

System Design for Bucket Tool

ODU DLIBUG group

First release: Dec-5-1999
Latest update: Jan-21-1999

Index

1	SCOPE.....	3
1.1	IDENTIFICATION.....	3
1.2	SYSTEM OVERVIEW.....	3
1.3	DOCUMENT OVERVIEW.....	3
2	APPLICABLE DOCUMENTS.....	3
2.1	BUCKET PAGE:.....	3
2.2	DIENST PAGE.....	3
2.2.1	<i>Dienst protocol 4.1.....</i>	<i>3</i>
2.2.2	<i>How to install and administrate Dienst.....</i>	<i>3</i>
3	DEFINITION.....	3
4	FUNCTION REQUIREMENT.....	5
4.1	SERVER.....	5
4.2	CREATION TOOL.....	5
4.3	MANAGEMENT TOOL.....	5
4.4	ADMINISTRATION TOOL.....	5
4.5	SECURITY MODEL.....	5
5	SYSTEM DESIGN.....	5
5.1	HANDLE PROCESS.....	5
5.2	SECURITY MODEL.....	5
5.2.1	<i>Tool security.....</i>	<i>6</i>
5.2.2	<i>Bucket Security.....</i>	<i>6</i>
5.2.3	<i>Combination of toolkit and bucket security.....</i>	<i>6</i>
5.3	CREATOR TOOL.....	6
5.4	MANAGE TOOL.....	8
5.5	ADMINISTRATION TOOL.....	9
6	APPENDIX.....	9
6.1	DIRECTORY ORGANIZATION.....	9
6.2	VERSION HISTORY.....	9

1 Scope

1.1 Identification

Old Dominion University's Bucket tool project has several design requirements for satisfactory completion. These requirements include bucket design, system design, creation tool design, management tool design and administration tool design.

1.2 System overview

This project is an attempt to create mechanisms used to submit material to archives. Specifically, in ODU digital library, we are focusing on creating and managing buckets. The characteristics of buckets tool will include:

- Creation tool: Create new buckets from a template.
- Management tool: Approve a new-created bucket.
- Administration tool: Mass update selected buckets; system management.
- Security model: All tools listed above will be abided to one security model.

1.3 Document overview

The purpose of this document is to define the overarching system design for the bucket tools, this document is focusing on relationships between different tool and security model.

2 Applicable documents

The following documents of the exact issue shown shall form a part of this specification to the extent specified herein. In the event of a conflict between the documents referenced herein and the contents of this specification, the contents of this specification shall be considered a superseding requirement.

2.1 Bucket Page:

<http://home.larc.nasa.gov/~mln/buckets/>

has a very simple page about buckets. it includes:

- a fully documented API for bucket version 1.3.2 (the UPS templates were based on this version).
- a list of proposed methods.
- a list of all known bucket templates.
- a list of possible bucket/DL projects.

2.2 Dienst Page

2.2.1 Dienst protocol 4.1

<http://www.cs.cornell.edu/NCSTRL/protocol.html>

2.2.2 How to install and administrate Dienst

http://www.cs.cornell.edu/NCSTRL/dienst_install_admin.html

3 Definition

To ensure consistency within this document; there are a number of terms that require definition. They are:

