to see the search examples and test scenarios before arriving in Jerusalem.
- Would like to see list of what steps will need to be done manually after the upgrade has been applied.
- Submitted enhancements to loosen some restrictions on adding local information. Example is changing parser/parse param, not allowed to change one without changing the other, these two items are locked together in v.4.
- Proxy checkbox removed from Target level causing no proxy in v.4 after the upgrade. Requested to have proxy activated at target service level during upgrade so libraries won't have to go back and re-check the proxy boxes for all of their targets after the upgrade.
- There is a bug in the object portfolio sorting, title sort doesn't work.

Individual test results: Remarks from Inga

- The collaborative testing procedure proved to be very helpful, in particular because the participants represented a multifaceted sample of SFX customers and the testing scenarios prepared by Ex Libris had been well thought out.
- Considering the dimension of the change in the overall database architecture, it was impressive how smoothly the new SFX Admin interface works already. From all I've seen, the database redesign is appropriate to prepare SFX for future challenges (cleaner separation of global from local data, handling of ebook records, etc). However, redesigning the database is a brave move at this stage because SFX became more complex over the last years. I expect that there still are features which need further refinement to incorporate the change, e.g. to present and handle relations between global and local settings correctly.
- It was very informative to test SFX version 4 on MPG data, but also made it more difficult to verify the problems found. The testing setup enabled me to identify various issues related to modifications of the global SFX configuration, e.g. while removing specific target displayers or applying institute specific linking parameters.
- Unfortunately, I failed to test the release systematically and/or thoroughly. In addition, I'm not sure how much automatic testing has been applied. According to Ex Libris, the SFX API has been used to do massive comparison tests between SFX version 3 and version 4. It looks like the SFX API remained stable (sfx.response_type=simplexml tested).
- It was not possible to test the SFX KBUUpdate process because Ex Libris just started testing it in the QA team, thus it was not "ready enough" for customer testing. Instead, we saw the design and a couple of screenshots. Excluding the KBUUpdate process implied an essential restriction of my testing assignment because the update process is by far the "most commonly used workflow" in the MPG.
- Unfortunately, executing KB tools via the command line was not very successful. They still require some rework, but Ex Libris is aware of the problems.
- Need for a generalized "threshold tool": The update process will be changed to a weekly schedule. This puts an additional burden to SFX administrators who try to keep track of the changes applied with each individual update. A generic report generator, which compares local with global settings could relieve us from the need to check each set of update reports individually.
- I like nearly all changes in the SFX Admin interface and believe that some of the new options will improve our maintenance workflows, e.g. the filter and sorting options for lists of object portfolios.
- It was very good to meet the team responsible for the new SFX version. They left an extremely knowledgeable and experienced impression. My gut feeling: "This release is in good hands".

Individual test results: Remarks from Holly

- SFX Version 4 brings numerous enhancements for our KnowledgeBase users. Two that will be especially appreciated are the ability to create our own objects, and the option to receive updates more frequently than once a month.
- My biggest worry about the v.4 human interface was that it was almost too easy to create objects. Within seconds I created half a dozen objects that were duplicates of each other, and duplicate objects cause problems in our OPAC after we run MARCIt. Ex Libris mentioned plans for forestalling the creation of duplicates, and such efforts will be appreciated. It would be good if the KnowledgeBase could check whether an object already exists before it adds a duplicate of it.
- The coming “Add Objects” tool will require training for our advanced users, to ensure that our KnowledgeBase only acquires accurate objects based on standards such as OCLC WorldCat or CONSER. During the collaborative testing, “Search Objects” and “Add Objects” shared a tab, and it looked as if any user who can access “Search Objects” will also be able to add objects. I hope there are ways to separate the access to the two tools, so only advanced users will be able to add objects.
- Another problem I noticed was that to edit a parse param, users needed to make sure that a proper value was in the parser field. I think the two fields should be disconnected from each other because we edit parse params much more often than we edit the parser field.
- We use the Threshold Tool each month to delete local thresholds that are no longer needed. We prefer to use global thresholds wherever possible because Ex Libris diligently maintains them for us. Version 4 has greater differentiations between global and local data, so it would be good if more tools like the Threshold Tool existed for eliminating outdated local data. For example, it would be nice to have a Parse Param tool to help us find and eliminate outdated local parse params.
- Inactive rows in the KnowledgeBase are no longer gray, so sometimes it was hard to tell whether a target, service, or portfolio was active. There's a new “Active” icon, but it was far to the right on my very wide screen.
- There were small inconsistencies in the human interface. Some sections used “D” buttons for deletions whereas other sections used “Delete” buttons. Some sections put the forward and backward navigation arrows on the right side of the screen, and other sections put them on the left. Also, some sections used the handy, new “Local” icon, and other sections didn't.
- I reported small bugs in the retrieval of local titles, the clearing of local data, the alphabetization of targets, combined searches in the “Search Objects” Tool, the “Site down” feature, the “Add Author” feature, the Threshold Tool, the Collection Tool, simplified menus, and related objects.
- On my spreadsheets, I made additional suggestions for the Version 4 software and documentation. I feel they are probably lower priorities than the topics I mentioned above, and I trust that Ex Libris will address them as time permits.

