

AT/Archon Integrated Application: Hi-Level Functional Requirements

This document identifies the critical characteristics, functional areas, and features that must be embodied in the integration of Archon and the Archivists' Toolkit applications. The purpose of this document is three fold:

1. Indicate to the AT and Archon user communities the expected properties of the integrated application
2. Provide the project's technical team a starting point for developing the technical architecture for the integrated application
3. Identify specifications and priorities for both the archives and technical teams.

The archives team (composed of Chris Prom and Scott Schwartz representing Archon and Brad Westbrook representing AT) derived these high-level requirements from work done last fall. A features matrix, first developed by Annie Ross of the Archivists' Toolkit team and later used by Lisa Spiro in her CLIR report on archives management software, was analyzed to generate a provisional list of requirements for the integration of AT and Archon. That list was supplemented by additional functionalities incorporated into AT and Archon after the features matrix was published. This provisional list of high-level requirements was distributed to the archives community on October 30, 2009, via several listservs. Recipients were invited to respond with questions and/or suggestions for adding or deleting requirements. Sixteen responses were received (see posting on AT Forums at <http://archiviststoolkit.org/node/153> for details), and suggestions contained in them were integrated into the list of high-level requirements.

The revised list of requirements served as the basis for two two-hour webinars that took place on December 8 and 10, 2009. Thirty-eight individuals were invited to participate in these webinars, 10 from the Archon community, 10 from the AT community, 10 representing various standards bodies and other groups, and 8 representing the project / webinar team. The core functionalities proposed in the high-level requirements were described in detail and discussed with the webinar participants. After the webinars were completed, participants were then asked to respond to a set of questions designed to indicate the relative importance of specific functions within the overall set of requirements. Their recommendations were then used by the archives team to develop a final prioritized list of high-level requirements which begins on page three.

The archives team also has provided additional information about whether a requirement will need a new specification, a revision of an existing specification(s), or no specification. This information is primarily intended to serve the archives team to accomplish its specification work. It will also guide the initial work of the technical team and the project team by identifying all specifications needed for this new application to be thoroughly built to the satisfaction of the user community.

The archives team, at the request of the project administrators, has added to this list of requirements a set of "Framing Principles" to guide the overall development of this integrated application. These principles establish the expected characteristics of this new application which may not be explicitly articulated in these high-level requirements.

This list of requirements will be supplemented with further functional and technical specifications authored by the archives and technical teams. These specifications will lead to the development of a stable data model for the integrated applications.

Framing Principles

The new application must:

1. have all the data management power of a client-based application and all of the access/input/output flexibility of a web-based application.
2. be designed around a core data and object model that will allow for quick search, retrieval, and display of information of any kind.
3. be scalable to millions of records.
4. be designed to lower maintenance over-head, especially the costs associated with testing and bug resolution.
5. present the lowest possible barriers for acquisition, installation, implementation, and operation for archives of all sizes.
6. be implementable using simple installers without extensive prior configuration or the installation of complex and/or non-standard dependencies.
7. be installed/packaged for the following uses:
 - a. a single user without a network connection (perhaps packaged as a virtual appliance).
 - b. one or more repositories in a networked configuration, using widely-available server configurations;
 - c. a hosted service for multiple communities (potentially allows for implementation across multiple servers as a service oriented architecture).
8. provide granular access controls for staff users and user groups.
9. support Unicode character encoding.
10. support the creation/installation of alternate UI languages.
11. utilize 1) open source code; 2) a standard plugin framework; and 3) industry standards for data exchange.
12. support core archives administration functions
13. support the migration and import of common data types.
14. support the creation of the metadata.
15. utilize a staff user interface that will support all archives functions/requirements.
16. provide public web access functionality to repositories / users who choose to provide users direct access to collection data.
17. include either:
 - a. integrated digital object storage or
 - b. a migration pathway for digital objects currently stored in Archon, so that access can be preserved in other repository systems
18. include a web-like interface for authoring/editing records.¹
19. include functionalities to expedite data entry, manipulation, and reporting.

¹ During the webinar, many respondents spoke in favor implementing the staff editing interface on a web platform, and 87% of respondents in the post webinar survey stated they would prefer to use a web application. (20% indicated they would prefer a client application, indicating some overlap and/or confusion regarding the question asked.) If feasible, the technical team should consider and or suggest technologies that would allow for client-like functionality in a web interface (e.g., AJAX), and/or a hybrid approach (dual interfaces). In any case, the technical team should consult carefully with the archives team when selecting a technology or technologies to implement the requirements listed under "Application Functions."

In the table below, ratios in the “Must Have + Very Useful” column and numbers in “Overall Ranking Avg.” column are derived from the survey conducted after the AT / Archon HiLevel Requirement webinars conducted December 8 & 10, 2009.