- **Access Groups:**
Collection of users. Elements of one group have same access control privilege, what privilege is defined by bucket itself. Available group may include manager group, creator group, administrator group and many viewer groups.
- **Administrator Group:**
Group of user that has the administration right in a specific server.
- **Bucket**
Bucket is directory named by handle of bucket; Index.cgi is main method for accessing bucket; other methods are stored in bucket server; methodredirect file (index.cgi uses redirect to invoke method), bib file, T&C files (password, domain,IP), state file (containing status about bucket such as submitted, temporary,...) ; files constituting bucket.
- **Bucket Repository**
Place where buckets are deposited. It's possible that there is some sub directory called authority under bucket repository, under authority we have all buckets as subdirectories named by handle(unique authority plus unique id within authority) of bucket; this subdirectory is bucket itself.
- **Bucket Status**
Bucket status field will be one of: accumulation, evaluation, published. When creator is still editing bucket contents, bucket status is "accumulation"; after bucket is submitted to manager, its status is changed to "evaluation"; after manger approves bucket and put it in publishing stage, bucket status is changed to "published". This field shall set/get by bucket API set_state/get_state.
- **Bucket Templates**
Each server will have a directory of sample buckets with no content but structure(e.g., one element bucket) or generic buckets for a particular collection (e.g., LTRSbucket). At least two kinds of bucket should be defined: no restrictions or author and manager can edit metadata.
- **Creator Group**
Group of user has the right of creating new bucket in a specific server.
- **Handle**
Unique name of archive collection plus unique id, for example: org.nasa.ltrs//tr-1999-1, edu.odu.cs//tr-99-01. It's a global identifier of bucket.
- **Manager Group**
Group of user has the right of approving buckets in staging archival, and setting buckets t&c in staging archival.
- **Publishing Archival**
One kind of bucket repository, each server will have a publishing area for buckets that had been approved by manager and can make public service.
- **Staging Archival**
One kind of bucket repository, each server will have a staging area for buckets to be approved for publication into an archive/archive's collection (a directory in well-known place called staging archive).
- **Sub- Staging archival**
Staging archival may includes sub-archival, these sub-archival should be assigned by author at the creation of bucket. Normally sub-archival may be defined by author or other structure information.

4 Function Requirement

4.1 Server

Every installation of NCSTRL+ has its own bucket tool (creation, management, administration), publishing archival, group file, passwd file and staging archival, these tools and files may just operate on buckets located in this installation.

4.2 Creation tool

Creation tool creates new bucket from pre-defined template, create/modify bucket metadata, create/modify/delete package, upload/delete element files in bucket.

4.3 Management tool

Management tool see list of pending buckets (implicitly what's in unix file system under staging archive), can review metadata and approve or disapprove (communication off-line outside toolkit, e.g., e-mail), set bucket handle, setup bucket t&c.

4.4 Administration tool

Administration tool administer authentication and access control; create templates; set location for code factoring, location of server and various groups, mass updating buckets, etc.

4.5 Security model

Authentication is enforced by the web server security.

5 System design

5.1 Handle Process

Handle is comprised of unique name of archive collection plus unique id. In NASA, document id is generated by policy and typed in manually and not done automatically.

The handle generation procedure is defined below:

1. Creation tool will generate a unique id for each bucket in local server. Then handle (archival global id + unique id in local server) is global unique.
2. In manager tool, manager must change this id to NASA assigned document id. Logically NASA assigned ID is unique, but in order to avoid any typing error, manager tool must check the uniqueness of document ID in publishing archival.
3. The old local ID is just discarded, NASA assigned ID will be kept as bucket handle.

Example:

Gov.nasa.ltrs//tr-1999-01

5.2 Security model

There are two kinds of security issues: toolkit security and bucket security. Both use authentication model provided by web server. They mainly work independently but have interaction. This chapter will cover toolkit security, bucket security and their relationship.

5.2.1 Tool security

Basic username and password authentication enforced by the Web server, currently, Apache web server security model is used.

There are three pre-defined user groups: creator, manager and administrator group. Only users in creator group can use creator tool. Only users in manager group can use manager tool. Only users in administrator group may use administrator tool.

Below is a scenario of bucket administration tool:

1. User accesses the administrator page (URL) that is restricted by Web Server security mechanism.
2. Web server requests the client to send the user name and password.
3. Client (Web Browser) prompts a window for user to enter the username and password.
4. Client sends the username and password to the server and also caches it for future use. (Note that if the user goes to the same URL or to a URL, which has the first part identical to the original URL, the client sends the username and password with the request. This way user is not prompted for username and password again.)
5. Once the basic authentication is done the bucket logic (or the factored logic) enforces the ACL by getting the username from CGI environment variable and checking against the group file stored locally.

