DSpace Service based api: API Changelist

- administrative package
  - DCType (Deleted)
- app.bulkedit package
  - BulkEditChange
  - MetadataImport
- app.itemexport package
  - ItemExport refactored into: ItemExportService
- app.itemimport package
  - ItemImport refactored into: ItemImportService
- app.itemupdate package
  - ItemArchive
- app.util package
  - AbstractDSpaceWebapp
  - MetadataExposure refactored into: MetadataExposureService
  - OpenSearch refactored into: OpenSearchService
  - SyndicationFeed
  - Util
  - MetadataUtilities
- app.mediafilter package
  - MediaFilterManager refactored into: MediaFilterService
  - FormatFilter
  - ImageMagickThumbnailFilter
  - MediaFilter
- app.sfx package
  - SFXFileReader refactored into: SFXFileReaderService
- app.statistics package
  - LogAnalyser
- authenticate package
  - AuthenticationManager refactored into: AuthenticationService
  - AuthenticationMethod
  - PasswordAuthentication
  - ShibAuthentication
- authority package
  - AuthorityValue
  - AuthorityValueGenerator & AuthorityValueFinder refactored into: AuthorityValueService
  - AuthorityIndexClient
- authorize Package
  - AuthorizeManager refactored into: AuthorizeService
  - PolicySet
- Browse package
  - General
    - BrowseItem (Deleted)
    - BrowseItemDAO (Deleted)
    - BrowseInfo
    - BrowseCreateDAO
    - BrowseDAO
    - BrowseDAOFactory
    - BrowseDAOUtils (Deleted)
    - BrowseDAOUtilsDefault (Deleted)
    - IndexBrowse
    - SolrBrowseDAO
- Checker package
  - BitstreamDAO (Deleted)
  - BitstreamDispatcher
  - BitstreamInfoDAO (Deleted)
  - ChecksumCheckResults refactored into: ChecksumResultCode
  - ChecksumHistoryDAO
  - DAOsSupport (Deleted)
  - ChecksumResult (New)
  - DSpaceBitstreamInfo
  - ChecksumResultDAO refactored into: ChecksumResultService
  - ReporterDAO (Deleted)
  - BitstreamInfo refactored into: MostRecentChecksum
  - CheckerCommand
  - ChecksumHistory
  - ChecksumResultsCollector
  - HandleDispatcher
  - ListDispatcher refactored into: IteratorDispatcher
  - LimitedCountDispatcher
  - LimitedDurationDispatcher
  - ResultsLogger
  - ResultsPruner
  - SimpleReporter refactored into: SimpleReporterService
SimpleReporterImpl refactored into: SimpleReporterService
SimpleDispatcher

content package
Bitstream refactored into: Bitstream, BitstreamService
Bundle refactored into: Bundle, BundleService
BitstreamFormat refactored into: BitstreamFormat, BitstreamFormatService
Collection refactored into: Collection, CollectionService
Community refactored into: Community, CommunityService
DSpaceObject refactored into: DSpaceObjectService
FormatIdentifier (Deleted)
InProgressSubmission refactored into: InProgressSubmission, InProgressSubmissionService
InstallItem refactored into: InstallItemService
Item refactored into: Item, ItemService
ItemDAO
ItemDAOFactory (Deleted)
ItemDAOOracle (Deleted)
ItemIterator (Deleted)
ItemService
MetadataAuthorityManager refactored into: MetadataAuthorityService
MetadataAuthority (Deleted)
MetadataField refactored into: MetadataField, MetadataFieldService
MetadataSchema refactored into: MetadataSchema, MetadataSchemaService
MetadataValue refactored into: MetadataValue, MetadataValueService
SupervisedItem refactored into: SupervisedItemService
Site refactored into: Site, SiteService
WorkspaceItem refactored into: WorkspaceItem, WorkspaceItemService

content.authority package
ChoiceAuthorityManager refactored into: ChoiceAuthorityService
ChoiceAuthority
SHERPARoMEOProtocol

content.crosswalk package
AIPDIMCrosswalk
AIPTechMDCrosswalk
DIMDisseminationCrosswalk
DisseminationCrosswalk
IngestionCrosswalk
METSDisseminationCrosswalk
METSRightsCrosswalk
MODSDisseminationCrosswalk
OREDisseminationCrosswalk
PREMISCrosswalk
QDCCrosswalk
RoleCrosswalk
SimpleDCDisseminationCrosswalk
XHTMLHeadDisseminationCrosswalk
XSLTDisseminationCrosswalk

