

# AipCoreAPIChanges

- API Changes in the current AipBackupRestorePrototype
  - Change Summary (work in progress)
    - configs
    - org.dspace.app.packager.\*
    - org.dspace.content.\*
    - org.dspace.content.crosswalk.\*
    - org.dspace.content.packager.\*
    - org.dspace.handle.\*
    - org.dspace.workflow.\*
    - dspace-lni-api
    - dspace-sword-api

These patches are now out-of-date, but are left here temporarily for reference. The latest AIP code is now available on DSpace Trunk: <http://scm.dspace.org/svn/repo/dspace/trunk/>

## API Changes in the current AipBackupRestorePrototype

- See SVN Sandbox for actual source code: [http://scm.dspace.org/svn/repo/sandbox/aip-external-1\\_6-prototype/](http://scm.dspace.org/svn/repo/sandbox/aip-external-1_6-prototype/)
- See JIRA issue: <http://jira.dspace.org/jira/browse/DS-466>

### Change Summary (work in progress)

#### configs

Full patch : [dspace-configs.patch](#)

- `dspace.cfg` – new configs for AIPIngestor & Disseminator. Also added configs for some new crosswalks used by the AIP ingest/disseminate functionality.

#### org.dspace.app.packager.\*

Full patch : [org\\_dspace\\_app\\_packager.patch](#)

- `org.dspace.app.packager.Packager` – MAJOR refactoring to move most of the actual processing code into `org.dspace.content.packager` classes (this allows most packaging to also be called via API calls, rather than only through CLI). Packager is now basically a CLI to the `org.dspace.content.packager` classes.

#### org.dspace.content.\*

Full patch : [org\\_dspace\\_content.patch](#)

##### UPDATED CLASSES

- `org.dspace.content.BitstreamFormat` – HARDCODED Support level text (Used to restore proper support level values to bitstreams)
- `org.dspace.content.Site` – new `getURL()` method, used to store Site URL info in Site-level AIP
- ~~`org.dspace.content.WorkspaceItem` – add ability to create a `WorkspaceItem` with a specified Handle~~
- `org.dspace.content.Community` – add ability to create a Community with a specified Handle, also create a SubCommunity or Collection with specified Handle
- `org.dspace.content.InstallItem` – now supports installing an item with a specified Handle, new `restoreItem()` method (to restore a pre-existing item).
- `org.dspace.content.Collection` – add ability to create a Collection with a specified Handle. Unbind a handle when a Collection is removed.
- `org.dspace.content.Item` – now calls `HandleManager.unbindHandle()` to unbind a handle when an Item is removed.

#### org.dspace.content.crosswalk.\*

Full patch : [org\\_dspace\\_content\\_crosswalk.patch](#)

##### NEW CLASSES

- `org.dspace.content.crosswalk.NullIngestionCrosswalk` – can be used to ignore metadata formats which you don't want to crosswalk during ingest (NOT USED BY DEFAULT)
- `org.dspace.content.crosswalk.NullStreamIngestionCrosswalk` – can be used to ignore metadata formats which you don't want to crosswalk during ingest (NOT USED BY DEFAULT)
- `org.dspace.content.crosswalk.CreativeCommonsTextStreamDisseminationCrosswalk` – used to export a text-based CC license during dissemination (NOT USED BY DEFAULT)
- `org.dspace.content.crosswalk.LicenseStreamIngestionCrosswalk` – used to ingest a license from a stream (NOT USED BY DEFAULT)

- `org.dspace.content.crosswalk.CreativeCommonsRDFStreamIngestionCrosswalk` – used to ingest a CC license from RDF (NOT USED BY DEFAULT)
- `org.dspace.content.crosswalk.AIPDIMCrosswalk` – Crosswalk from DIM to AIP. This essentially just calls the XSLTCrosswalk to disseminate all DIM metadata in an XML format, so that it can be saved to the AIP. Also calls the XSLTCrosswalk to ingest DIM from an AIP and restore to an object.
- `org.dspace.content.crosswalk.LicenseStreamDisseminationCrosswalk` – used to export a license (NOT USED BY DEFAULT – Licenses are exported just like Bitstreams in AIPs)
- `org.dspace.content.crosswalk.CreativeCommonsRDFStreamDisseminationCrosswalk` – used to export a CC license to RDF (NOT USED BY DEFAULT)
- `org.dspace.content.crosswalk.AIPTechMDCrosswalk` – Crosswalk all technical metadata to/from an AIP.