Four measures are provided in the Status column below:

- Required— absolutely must be in the first release of the new application
- High— should be in the first release of the new application. Otherwise consider for subsequent releases.
- Low— consider for including in the first release of the new application. Otherwise consider for subsequent releases, or as a community provided plugin or independent application
- Not do— will not include this functionality in the first or subsequent releases of the application. Consider as a community provided plugin or independent application.

Four indicators are provided in the Specification Work Required column below:

- AT-- Revise AT spec
- Arch-- Revise Archon spec
- Combine-- AT / Archon spec
- New-- Compose new specification
- None-- No spec needed

AT / Archon HiLevel Requirements					
Groups	Functional Area	Must Have + Very Useful	Overall Rating Avg.	Status	Specification Work Required
Archives Functions					
	Resource description	100%	3.89	Required	Combine
	Accessions management	89.5	3.58	Required	Combine

	Name management	94.4	3.56	Required	Combine
	Subject management	89.4	3.37	Required	Combine
	Digital object description	78.9	3.21	Required	Combine
	Location / space management	57.9	3.00	Required	Combine
	Collection management	68.5%	3.00	Required	New
	Repository Information	70%	2.95	Required	Combine
	User (staff) management	55%	2.90	Required	Combine
	Deaccessions managements	57.9	2.79	Required	AT
	User (public) management	36.8%	2.26	Required	Archon
	Classification			Required	Archon
	Rights management	52.6%	2.63	High	AT
	Repository Statistics	60%	2.65	Low	AT
	Assessments (General)	52.6%	2.58	Low	AT
	Assessments (Audio Visual)	36.8	2.21	Low	Archon
	Work orders	21.1	1.58	Not do	AT
Imports (by record type)					Completion of import specifications are dependent on stabilization of integrated application data model
	Database migration <ul style="list-style-type: none"> From AT (latest version) From Archon (latest version) From Integrated application 	94.7	3.84	Required	Three specifications, all new

	Resource—EAD	89.5	3.63	Required	AT
	Accessions—CSV format	72.2	2.89	Required	AT
	Resource—MARCXML	58.8	2.82	Required	Combine
	Digital object—CSV format	66.7	2.67	Required	AT
	Subjects—from MARCXML	56.3	2.63	Required	AT
	Names—from MARCXML	56.3	2.50	Required	AT
	Accessions—XML format	37.1	2.41	Required	AT
	Names—EAC	64.7	2.53	High	New
	Digital objects—XML	70.2	3.00	High	New
	Subjects—XML	55.5	2.61	Low	New
	Subjects—CSV	50.0	2.56	Low	New
	Names—CSV	47	2.47	Low	New
	Location—CSV	52.7	2.42	Low	New
	Names—MADS	13.3	1.47	Low	New
Exports (by record type)					Completion of import specifications are dependent on stabilization of integrated application data model
	Resource—EAD	89.5	3.74	Required	AT
	Resource—MARCXML	76.4	3.24	Required	Combine
	Digital Object--DC	66.4	3.00	Required	AT
	Resource—Container labels	61.1	2.94	Required	AT
	Digital Object—MODS	62.6	2.81	Required	AT

	Names—EAC	61.1	2.67	Required	New
	Digital Object—MARCXML	62.6	2.63	Required	AT
	Digital Object—METS (DC)	41.2	2.59	Required	AT
	Digital Object—METS (MODS)	41.1	2.53	Required	AT
	Resource—Folder labels	33.3	2.39	Required	New
	Digital Object—PREMIS Rights MD	37.6	2.06	Low	New
	Digital Object—PREMIS Technical MD	31.3	2.00	Low	New
	Names—MADS	25.1	19.4	Low	New
Reports					Completion of import specifications are dependent on stabilization of integrated application data model
	Repository Profile			Required	AT
	Repository Productivity			Required	AT
	Registered users sortable by defined characteristics			Required	New
	Material used sortable by user or defined material characteristics			Required	New
	Extent of holdings			Required	New
	List of accession, resources, digital objects processed / unprocessed			Required	AT
	List of accession, resources, digital objects cataloged / uncataloged			Required	AT
	Accession(s) with inventory			Required	AT
	Accession(s) with processing plan			Required	AT
	Accession production			Required	AT
	Accession receipt(s)			Required	AT
	Accession record(s)			Required	AT