content.packager package
AbstractMETSDissemintor
PackageUtils

core package
Context
LicenseManager refactored into: LicenseService
NewsManager refactored into: NewsService

curate package
Curator
CurationCli
WorkflowCurator refactored into: WorkflowCuratorService

disseminate package
CitationDocument refactored into: CitationDocumentService

embargo package
EmbargoManager refactored into: EmbargoService, EmbargoCLITool

eperson package
Group refactored into: Group, GroupService, GroupCLITool
Subscribe refactored into: SubscribeService, SubscribeCLITool
Supervisor refactored into: SupervisorService
AccountManager refactored into: AccountService

event package
Event

handle package
HandleManager refactored into: HandleService
UpdateHandlePrefix

harvest package
HarvestedItem refactored into: HarvestedItem, HarvestedItemService
HarvestedCollection refactored into: HarvestedCollection, HarvestedCollectionService
OAIHarvester

identifier package
• DOI refactored into DOI, DOIService
  • DOIdentifierProvider
  • DOIOrganiser
  • IdentifierService
• license package
  • CreativeCommons refactored into: CreativeCommonsService
• rdf package
  • RDFUtil
  • rdf_conversion package
    • SimpleDSORelationsConverterPlugin
• rdf_storage package
  • DOIURIGenerator
  • HandleURIGenerator
  • LocalURIGenerator
  • URIGenerator
• statistics package
  • ElasticSearchLogger refactored into: ElasticSearchLoggerService
  • SolrLogger refactored into: SolrLoggerService
• storage_bitstore package
  • BitstreamStorageManager refactored into BitstreamStorageService
• requestitem package
  • RequestItem
• version package
  • Version
  • VersionDAO
  • VersionHistory
  • VersionHistoryDAO
  • VersionHistoryImpl refactored into: VersionHistoryService
  • VersionImpl, VersioningService, VersioningServiceImpl refactored into: VersioningService
• workflow package
  • WorkflowItem refactored into: Interface (old usage moved to: BasicWorkflowItem)
  • WorkflowManager refactored into: WorkflowService (old usage moved to BasicWorkflowService)
• workflow_basic package (new)
  • TaskListItem, TaskListItemService
  • BasicWorkflowItem (previously WorkflowItem)
  • BasicWorkflowService (previously WorkflowManager)
• xmlWorkflow package
  • ClaimedTask refactored into: ClaimedTask, ClaimedTaskService
  • CollectionRole refactored into: CollectionRole, CollectionRoleService
  • InProgressUser refactored into: InProgressUser InProgressUserService
  • RoleMembers
  • WorkflowItemRole refactored into: WorkflowItemRole, WorkflowItemRoleService
  • WorkflowRequirementsManager refactored into: WorkflowRequirementsService
  • WorkflowUtils
  • XmlWorkflowItem refactored into: XmlWorkflowItem, XmlWorkflowItemService
  • WorkflowFactory refactored into: XmlWorkflowFactory
  • XmlWorkflowManager refactored into: XmlWorkflowService

administrative package

DCTYPE (Deleted)
Class has been removed, isn’t used anywhere in the codebase

app.bulkedit package

BulkEditChange

• registerRemove, registerAdd, registerConstant, getAdds, getRemoves, getConstant, getComplete, now require a newly created BulkEditMetadataValue instead of a Metadatum (values retrieved from a csv cannot be transformed into a MetadataValue anymore)

MetadataImport

• getAuthorityValueType() method moved to AuthorityValueService

app.itemexport package

ItemExport refactored into: ItemExportService
Factory usage: ItemExportServiceFactory.getInstance().getItemExportService()

- Class has been split into ItemExportService & ItemExportCLITool
- exportAsZip now requires an Iterator<Item> instead of an ItemIterator
- getExportDownloadDirectory, deleteOldExportArchives methods now require an EPerson instead of an identifier
- getExportFileSize & getExportFileLastModified now require a context

app.itemimport package

ItemImport refactored into: ItemImportService

Factory usage: ItemImportServiceFactory.getInstance().getItemImportService()

- Class has been split into ItemImportService & ItemImportCLITool
- addItemAtomic(), addItem(), require a collection list instead of an array
- processUIImport() methods now require a template boolean
- now has setters/getters to control if service should be used in test/resume, useWorkflow,

app.itemupdate package

ItemArchive

- addUndoDeleteContents method now requires a UUID.

app.util package

AbstractDSpaceWebapp

- The class remains but all code related to the WebApp database table has been moved to the respective WebApp, WebAppService, WebAppDAO classes.
- getApps method has been moved to WebAppService and now returns a list of WebApp objects

MetadataExposure refactored into: MetadataExposureService

Factory usage: UtilServiceFactory.getInstance().getMetadataExposureService();

- Has been refactored into MetadataExposureService

OpenSearch refactored into: OpenSearchService

Factory usage: UtilServiceFactory.getInstance().getOpenSearchService();

- Has been refactored into OpenSearchService
- getResultsString(), getResultsDoc() methods now require a context

SyndicationFeed

- populate() method now requires a context.