5.2.2 Bucket Security

Bucket terms and conditions are currently implemented using the facilities available via http and CGI. Building from the CGI facilities, buckets implement simple access control lists (ACLs) that restrict access based on username/password pairs, hostnames, and Internet Protocol (IP) addresses. It is also possible to apply these restrictions to entire methods, entire packages, or package/element pairs.

Bucket security setup has to be implemented by combination of different tool. Template, creator tool, management tool and administration tool will cooperate to determine bucket T&C.

1. In administration tool, administrator may define series of T&C template. The t&c of bucket template is unrestricted.
2. In creator tool, creator will select appropriate bucket template and create new bucket under creator's private directory. After that, new bucket t&c will be configured to only current creator write-enabled.
3. When bucket is submitted to staging archival, bucket t&c will be configured to only current manager write-enabled.
4. In management tool, manager will setup appropriate t&c for bucket. After approval, these new t&c setup will be saved in bucket.
5. In administration tool, administrator may update t&c of buckets.

5.2.3 Combination of toolkit and bucket security

- Bucket has its own security model. However, bucket will use group file and passwd file provided by toolkit to authenticate user.
- Before bucket is finally released as formally published, it will be protected by both toolkit and bucket security mechanism. In implementation, we will put these buckets in appropriate working directory under different tool.

5.3 *Creator tool*

5.3.1 Create/Edit new bucket

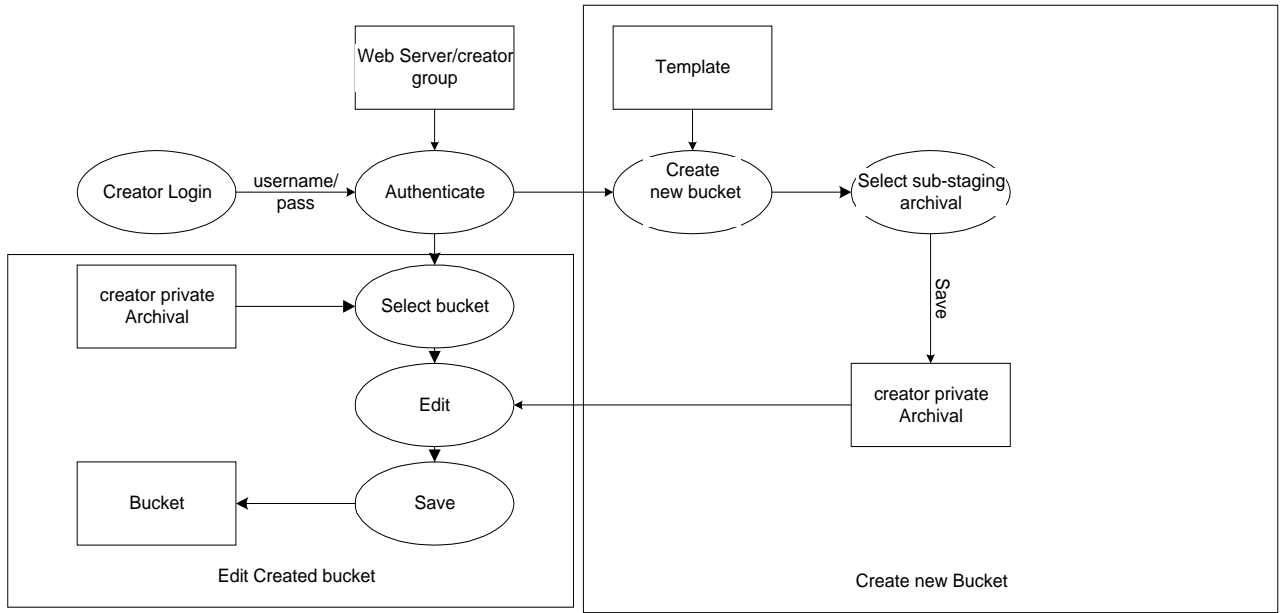


Figure-1 Create/Edit new bucket

- Any user in creator group may create a new bucket.
- After new bucket is created, only current user can view and edit it, this is guaranteed by bucket security mechanism itself.
- Creator tool will edit bucket metadata and add/remove/edit package and element. However, Creator will not decide on bucket t&c setup.
- Creator shall select appropriate manager to approve bucket, this bucket will be put in manager's private staging archival.