#### UPDATED CLASSES

- `org.dspace.content.crosswalk.XSLTIngestionCrosswalk` – several methods are now static, new `ingestDIM` method, which applies any DIM metadata to an existing object. Now supports ingest of Community/Collection objects (in addition to Items)
- `org.dspace.content.crosswalk.XSLTDisseminationCrosswalk` – now supports dissemination of Community or Collection objects (in addition to Items)
- `org.dspace.content.crosswalk.MODSDisseminationCrosswalk` – now supports MODS dissemination of Community or Collection or Site metadata (in addition to Items)
- ~~`org.dspace.content.crosswalk.XSLTCrosswalk` – HARDCODED mapping of Community/Collection DB columns to DC fields (Are these changes NEEDED?? – Tim)~~ Moved these DB column to DC field mappings to `PackageUtils` utility class. They are necessary for Crosswalks when crosswalking Community or Collection info (as normally crosswalks will actually translate the metadata to DC, and then this mapping can ensure that DC metadata gets into the correct Community or Collection DB column).
- `org.dspace.content.crosswalk.METSDisseminationCrosswalk` – small modifications to work with new `PackageDisseminator` interface (which deals with File rather than `OutputStream`)

### org.dspace.content.packager.\*

Full patch : [org\\_dspace\\_content\\_packager.patch](#)

#### NEW CLASSES

- `org.dspace.content.packager.AbstractPackageDisseminator` – handles recursive dissemination, so that each packager need not implement its own version
- `org.dspace.content.packager.AbstractPackageIngester` – handles recursive ingests/replaces, so that each packager need not implement its own version
- `org.dspace.content.packager.DSpaceAIPDisseminator` – to disseminate AIPs
- `org.dspace.content.packager.DSpaceAIPIngester` – to ingest/restore AIPs

#### UPDATED CLASSES

- `org.dspace.content.packager.METSManifest` – some refactoring & added utility methods to help parse METS files (or grab info from METS files).
- `org.dspace.content.packager.PDFPackager` – refactoring to support updates to `PackageIngester` (which now uses `java.io.File` instead of `InputStream`) and `PackageDisseminator` (using `File` instead of `OutputStream`)
- `org.dspace.content.packager.DSpaceMETSDisseminator` – refactoring to support changes to `AbstractMETSDisseminator` and allow for dissemination of Site, Community and Collections
- `org.dspace.content.packager.PackageUtils` – new `findOrCreateBitstreamFormat()`, `findDepositLicense()` methods (used for ingestion). Also new utility methods to create a DSpace object when ingesting.
- `org.dspace.content.packager.PackageDisseminator` – this interface now includes `disseminateAll()` method (to perform recursive export). Also this interface changed to deal with `java.io.File` instead of `OutputStream` (as recursive export needs a File reference so that it can recursively export everything to the same directory).
- `org.dspace.content.packager.AbstractMETSDisseminator` – MAJOR refactoring to support Site, Community & Collection exports (and code cleanup). This also takes on more of the code once in `DSpaceMETSDisseminator`, as it is generic to both that class and `DSpaceAIPDisseminator`.
- `org.dspace.content.packager.PackageParameters` – Added utility methods to determine/set the normal parameters for a packager call (this is to better support calling any Packager via API calls – rather than relying on CLI only)
- `org.dspace.content.packager.DSpaceMETSIngester` – refactoring to work with Site, Community, Collections & to support `AbstractMETSIngester`
- `org.dspace.content.packager.PackageIngester` – this interface now includes `ingestAll()` and `replaceAll()` methods (to perform recursive ingest/replace). Also this interface changed to deal with `java.io.File` instead of `InputStream` (as Files are easier to deal with when using Zip packages). Also now returns `DSpaceObject` instead of `WorkspaceItem` (so that it supports ingestion of Site/Community/Collection packages).
- `org.dspace.content.packager.AbstractMETSIngester` – MAJOR refactoring to support Site, Community & Collection imports (and code cleanup)

### org.dspace.handle.\*

Full patch : [org\\_dspace\\_handle.patch](#)

- `org.dspace.handle.HandleManager` – `unbindHandle()` method (to officially unbind handles when objects are deleted), lookup Site Object by Handle suffix "0", `getHandleInternal` now returns `TableRow`

### org.dspace.workflow.\*

- `org.dspace.workflow.WorkflowItem` – added a new `findByItem(Context, Item)` method. This allows you to check if an Item is currently under workflow approval.

## dspace-lni-api

Full patch : [dspace-lni.patch](#)

All updates are minor refactoring to support new PackageIngester and PackageDisseminator interfaces (which both now use java.io.File)

- `org.dspace.app.dav.DAVCollection`
- `org.dspace.app.dav.DAVItem`

## dspace-sword-api

Full patch : [dspace-sword.patch](#)

All updates are minor refactoring to support new PackageIngester (which uses File instead of InputStream)

- `org.dspace.sword.CollectionDepositor`
- `org.dspace.sword.ItemDepositor`
- `org.dspace.sword.SWORDMETSIngester`
- `org.dspace.sword.SimpleFileIngester`
- `org.purl.sword.server.DummyServer`
- `org.purl.sword.server.DepositServlet`
- `org.purl.sword.base.Deposit`