Util

- getControlledVocabulariesDisplayValueLocalized now requires a list of MetadataValues instead of an array of Metadatum

MetadataUtilities

- deleteMetadataByValue(), appendMetadata() methods now require a context
- getDCValueString now requires a metadata value
- readDeleteContentsFile() now returns a list of strings, this is due to the fact that exported files can have both legacy integers as well as UUIDS.

app.mediafilter package
MediaFilterManager refactored into: MediaFilterService
Factory usage: MediaFilterServiceFactory.getInstance().getMediaFilterService()
  - Class been split into MediaFilterService & MediaFilterCLI Tool

FormatFilter
  - getDestinationStream(), preProcessBitstream() now require verbose boolean
  - getDestinationStream() now requires an item

ImageMagickThumbnailFilter
  - getThumbnailFile(), getImageFile(), now requires verbose boolean

MediaFilter
  - preProcessBitstream() now requires verbose boolean

app.sfx package

SFXFileReader refactored into: SFXFileReaderService
Factory usage: SfxServiceFactory.getInstance().getSfxFileReaderService()
  - SFXFileReader migrated to SFXFileReaderService

app.statistics package

LogAnalyser
  - Now uses a discovery query to retrieve items by accessioned date instead of a complex query.

authenticate package

AuthenticationManager refactored into: AuthenticationService
Factory usage: AuthenticationServiceFactory.getInstance().getAuthenticationService()
  - Migrated to: AuthenticationService
  - getSpecialGroups() method now returns a list of groups instead of array of identifiers

AuthenticationMethod
  - getSpecialGroups() method now returns a list of groups instead of array of identifiers

PasswordAuthentication
  - getSpecialGroups() method now returns a list of groups instead of array of identifiers

ShibAuthentication
  - getSpecialGroups() method now returns a list of groups instead of array of identifiers

authority package

AuthorityValue
  - getAuthorityTypes() method moved to AuthorityServiceFactory.getInstance().getAuthorityTypes();
  - fromSolr() method moved to AuthorityValueService.
• updateItem() method now requires context

AuthorityValueGenerator & AuthorityValueFinder refactored into: AuthorityValueService
Factory usage: AuthorityServiceFactory.getInstance().getAuthorityValueService();
• Migrated into AuthorityValueService

AuthorityIndexClient
• indexItem() method moved to AuthorityService

authorize Package

AuthorizeManager refactored into: AuthorizeService
Factory usage: AuthorizeServiceFactory.getInstance().getAuthorizeService()
• getAuthorizedGroups now returns a list instead of an array of groups
• isAnIdenticalPolicyAlreadyInPlace now requires a group as argument instead of a group identifier
• removeGroupPolicies(Context c, int groupID) now requires a group instead of a group identifier
• getAuthorizedGroups now returns a list instead of an array of groups
• findByTypeIdGroupAction(Context c, int dsoType, int dsoID, int groupID, int action, int policyID) has been changed to use Group & DSpaceObject instead of integers.
• createOrModifyPolicy groupID parameter has been adjusted to use the group object

PolicySet
• setPolicies() methods now require UUIDs instead of integers

Factory usage: AuthorizeServiceFactory.getInstance().getResourcePolicyService()
• setResource() renamed to setdSpaceObject()
• update(context) now also requires an object to update
• delete(context) now also requires an object to delete

Browse package

General
Removed the following classes since this isn’t supported any more:
• BrowseCreateDAOPostgres
• BrowseCreateDAOOracle
• BrowseDAOUtilsOracle
• BrowseDAOUtilsPostgres
• BrowseItemDAOOracle
• BrowseItemDAOPostgres
• ItemCountDAOOracle
• ItemCountDAOPostgres
• BrowseDAOOracle
• BrowseDAOPostgres

BrowseItem (Deleted)
• Class has been removed, uses Item/ItemService now

BrowseItemDAO (Deleted)
• Class has been removed, use ItemService now

BrowseInfo
• getBrowseItemResults() now returns list of items instead of array of BrowseItem objects
- `getItemResults()` method removed, use `getBrowseItemResults()` instead