Individual test results: Remarks from Bas

- The general impression of the release is very good, it feels solid and well thought out
- The workflow as used by the KB team has been well tested and seem to be without major faults, customer use cases and scenarios were less well tested
- The major changes in the data model have a very big impact on the whole system, in some cases some instability in the storage of local data was found. More attention for this is required. Copying targets, adding, editing or removing local targets, objects and attributes did not always work correctly.
- Not all of the KB tools were working at the time of testing, testing showed that some of the tools (mostly involving export and publishing) needed more attention in stability, performance and results
- The environment for local categories was not functioning
- A number of high rating bugs concerning the AZ-list were reported
- The overall look and feel of SFX Admin is improved significantly
- Quite a few defects and suggestions were made on details of SFX Admin, there is a fine line between bugs and personal opinions regarding the user interface of SFX Admin.
- Customizing the SFX menu from within SFX Admin is not changed very much, there were some bugs to be found with this.
- Translation works fine except for some minor details
- Statistics, including the loading of older data, works good overall. Some queries were already labeled as “known issues”.
- External relations, for instance with EZProxy and bX, worked fine.
- More validation checks in some of the forms from within SFX Admin is required
- Validation for local targets, objects etc. is in some cases to strict
- Overall performance of the system was good, even under stress caused by extreme simultaneous use of SFX Admin and KB tools. At some moments there was some yet unexplained slowing down of the system.
- Changing the data model will have significant repercussions for many customers, a short survey within the Dutch speaking SFX user community showed that a large number of libraries make direct use of the MySQL database at table and field level. This will require extra attention for both Ex Libris and the customers as part of the upgrade process.
- The direct benefits of the new data model may seem limited, this will also require extra attention in communication about the new version and future developments
- Ex Libris has tried to bring more automation to the KB administration, for instance with the automatic updating of so called “aggregator” targets. In principle this is a good thing, it may not work for every customer, awareness and flexibility in use is important
- Extensive testing of the update process and the resulting reports is still required

Conclusions

1. The quality of the release:

We felt the release was impressive. The underlying database structure has changed dramatically, and the human interface on top of it was also re-designed. We found a fair number of defects, but given the scope of the changes, we are a little surprised that we didn't encounter more problems than we did. We like the new navigation and the look of v.4.

2. The quality of the testing being performed by Ex Libris:

Before our arrival, extensive testing had taken place by Ex Libris, mostly on the use of the KBManager. A list of known errors was supplied at the start of testing, and a presentation was given. The exact extent of the testing by Ex Libris was a little hard for us to determine because their testing took place separately from us, and while we were in Jerusalem, we didn't venture far from our room to find the QA area. In retrospect, perhaps we should have talked more with the QA group about their work on v.4 (although that, of course, may have influenced our testing). SFX development was actively fixing defects while we were in Jerusalem, and they asked us to test some of the fixes. They seemed to be working very quickly to fix defects. The test plans that we received were fairly thorough, and we imagine that the QA staff has also gone through those (or similar) steps. From the people we met with daily, to the individuals we only encountered once or twice in the halls, we saw the same pride in their product. They asked us eagerly, "How is Version 4?" They cared very much about getting things right.

3. The utility of the collaborative testing process:

As a whole, the collaborative testing felt very useful. The Ex Libris staff told us this as well. We found quite a few bugs that had not been discovered previously by the Ex Libris testers. In some cases, the bugs probably wouldn't have surfaced until beta testing or later. The testing scenarios used by us, based on real data and real use cases, were significantly different from the earlier tests by Ex Libris, and they produced worthwhile results. After a day of testing, it was helpful to be able to talk to each other not only about the defects we'd found, but also about the improvements we saw in v.4. Learning how each institution works with SFX was helpful in seeing how it's used outside of our own institutions and workflows. We very much appreciate the chance to voice our opinions from our respective institutions and experiences, and we hope our hosts found the collaborative testing to be beneficial to them also. At the closing meeting, we

received the impression that our testing will result in significant improvements in the release. It was fun to meet Ex Libris staff from the London, Boston, Chicago, and Jerusalem offices.

4. The ways that Ex Libris might improve future testing:

Perhaps it would have helped if we'd gotten a look at the test plans before arriving at Ex Libris. Also, a quick tour of the Ex Libris departments would have been interesting. One of us, who was new to Jerusalem, was surprised to see so many people in the cafeteria because she hadn't pictured Ex Libris as so very large. Later in the week, she was also surprised at the developer's dinner to learn that the developers were only a small subset of the people she'd seen in the cafeteria.

5. The ways that the user groups might improve future testing:

It's hard to think of things that could use improvement. Perhaps we could have learned the expectations of us a little more in advance.

6. Do we think collaborative testing should be a part of any major release?

Definitely YES!

Finally, we are all extremely grateful to Ex Libris, IGeLU, and ELUNA for this wonderful opportunity to provide feedback on SFX Version 4 and to visit beautiful Israel. We were shown a gracious hospitality that we will never forget. Thank you!