

Required Aspects

Explanation/Notes

Unix server environment

We have a significant investment in personnel trained for the Unix environment; PINES is currently hosted on a Sun machine, as are other enterprise-level systems of BOR.

Proven performance/load capacity

The system must have proven performance at the levels we require, with the *type* of load PINES encounters. While our circulation numbers may look comparable, say, to a large metropolitan area public library system, PINES is a very different animal. For example, PINES has a higher cataloging load (10,000 or more titles added, edited, or deleted daily) and a higher number of staff users and public workstations. Whereas a large metropolitan library system will see highly *concentrated* usage, we see highly *diffused* usage. We would like to see software capable of handling double the load PINES currently sees, with a comfortable amount of headroom: in a couple of vectors, that is approximately 40 million annual circulations and 2,500 staff logged in simultaneously. This is the level at which the software would have to operate if all Georgia public libraries joined PINES.

Relational database

For system flexibility and performance, the relational database must be “open” and SQL-compliant. NO vendor-rolled or proprietary database formats. Also, this requirement assumes that if a vendor uses a relational database, it actually uses the database’s capabilities: this is not buzzword compliance.

Implementation as a single union database

We believe in a single conceptual database for PINES. We must have universal borrowing as opposed to generic interlibrary loan, and it must be a well-designed implementation that is transparent, scalable, and easy to report on. Because of this, we dislike the idea of federated databases that are unified through an ad-hoc middle layer. If replication and distributed databases must be used, the work should be done by the database engine itself. We should be able to access this database through ordinary SQL and not have to use custom API's to view the system as a single image. We do not want the added complexity of a middle layer here. Also, the hardware needed to support the software isn't so much of a concern to us: it could be a single monolithic server, or it could be a cluster, as long as there is a single database.

Mature API

Well-documented, supported, and useful Application Programmer Interface (API). We need to be able to modify any software we use and make needed customizations.

Ability to apply configuration changes immediately

The system must be able to handle configuration changes immediately without a restart of the software. (In Unix programming terms: SIGHUP signal capability.)

Required Aspects

Explanation/Notes

Large public library consortia experience

The vendor must have other large public library consortia customers in production. Experience with issues unique to large public library consortia is critical.

Ability to define and work on 'N' tiers

We need customizable groupings for the libraries. Conceptually we operate on a 3-tier system (system, region, and facility), but most vendors really only support 2 tiers. We would actually want an n-tier system for flexibility. These groupings would affect all aspects of the ILS (permissions, circulation, holds, etc.) and would need a flexible management interface (the groups must be able to manage themselves subject to permission restrictions). For example, we should be able to set up a 1. Global level, 2. Area level (east Georgia), 3. Region level (library system), 4. County level and 5. Branch level, if we wanted, with permissions and flexibility cascading correctly and seamlessly down the line. Note this also includes all areas of ILS management: some policy decisions should be retained centrally, while much should be delegated to a lower (regional, or even branch) level.

Immediate indexing

If a cataloger updates a record, those changes should be effective/searchable immediately, both to the staff and patrons.

Powerful holds matrix

The holds matrix must be flexible and powerful. It needs to be able to handle (in a single hold scenario) the patron type, patron profile, patron home library, item type, item owning library, item creation date, item location, and the ability to refer to more data aspects, if needed. The interface in which all of this is logic is laid out needs to be easy to use and maintain. Furthermore, the system must have the ability to see where an item is physically located in PINES, and grab the closest available copy (geospatial ranking).

Client/server encryption

Because our communications go across a public WAN, we need to take steps to protect confidential/personal data. The encryption layer must be integrated into the client and server software. We feel this is necessary, as any software that handles private data such as social security numbers, addresses, and names should utilize data encryption, as is common in the banking industry, for example. While Endeavor currently encrypts certain parts of the data (SSN), we wish to encrypt all data. No other vendor does any encryption.

Required Aspects

Explanation/Notes

Flexible, powerful acquisitions system

In most regards, PINES is a very open system, and everyone can see everyone else's "business". This is fine, and this is the way PINES was envisioned from the beginning—the epitome of resource sharing. However, when it comes to sensitive areas such as funding and budgets, only the library involved should have any access at all to those areas. Furthermore the ability to view account information, holdings codes, and other library-specific aspects should be controlled by login. NOTE: see also "Ability to define local data elements". In PINES, for example, the regional headquarters handles ordering and distribution of materials. The ILS software needs to seamlessly handle all possible cases of local acquisition workflow, and we have not seen this in any available product.

Advanced duplicate title identification

Related to the above, duplicate titles are currently a big problem in PINES. The software needs to be able to detect, according to certain configurable rules, if an incoming title is a duplicate of a title already in place in the PINES database, and give the cataloger the option of aborting the title import, and just attach copies to the existing title. A few of the systems have very rudimentary duplicate detection systems, but none are powerful (configurable) enough to handle the wide variety of materials that are cataloged in PINES. Many vendors talk about the de-duplication capabilities of their software, but often it is extremely rudimentary.