**BrowseCreateDAO**

- `deleteByItemID()`, `deleteCommunityMappings()`, `updateCommunityMappings()`, `insertIndex()`, `updateIndex()`, `updateDistinctMappings()`, `deleteMappingsByItemID()`, now require a UUID instead of an integer

**BrowseDAO**

- `getContainerID()`/`setContainerID()` methods now require/return a UUID instead of an integer
- `oQuery()` now returns items instead of browseItems

**BrowseDAOFactory**

- `getItemInstance()` method has been removed, we are now using `itemService` directly
- `getUtils()` method removed, wasn’t used.

**BrowseDAOUtils (Deleted)**

- Class has been removed (wasn’t used anymore)

**BrowseDAOUtilsDefault (Deleted)**

- Class has been removed (wasn’t used anymore)

**IndexBrowse**

- `itemRemoved()` method now requires a UUID instead of integer
- `ItemMetadataProxy` inner class has been removed, use `Item` class instead

**SolrBrowseDAO**

- `getContainerID()`/`setContainerID()` methods now require/return a UUID instead of an integer
- `deleteByItemID()`, `deleteCommunityMappings()`, `updateCommunityMappings()`, `insertIndex()`, `updateIndex()`, `updateDistinctMappings()`, `deleteMappingsByItemID()`, now require a UUID instead of an integer

**Checker package**

**BitstreamDAO (Deleted)**

- Class has been removed, use `bitstreamService.retrieve` to get an input stream for a bitstream

**BitstreamDispatcher**

- `next()` method uses bitstreams instead of integers

**BitstreamInfoDAO (Deleted)**

- Uses combo of `bitstreamService` & `ChecksumHistoryService` to perform queries

**ChecksumCheckResults refactored into: ChecksumResultCode**

- `ChecksumCheckResults` has been renamed to `ChecksumResultCode` and no longer is a class but an enum

**ChecksumHistoryDAO**

- `ChecksumHistoryDAO` has been moved from `org.dspace.checker.ChecksumHistoryDAO` to `org.dspace.checker.dao.impl.ChecksumHistoryDAOImpl` & code has been made hibernate compatible

**DAOSupport (Deleted)**
• Class removed, relied on the old Database code

ChecksumResult (New)
• ChecksumResult class created to represent a “checksum_results” table row

DSpaceBitstreamInfo
• Class has been removed, was a value object for Bitstream, uses bitstream now

ChecksumResultDAO refactored into: ChecksumResultService

ReporterDAO (Deleted)
• Class removed, use MostRecentChecksumService & ChecksumHistoryService

BitstreamInfo refactored into: MostRecentChecksum
• Has been refactored into “MostRecentChecksum” class

CheckerCommand
• Constructor now requires a context

ChecksumHistory
• getBitstreamId returns a UUID
• getResult/setResult now require/return a ChecksumResult object instead of a string

ChecksumResultsCollector
• collect() method now requires a context

HandleDispatcher
• next() method returns a bitstream instead of integer
• Constructor now uses context & handle instead of BitstreamInfoDAO & handle.

ListDispatcher refactored into: IteratorDispatcher
• Renamed to IteratorDispatcher
• Now uses an iterator instead of a list for a performance boost

LimitedCountDispatcher
• next() method returns a bitstream instead of integer

LimitedDurationDispatcher
• next() method returns a bitstream instead of integer

ResultsLogger
• collect() now uses MostRecentChecksum instead of BitstreamInfo

ResultsPruner
• getDefaultPruner(), getPruner(), constructor now require a context
• addInterested() methods require a ChecksumResultCode object instead of string representation of this object.

SimpleReporter refactored into: SimpleReporterService
Factory usage: CheckerServiceFactory.getInstance().getSimpleReporterService();

- getDeletedBitstreamReport(), getChangedChecksumReport(), getBitstreamNotFoundReport(), getNotToBeProcessedReport(),
  getUncheckedBitstreamsReport() methods all require a context.