	Accession(s) acquired			Required	AT
	Accession(s) with deaccession			Required	AT
	Accession(s) with linked name(s) and subject(s)			Required	AT
	Cumulative list of surveyed materials			Required	AT
	Surveyed materials with conservation issues			Required	AT
	Assessments requiring review			Required	AT
	Digital object list			Required	AT
	Digital object table			Required	AT
	Digital object record(s)			Required	AT
	Digital object file version list			Required	AT
	Digital object with deaccession			Required	AT
	Digital object restricted			Required	AT
	Location shelf list / locations with resource or accession record(s)			Required	AT
	Names list			Required	AT
	Names record(s)			Required	AT
	Name(s) with linked resource(s), digital object(s), and accession(s)			Required	AT
	Name(s) with non-preferred name(s)			Required	AT
	HTML finding aid			Required	AT
	PDF finding aid			Required	AT
	Resource record(s)			Required	AT
	Resource(s) restricted			Required	AT
	Resource(s) with location(s) inc. barcodes			Required	AT
	Resource(s) with name(s) linked as creator			Required	AT
	Resource(s) with name(s) linked as source			Required	AT
	Resource(s) with linked names and			Required	AT

	subjects				
	Resources list			Required	AT
	Resource(s) with deaccession(s)			Required	AT
	Subject record(s)			Required	AT
	Subject(s) with linked resource(s), digital object(s), and accession(s)			Required	AT
	List of occupied storage space with summary total of space used			High	New
	List of un-occupied storage space with summary total of space not yet used			High	New
	List of all deaccessions sorted by resource, accession, digital objects			Required	New
	List of all resources, accessions, and digital objects that have been completely deaccessioned			Required	New
	Acquisition acknowledgements sent			Low	New
	Acquisition agreements sent / received			Low	New
	List of IP rights transferred to repository / institution			Low	New
	List of rights holders by type, identifier, material			Low	New
	List of rights type with identifier and right holders and restriction dates			Low	New
	List of restrictions in effect			Low	New
	List of restrictions expired			Low	New
	List if copyrights expired			Low	New
Stylesheets					Completion of import specifications are dependent on stabilization of integrated application data model
	EAD > HTML			Required	AT

	EAD > PDF			Required	AT
	EAC > HTML			Required	New
Application features					
	Enable individual data fields to be omitted from public output	94.4	3.44	Required	Archon
	Fielded searching	77.7	3.00	Required	Combine
	Interface / method for connecting with, and dynamically updating, metadata records in other web-based digital object access and/or repository systems	72.2	2.78	Required	New
	API includes generic method to relate any objects of one type (e.g names) to any objects of another type (e.g. resources) and to specify nature of relationship			Required	Combined; full list of required object relationships will be provided for each archival functions area as spec is developed.
	Public Browse interface for specific types of data content, with links to related content			Required	Archon
	Integrated data / record validation rules, processes, and messages			Required	AT
	Context specific menus / tab system for each functional area			Required	Combine

	Integrated help / tool tips, with modifiable content			Required	Combine
	Internal bug report mechanism			Required	AT
	Configurable UI Language			Required	New
	Configurable UI Labels			Required	AT
	Ability to link to and open external documents within specified records			Required	Combine
	Drag and drop for hierarchical displays			Required	AT
	Drag and drop for notes			Required	AT
	Filters (auto-adjusting)			Required	Combine
	Configurable user defined fields for most record types			Required	Combine
	Wrap and tag XML editor for specified records			Required	AT
	Configurable browse / return screens			Required	Combine
	Configurable search editors			Required	AT

	Configurable “sticky “values” for specified record types			Required	AT
	Configurable “look up” lists			Required	Combine
	Configurable default values			Required	Combine
	Customizable public output templates: Accessions, resources/collections, Digital objects, Name authorities, subjects, and classifications			Required	Archon
	Ability for user to limit search queries by date	83.3	3.11	High	Archon
	Batch editing			High	New
	Global find and replace			High	New
	Spell checker for selected records / fields			High	New
	Configurable record validation requirements (for increasing record requirements only)			High	New
	Ability to specify how search	44.5	2.39	High	New

	results are to be weighed				
	Request management	35.2	2.29	High	Archon
	Web 2.0 bookmarking and end-user annotations	44.4	2.28	Low	New
	E-commerce features	33.3	2.06	Low	Archon
	Command line interface			Low	New
Web access					
	Customization Features				
		Theme system	Controls overall look and feel of public web application, default theme includes search and browse features, repositories can define home themes via header, footer, and css	Required	Archon
		Template system	Controls output/ordering of elements on specific pages, each page provides	Required	Archon

			clickable links to related content (e.g. resources to digital objects, names, etc and vice versa)		
--	--	--	---	--	--

User Interface Features/Common Behaviors and Properties

We recommend that a significant amount of time be placed into designing a basic staff and public user interface(s) that is **simple, logical and consistent in layout**. The interface(s) for staff should be built using a theme system, so that if repositories wish to change properties in the public or staff interface(s) (such as color, font size, branding, and certain layout principles), they can do this easily.