Ability to define local data elements

In PINES, while we have come together and agreed on certain common procedures and practices, we need to have the ability, when feasible, for libraries to have local workflows or procedures. For example, if library A would like a location of "African American History", then they should be able to add that location within the software locally. This extends to all areas of the ILS

Also, where local decisions are allowed, they cannot interfere with the greater PINES system, cluttering interfaces/screens with thousands of options, for example. The software should know that, for example, I have an Athens cataloger logged in, so I only display what is applicable in Athens (of course, what is applicable is locally defined). The management of this is paramount, and it must be easy to centrally "referee".

Kid's online catalog

The idea of a kid's catalog has been enthusiastically adopted by ILS vendors, but has rarely been executed well to any degree. This feature should pull records designated as "juvenile" into a searchable database, with a graphical, easy-to-understand front-end, and should display results first by facility, then by regional system, then by all of PINES.

Required Aspects

Explanation/Notes

Outreach / bookmobile

Public libraries require a module to keep track of bookmobile and outreach/homebound services that operate on alternate scheduling/loan periods. This module must allow entry of "routes", detailing patrons visited on each route, and special due dates demanded by the visitation schedule. Must also allow for variations of fine schedules.

Software support

In a vendor-supported system, we require true 24/7 support; **immediate** response time for critical issues, and 1 day response time for non-urgent calls.

Serials

In PINES, we can have no limits to the number of issues that can be attached to a particular serial record. Also, serials must display in a manageable and easy to understand format in the public catalog as well as the staff client.

Preferred Aspects

Explanation/Notes

“Network aware” staff client software

If the client loses communication with the PINES server, it should automatically roll into a “standalone” mode, and alert the staff member. This should be seamless, and when the staff client software detects the server is available again, the client software should then automatically upload the transactions recorded during the downtime, and immediately alert staff of any problems with the recorded transactions. This functionality is absolutely necessary in a highly distributed environment such as PINES, as there are often local (small, mostly brief) network problems.

“Super Cataloger” interface

Often, very broad and over-arching cataloging work needs to be accomplished in PINES. Because all of the existing software assumes a much smaller environment in PINES, the cataloging software’s workflow doesn’t have needed capabilities that are many times required on a global, system-wide level. One particular place where this need is obvious is in merging multiple duplicate titles into a single title in the PINES database. This is a feature we looked at closely in every vendor’s software, and after trying to merge about 3 titles, all of the available software becomes clunky and difficult to use. Our catalogers must be able to easily identify and select multiple (sometimes dozens) of titles, and merge them in a single click. There are other instances where a “super cataloger” interface is needed in PINES, but the no available ILS meets this need.

Standard HTML/browser independent web catalog

We believe in browser independence. If a patron wants to use Lynx to browse our online catalog, they should be able to.

Booking module

This module allows creation of records for meeting rooms, auditoriums, and equipment. Allows for advance reservations and varying rules of use for each item.

Quality online documentation

When given the opportunity, we’ve found that librarians actually do read well-written and understandable help documents. The software must have extensive documentation for the web catalog (for the public and staff) as well for the staff client. While Sirsi maintains they have thorough and easy to understand documentation, we feel differently. End-user documentation for WebCat is non-existent.

Preferred Aspects

Explanation/Notes

Communication ability through staff client

1. Within the staff client, there needs to be an “announcement” communication interface. In other words, if in the event the sys-admin needs to restart the server, or send some other critical message, we should not have to rely on email. The client software should be able to handle and display incoming alerts, so that the staff working the desks can get critical information quickly. (and not have an email client open at all times). This capability was included in the previous iteration of Dynix’s ILS offering, but it does not currently appear in any ILS.

2. There is also an issue of communication and coordination in cataloging in PINES. Currently, if data work must be done on a title that is owned by multiple libraries (for example, all libraries need to go to the shelf and check their copies attached to the title in question, and if needed, move to a different title), our central cataloging coordinator sends out an email message to a listserv, and the catalogers in each system are supposed to watch the list for issues like this. We’ve found this process/means to be very inefficient, and hard to track. (Lin has to track these issues manually). What we envision, within the staff client software, is an interface in which the PINES cataloging staff can communicate, distribute, and track needed cataloging work. An easy interface where a central cataloger can see outstanding work that must be done, who has completed the task, (and who still hasn’t even read the message). The message to do the cataloging work should appear within the ILS staff client when that particular cataloger logs in, and the cataloger should be able to easily report when the work is complete. This tracking interface, as we currently envision it, will look much like a bulletin board or a task list.

Integrated helpdesk support

While we do not see this as a strict requirement for an ILS, we do see it as an imminent need in PINES, especially as we grow and become more complex. PINES runs a central helpdesk system that any PINES librarian can contact for any sort of technical assistance (or any PINES-related issue). The capability to report a bug or a problem should be included in the staff client. For example, if a circulation librarian finds a bug in a particular hold scenario, he should be able to select a helpdesk option, in which he can describe the problem and report it to the helpdesk. The software will also report, if needed, a screenshot and any applicable client-side logs. Also, staff members will be able to go to the helpdesk, and track any issues they have reported or browse the knowledgebase. (this sort of functionality would be great for bug-tracking)

Inventory

Need formal module for inventory process. Ability to use wireless and handheld (IPAQ) to browse the shelf; check on item status; and ability to mark items lost live.