SimpleReporterImpl refactored into: SimpleReporterService

Factory usage: CheckerServiceFactory.getInstance().getSimpleReporterService()

SimpleDispatcher

- Constructor now requires a context instead of BitstreamInfoDAO object
- next() method now returns bitstream instead of integer

content package

Bitstream refactored into: Bitstream, BitstreamService

Factory usage: ContentServiceFactory.getInstance().getBitstreamService();

- findAll, getBundles, now return a list instead of an array
- create method has become public and can be called with our without a bundle. When called without a bundle the class calling on this method is responsible for linking it to the proper object (logo for community for example).
- register method is now public and requires a bundle parameter
- setUserFormatDescription(), setFormat(), setName(), setSource(), setDescription() now require a context
- delete, isDeleted, setDeleted, setInternalId methods are now public
- delete, update, retrieve, getFormat(), getFormatDescription() now require a context & a bitstream
- isRegisteredBitstream now requires a bitstream

Bundle refactored into: Bundle, BundleService

Factory usage: ContentServiceFactory.getInstance().getBundleService();

- create method has become public and must be called with an item.
- getBitstreamByName now requires a bundle
- createBitstream method has been removed from bundle, use bitstreamService.create() instead
- registerBitstream method has been removed from bundle, use bitstreamService.registerBitstream() instead
- addBitstream, inheritCollectionDefaultPolicies, replaceAllBitstreamPolicies, getBitstreamPolicies, update, delete, remove methods require a context and a bundle
- getBundlePolicies has been removed, use authorizeService.getPolicies
- delete is now public
- setOrder now requires a context a bundle & a list of UUIDS instead of integers
- getPrimaryBitstreamID & setPrimaryBitstreamID now renamed to getPrimaryBitstream & setPrimaryBitstream and require a bitstream
- setName(),getBundlePolicies() now require a context

BitstreamFormat refactored into: BitstreamFormat, BitstreamFormatService

Factory usage: ContentServiceFactory.getInstance().getBitstreamFormatService();

- findAll, findNonInternal now return a list instead of an array.
- setShortDescription is set on BitstreamFormatService & requires a context and the bitstreamformat alongside the description.
- getSupportLevelText, setSupportLevel now require a bitstreamformat since they have been moved to the service.
- update, delete now require a context and the bitstream format.
- getExtensions & setExtensions now return/require a list instead of an array

Collection refactored into: Collection, CollectionService

Factory usage: ContentServiceFactory.getInstance().getCollectionService();

- create methods have been made public
- findAll(), findAuthorized(),getCommunities(),findDirectMapped(), findGroup2CommunityMapped(),findGroup2GroupMapped(),findGroupMapped() methods now return a list instead of an array
- getItems() methods have been moved to the ItemService and has been renamed to findByCollection()
- getAllItems() method has been moved to the ItemService and has been renamed to findAllByCollection()
**Community refactored into: Community, CommunityService**

**Factory usage:** ContentServiceFactory.getInstance().getCommunity();

- findAll, findAllTop now return list instead of array
- getMetadata method now requires community as param
- setMetadata, setLogo, createAdministrators, removeAdministrators, getAParents, getACollections, addCollection, createSubcommunity, addSubcommunity, removeCollection, removeSubcommunity, delete, canEditBoolean methods now require context & community as param
- getCollections, getSubcommunities, getAParents, getACollections now return a list instead of an array
- createCollection has moved to the collection service
- getParentCommunity has been renamed to getParentCommunities now returns a list of communities since it is possible in the code to have multiple parent communities
- countItems now requires a context

**DSpaceObject refactored into: DSpaceObjectService**

**Factory usage:** ContentServiceFactory.getInstance().getDSpaceLegacyObjectService(type)

- getIdentifiers now returns a list instead of an array
- getParentObject now requires a context & the dspaceObject you want the parent for
- updateLastModified now requires context & the dspaceObject you want to update
- deprecated getDC, addDC methods have been removed
- getMetadata methods now return a list of metadata values instead of an array of Metadatum
- getElements(String fieldName) & getElementsFilled(String fieldName) have been made protected, wasn't used outside of the object
- addMetadata, getMetadata & clearMetadata methods now require a dspaceObject as a parameter
- addMetadata & clear metadata require a context as a parameter
- addMetadata has been altered to use lists instead of arrays to pass repeatable params
- find method has been removed, use: ContentServiceFactory.getInstance().getDSpaceObjectService(type).find(context, id) instead
- finding by legacy id can be done using:
  - ContentServiceFactory.getInstance().getDSpaceLegacyObjectService(type).findByIdOrLegacyId(context, String id)
- resetIdentifiersCache() has been removed, caching is not used, hibernate takes care of the caching for us.
- getAuthoritiesAndConfidences() method now requires a collection.

