

DuraCloud Bridge Intake -

1. Chronopolis Intake service polls the Bridge application for new snapshots to process
2. Chronopolis [Intake service](#) uses the content in bridge storage to construct a bag for deposit
 - a. The Intake service validates content against the manifest written by the bridge application (filepath, SHA-256 of each payload file, if a file is missing on disk that is listed in manifest, bagging will fail. Files not in manifest will be ignored)
 - b. The Intake service creates the necessary bag files (bagit, bag-info, etc) that are included in the bag
3. Once bagging is complete, the intake service will make a call to the Bridge application to indicate that the content has been successfully Bagged
4. Information about the Collection is registered through the Ingest Server's API
 - a. Files and fixity (file hash SHA-256) are registered with the Ingest Server database.
 - b. Staging information for the location of the Bag on disk
5. Ace Tokens are created for the Bag for other Chronopolis nodes
 - a. Requests are sent to the ACE IMS for the ACE Tokens (show example of token request)
 - b. Tokens created for payload and tag files
 - c. All files will have their hashes validated during token creation
 - i. Failure to validate will result in the file not receiving token. Bag will not be processed by ingest server.
 - d. Tokens registered through the Ingest Service API

ChronCore Ingest

6. Creates an ACE Token Store on file system local to ingest service containing all ACE Tokens for a collection in the ingest database
7. Chronopolis [Ingest service](#) creates replication requests for the appropriate Chronopolis nodes
 - a. Requires all ACE Tokens to be registered for a Collection
 - b. Requires all Files and Fixity to be registered for a Collection
 - c. Requires staging information to be present for the Collection and ACE Token Store

ChronCore Replication

8. Rsync is used to transfer ACE Token Store and Collection to Chronopolis nodes
 - a. ACE Token Store located at the Ingest Server's filesystem
 - b. Collection is still located on its staging resource where bagging occurred
 - c. Failed rsyncs are retried.
9. Collection file path info, name, checksum algorithm, and depositor info is registered with local ACE-AM node instance. (top-level collection info)
10. ACE Token store used to populate the collection with its initial set of monitored files and ACE Tokens

11. ACE-AM audit is kicked off, retrieves the summary of the ACE token from ACE-IMS, uses the hash of the file on disk to compute the summary of its local ACE token. Compares both summaries - requires to be the same for the file to be considered valid.
 - a. Example of ACE Token summary
 - b. Failures result in manual steps (reaudit, stage new replication, etc.)
12. Once collection is valid in ACE-AM, Replication Service notifies Ingestion Service that the replication has completed successfully. Individual completed replications are set to "SUCCESS" in Ingest service.
13. Once all replications have completed successfully, collection state in Ingest Service is set to "PRESERVED"
14. Intake periodically polls ingest service for status of collection. When status is PRESERVED:
 - a. The intake will remove the collection from its staging resource.
 - b. Calls bridge to update history with replication information (node locations, collection ID) of the collection
 - c. Calls bridge to complete the snapshot (link to bridge api complete snapshot)