Generically, the staff user interface should include the following features:

- The ability for individual users to edit 'my preferences', including UI language, Password, and other configuration values/views to be defined later.
- Links out/preview (to web view for whatever is being edited)
- Direct access to editing interface from public web interface for authenticated users (i.e. Archon's edit pencils)
- Drag and drop reordering for hierarchical/nested content
- Configurable labels
- Configurable templates (rapid data entry screens)
- Configurable tool tips/help (label rollovers containing as defaults: definition, examples, reference to standards)
- Configurable user defined fields (per selected templates)
- Wrap/tag editor for certain fields (e.g CK editor or another bbcode-like editor, but also including some support for semantic markup into EAD output)
- A method to suppress any particular data elements from public output. Ideally, this would be granular enough to be configured for particular repositories, or even individual records (such as particular resource records)

System Admin Panel

The staff user interface should include a system administration panel. This panel should be visible and accessible only to users who have been granted access to this area. The system admin panel should use a consistent layout and design, similar to that described below for the staff user interface. At a minimum, this system admin panel should allow 'admin' users the ability to define:

- User accounts and permissions
- Package/module management (e.g. to install, enable, and uninstall entire areas of the application such as subjects, digital objects, or other areas)
- Database management (e.g. viewing of database stats; bulk imports, exports of defined data types)
- Default UI Language selection
- Repository information (names, address, stats)
- Report generation and exports (see reports section of requirements list) Should include canned reports, and the user should have the ability to define additional reports using data from all numeric fields and from record counts, for all areas of the application.
- Installation/configuration of new packages, modules, and plug-ins.
- System configuration values (such as length of result sets, field weighting for relevancy searches, or other values to be determined during the detailed specification development.)
- Batch editing/global find and replace.

Staff User Interface

Features such as the user control area, browse/filter area, main content editing area, and help area should appear in the same location on every screen, no matter what type of object is being edited. The technical and UI teams should consider interfaces for other successful web applications. The technical team should also examine the Archon 3.0 UI design and the AT design for potential UI design ideas. However, they should implement a unified approach based on established UI principles and use a design that is simple, logical, and consistent in layout and expected behaviors.

At this time, we would suggest that the UI should include the following three basic areas, with the sub-elements and sub-features listed in each area:

- High-level menu system to allow access to all application features to which the user is allowed at least read-only access.
 - It should allow sliding to reveal or conceal the entire menu. In addition, the specific sub-areas (e.g. collections, digital objects, accessions, subjects) should be expandable and collapsible.
 - The menu system should be customizable.
 - If a user does not have at least read access to the information provided in an area, a menu item should not display.
- User control area The user control area should provide a consistent, simple set of buttons that will enable the user to:
 - Add a new record
 - Save changes to the current record being edited (no changes should be saved until the button is clicked).
 - Undo/cancel current changes (prior to record being saved).
 - Remove/delete current record and all children of the current record. A warning should be provided before the action is completed. If user does not have delete power for the data type being displayed, the button should not appear.
 - Duplicate existing record (ID/title or other unique values change).
 - Access child content for editing (such as child resource records or child classifications)
 - Get help.
- A main content area for editing whatever type of data has been loaded (names, subjects, resources, etc). While this main editing area should go through significant UI design phase, we would like to suggest the following basic design principles for consideration:
 - A sub-area to browse and filter long lists of objects.
 - Preferably, this would be located on the first tab or an area outside the tab system (see below). The technical and UI teams should explore making this area concealable and lockable, although it will be important to clearly differentiate it from the main menu system described above.
 - It should operate more or less like the Windows file manager or Mac Finder
 - Expanding/contracting folders
 - Includes filter/search to narrow long lists
 - Will allow users to locate records, load them for editing
 - Will allow transfer to new nodes in the hierarchy, using drag/drop.
 - If the user is at the 'root' node for current object type (i.e. no record is loaded for editing), all records should be accessible via the sub area. The user will navigate through the hierarchy of objects and or use the filter box to locate a record for editing.
 - If the user has a record loaded for editing, only objects that are hierarchically-related to the data object that is currently being edited (e.g. child and parent resources), will be shown.
 - Ideally, information displayed in this area should be configurable by individual users. However, this is not an absolute requirement as long as the browsing and filtering features are intuitive and responsive.
 - Tabs to group similar information under the object being edited. For example, tabs provided when a resource is being edited might be "Resource Identification" "Description" "Subjects" etc. If tabs are used, the following basic principles should be included/supported:

- The list of specific tabs will be different for the different object types with different object types (e.g. different tabs for resources, subjects, etc.)
- All required fields should be grouped on a single tab.
- Related objects (e.g. names to resources, accessions to digital objects) should each appear on separate tabs.
- User-defined plug-ins should appear as new tabs at the end of the existing tabs.
- Relations: relations tabs or areas should include a dual pane interface and a 'quick add' feature so that users can relate items (such as subjects, storage location) to the main object without needing to leave their current work (see Archon 3.0 interface for one idea regarding how this might work.)