5.3.2 Edit published bucket

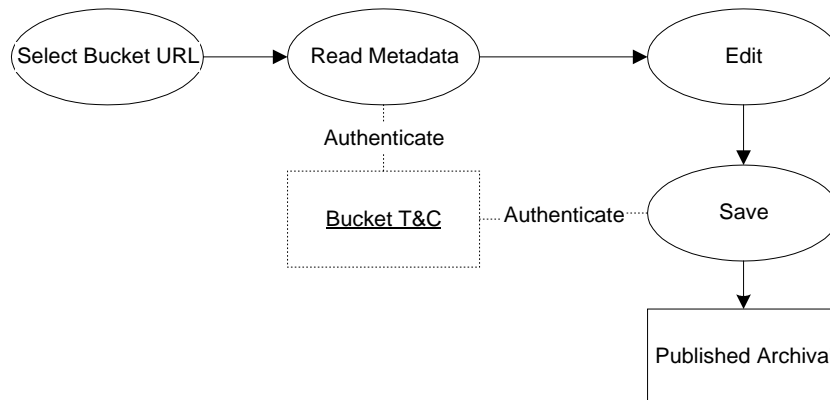


Figure-2 Edit published bucket

- This procedure is only used in published bucket.
- There is no login procedure in published bucket edit tool. It's assumed that bucket security is ensured by bucket itself.

5.4 Management tool

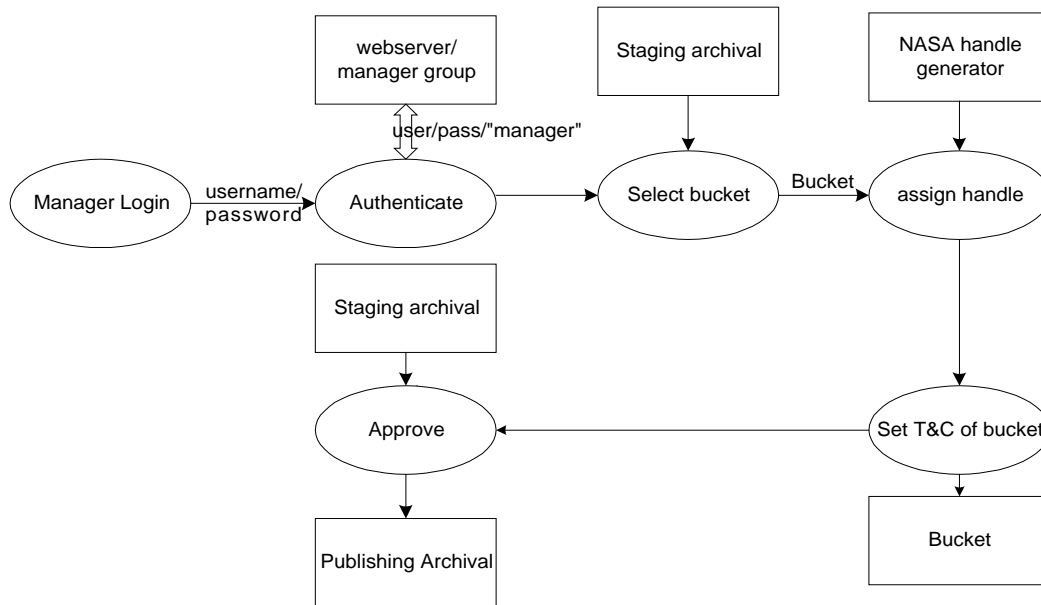


Figure 3 management tool

- Management tool doesn't provide bucket metadata editing function, however, if this user is in author group at the same time, he can login to author tool and edit bucket.
- NASA has some mechanics to create handle (document id), this document id should be assigned by manager.
- Manager decides term & condition of bucket.
- Management tool should provide interface for add/delete methods, display log and other management work.
- Every manager has own archival, and he can only processing bucket in this directory. All buckets in this directory is this manager write-enabled.