**FormatIdentifier (Deleted)**

- Class has been removed, uses BitstreamFormatService.guessFormat() instead

**InProgressSubmission refactored into: InProgressSubmission, InProgressSubmissionService**

**Factory usage:** ContentServiceFactory.getInstance().getInProgressSubmissionService(submissionInfo)

- deleteWrapper(), update() methods moved to InProgressSubmissionService

**InstallItem refactored into: InstallItemService**

**Factory usage:** ContentServiceFactory.getInstance().getInstallItemService();

- method getBitstreamProvenanceMessage() now requires a context.

**Item refactored into: Item, ItemService**

**Factory usage:** ContentServiceFactory.getInstance().get_ItemService();

- create is now public & requires a workspaceItem (if the workspace item is linked to another item an exception will be thrown).
- isIn(), getBundles(String name), isOwningCollection(), hasUploadedFiles() require an item
- getCommunities(), addBundle(), removeBundle(), createSingleBitstream(), removeDSpaceLicense(), removeLicenses(), withdraw(), reinstate(), delete(), replaceAllItemPolicies(), replaceAllBitstreamPolicies(), getNonInternalBitstreams(), inheritCollectionDefaultPolicies(), adjustBundleBitstreamPolicies(), adjustItemPolicies(), move(), getCollectionsNotLinked(), canEdit() now require a context and an item
- `getCommunities`, `getBundles`, `getNonInternalBitstreams`, `getCollectionsNotLinked()`, `getCollections()` now return a list instead of an array.
- `createBundle` has been removed, creating of a bundle is now done in the BundleService.
- item deletion is now public.
- `findByMetadataField()`, `findByAuthorityValue()`, `findByMetadataFieldAuthority()` now return an `Iterator<Item>` instead of `ItemIterator`.
- `getThumbnail` now requires an item instead of an itemid.

**ItemDAO**

- Is now an interface instead of a class and has become part of the DAO, model, is no longer public. Use `itemService` instead.

**ItemDAOFactory (Deleted)**

- Class has been removed, DAO's are autowired into the services & no factories are present.

**ItemDAOOracle (Deleted)**

**ItemDAOPostgres (Deleted)**

**ItemIterator (Deleted)**

- You can now use `Iterator<Item>`

**ItemService**

- Has now become an interface
- `getThumbnail` is available in this interface
- `getFirstMetadataValue` removed use

**MetadataAuthorityManager refactored into: MetadataAuthorityService**

**Factory usage**: `ContentAuthorityServiceFactory.getInstance().getMetadataAuthorityService();`

- `isAuthorityControlled` now requires a metadata field object instead of schema, element, qualifier, ...
- `isAuthorityRequired` now requires a metadata field object instead of schema, element, qualifier, ...
- `getMinConfidence` now requires a metadata field object instead of schema, element, qualifier ...

**Metadatum (Deleted)**

- Class has been removed, `MetadataValue` is used.

**MetadataField refactored into: MetadataField, MetadataFieldService**

**Factory usage**: `ContentServiceFactory.getInstance().getMetadataFieldService();`

- `create(Context context)` method has been moved to `MetadataFieldService.create(Context, MetadataSchema, element, qualifier, scopeNote)`
- `findByElement` now requires a metadataSchema OR a metadata schema name instead of an identifier
- `findAllInSchema` now requires a schema instead of the identifier of a schema
- `update(context)` now also requires an object to update
- `delete(context)` now also requires an object to delete
- `formKey` has been removed, use the `toString()` method from a metadataField
- `findAll`, `findAllInSchema` now return lists instead of arrays

**MetadataSchema refactored into: MetadataSchema, MetadataSchemaService**

**Factory usage**: `ContentServiceFactory.getInstance().getMetadataSchemaService();`

- `MetadataSchema.DC_SCHEMA_ID` has been removed, best to use the lookup by name.
- `MetadataSchema(String namespace, String name)` constructor has been moved to `metadataSchemaService.create(Context context, String namespace, String name)`
- `create(Context context)` method has been moved to `metadataSchemaService.create(Context context, String namespace, String name)`
- `update(context)` now also requires an object to update
- `delete(context)` now also requires an object to delete
- `findAll` now returns a list instead of array
MetadataValue refactored into: MetadataValue, MetadataValueService

**Factory usage:** ContentServiceFactory.getInstance().getMetadataValueService();

- `create(Context context)` method has been moved to `metadataValueService.create(Context, DSpaceObject, MetadataField)`
- `findByField(Context context, int fieldId)` now uses a metadata field object instead of its identifier
- `update(context)` now also requires an object to update
- `delete(context)` now also requires an object to delete

SupervisedItem refactored into: SupervisedItemService

**Factory usage:** ContentServiceFactory.getInstance().getSupervisedItemService();

- `getAll()`, `findByEPerson()` methods now return a list of workspaceItem objects instead of supervisedItems
- `getSupervisorGroups()` method has been removed, this is now a getter on WorkspaceItem

Site refactored into: Site, SiteService

**Factory usage:** ContentServiceFactory.getInstance().getSiteService();

- Is no longer a static object, but contains an actual database object.
- `Site.SITE_ID` is now `siteService.findSite(context).getID()`;
- `Site.getSiteHandle()` is now `siteService.findSite(context).getHandle()`;

WorkspaceItem refactored into: WorkspaceItem, WorkspaceItemService

**Factory usage:** ContentServiceFactory.getInstance().getWorkspaceItemService();

- When a workspaceItem is created the rights granted to reviewer groups used in the basic workflow are no longer hardcoded, this section has moved to the workflowService. This way the workflow determines which rights are granted.
- `findByEPerson()`, `findByCollection()`, `findAll()` now return a list instead of an array.
- `update()`, `deleteWrapper()` & `deleteAll()` methods now require a context and a workspace item.

content.authority package

ChoiceAuthorityManager refactored into: ChoiceAuthorityService

**Factory usage:** ContentAuthorityServiceFactory.getInstance().getChoiceAuthorityService();

- `getLabel()`, `getVariants()` methods now requires a metadataValue
- `getManager()` method removed, use the factory to get an instance.
- `getMatches()`, `getBestMatch()` methods now requires a collection object instead of an identifier
- `makeFieldKey()` method has been removed, use `metadataField.toString()`;

ChoiceAuthority

- `getMatches()`, `getBestMatch()` methods now require a collection instead of collection identifier

SHERPARoMEOProtocol

- `getMatches()` method now requires a collection instead of a collection identifier

content.crosswalk package

AIPDIMCrosswalk

- `disseminateList`, `disseminateElement` methods now require a context

AIPTechMDCrosswalk

- `disseminateList`, `disseminateElement` methods now require a context
**DIMDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**DisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**IngestionCrosswalk**
- ingest() method requires additional boolean that indicates if metadataFields should be created prior to adding metadata.

**METSDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**METSRightsCrosswalk**
- disseminateList, disseminateElement methods now require a context

**MODSDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**OREDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**PREMISCrosswalk**
- disseminateList, disseminateElement methods now require a context

**QDCCrosswalk**
- disseminateList, disseminateElement methods now require a context

**RoleCrosswalk**
- disseminateList, disseminateElement methods now require a context

**SimpleDCDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**XHTMLHeadDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context

**XSLTDisseminationCrosswalk**
- disseminateList, disseminateElement methods now require a context
  - createDIM now requires a list of MockMetadataValue object instead of array of metadatum
  - ingestDIM() methods now require an extra boolean createMissingMetadataFields

**content.packager package**

**AbstractMETSDisseminator**
- makeBitstreamURL() method now require a context
PackageUtils

- `updateDSpaceObject()`, `removeAllBitstreams()`, `clearAllMetadata()`, `getBitstreamByFormat()` methods now require a context

**core package**

**Context**

- `clearCache()`, `removeCached()`, `cache()`, `fromCache()`, `getCacheSize()` methods removed, hibernate takes care of our caching
- `getDBConnection()` now returns a list instead of an array of groups
- `getSpecialGroups()` now returns a list instead of an array of groups
- `commit()` method has been removed, hibernate can only commit a transaction ONCE, everything is persisted so there should be no need for the commit method. If the events needs to be flushed, a new `dispatchEvents()` method is available

LicenseManager refactored into: LicenseService

**Factory usage**: CoreServiceFactory.getInstance().getLicenseService();

NewsManager refactored into: NewsService

**Factory usage**: CoreServiceFactory.getInstance().getNewsService();

**curate package**

Curator

- `setCacheLimit()` method removed, no longer required since object cache is no longer used

CurationCli

- limit option removed, no longer required since object cache is no longer used

WorkflowCurator refactored into: WorkflowCuratorService

**Factory usage**: CurateServiceFactory.getInstance().getWorkflowCuratorService();

**disseminate package**

CitationDocument refactored into: CitationDocumentService

**Factory usage**: DisseminateServiceFactory.getInstance().getCitationDocumentService();

- `getOwningCommunity()`, `makeCitedDocument()`, `canGenerateCitationVersion()` methods now require a context object