5.5 Administration tool

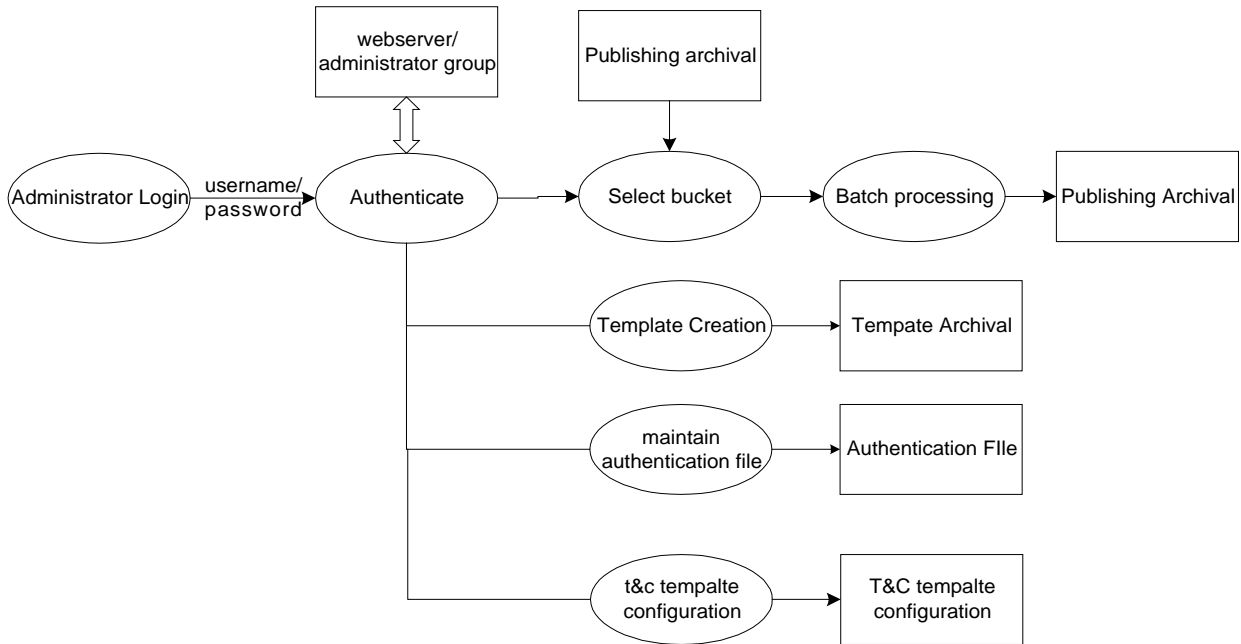


Figure 4 Administration tool

Administration tool has two main kinds of functions:

- Initialize System and maintain various system tables, including Authentication file, bucket template, default handle generation.
- Batch processing bucket.
- Administrator group has all privilege of bucket, this should be pre-defined in any bucket template.

6 Appendix

6.1 Directory organization

Ncstrlplus.cs.odu.edu/search	Search Interface (default)
Ncstrlplus.cs.odu.edu/author	Authoring tool
Ncstrlplus.cs.odu.edu/manage	Management tool
Ncstrlplus.cs.odu.edu/admin	Administration tool
Ncstrlplus.cs.odu.edu/tc	T&c template
Ncstrlplus.cs.odu.edu/manager/stage	Staging archival
Ncstrlplus.cs.odu.edu/template	Bucket templates

6.2 Version History

- Dec-7-1999
Manager tool process t&c of bucket, including add/delete author; add/delete method; add/delete principles.

Decide that at least two kinds of template are needed: no restrictions or author and manager can edit metadata.

Handle format;

- Dec-5-1999 -First release