**embargo package**

EmbargoManager refactored into: EmbargoService, EmbargoCLITool

**Factory Usage**: EmbargoServiceFactory.getInstance().getEmbargoService()

- Main method has moved to “EmbargoCLITool” class

**eperson package**

Group refactored into: Group, GroupService, GroupCLITool
Factory usage: EPersonServiceFactory.getInstance().getGroupService();

- addMember, removeMember, delete, update now require a context & group object parameters
- isMember(EPerson e) has been renamed to isDirectMember & now require a context & group object parameters
- isMember(Group g) has been renamed to isDirectMember & now requires the parent group parameter
- isMember(Context c, int groupId) now requires a group object instead of a groupId
- allMemberGroups, allMembers, findAll, search, getMembers, getMemberGroups now return a list instead of an array
- allMemberGroupIds is now removed, use allMemberGroups instead (returns a list of groups instead of identifiers)
- allMemberIds is now removed, use allMembers instead (returns a list of epeople instead of identifiers)
- isEmpty now requires a group
- setName() now requires a context.
- Group main code moved from “Group” to “GroupCLITool”
- Finding groups sorted by identifier is no longer possible since it doesn’t make sense to sort on a UUID.


- search(), findAll() methods now return a list instead of an array
- setPassword(), setPasswordHash(), getPasswordHash() methods now require an ePerson
- checkPassword(), setMetadata() methods now require a context and an EPerson
- main method has moved to the “EPersonCLITool” class.
- setLanguage(), setFirstName(), setLastName() methods now require a context

Subscribe refactored into: SubscribeService, SubscribeCLITool

Factory Usage: EPersonServiceFactory.getInstance().getSubscribeService();

- Main method has moved to “SubscribeCLITool” class
- getSubscriptions(), now return a list of Subscription objects instead of array of collections
- getAvailableSubscriptions() now return a list instead of an array

Supervisor refactored into: SupervisorService

Factory Usage: EPersonServiceFactory.getInstance().getSupervisorService();

- isOrder(), remove(), add() now require workspaceItem and group objects instead of identifiers
- removeRedundant() method no longer needed, linking table is auto cleared when deleting a workspaceItem/group.

AccountManager refactored into: AccountService

Factory usage: EPersonServiceFactory.getInstance().getAccountService();

event package

Event

- Constructor now requires an ArrayList of identifiers instead of an array (arrayList is used because it is cloneable)
- Constructors now require UUIDs as object id’s instead of integers
- getIdentifiers now returns a list instead of array

EventManager refactored into: EventService

Factory Usage: EventServiceFactory.getInstance().getEventService();

handle package

HandleManager refactored into: HandleService

Factory Usage: HandleServiceFactory.getInstance().getHandleService();
UpdateHandlePrefix

Script has been improved to update metadataValue objects instead of a regular query. The benefit is that the indexes & last modified dates are also updated.

harvest package

HarvestedItem refactored into: HarvestedItem, HarvestedItemService

**Factory usage:** HarvestServiceFactory.getInstance().getHarvestItemService();

- exists method removed, wasn’t used
- find, create methods now require item object instead of item id
- findById now requires collection object instead of collection identifier
- update, delete now require a context & HarvestedItem objects
- getItemId renamed to getItem & now returns item object

HarvestedCollection refactored into: HarvestedCollection, HarvestedCollectionService

**Factory usage:** HarvestServiceFactory.getInstance().getHarvestCollectionService();

- find(), create(), isHarvestable(), isReady() methods now require collection object instead of collection identifier
- findAll, findReady, findStatus, findOldestHarvest, findNewestHarvest, now return HarvestedCollection object(s) instead of integer(s)
- getCollectionId renamed to getCollection() now returns a collection
- setHarvestResult method removed, use setLastHarvested & setHarvestMessage

OAIHarvester

- HarvestScheduler, HarvestThread inner classes have moved to separate classes
- The scheduling logic of the Harvester has been moved to the “HarvestSchedulingService” class
  - **Usage:** HarvestServiceFactory.getInstance().getHarvestSchedulingService()
  - scrubMetadata() internal method that created metadata fields/schema’s has moved to the IngestionCrosswalk class

identifier package

DOI refactored into DOI, DOIService

**Service Usage:** IdentifierServiceFactory.getInstance().getDOIService();

DOIdentifierProvider

- getDOIOutOfObject, getObjectByDOI methods are no longer static, the only static call they had was from the unit test.

DOIOrganiser

- getDOIsByStatus() method removed use DOIService.getDOIsByStatus()
- register, reserve, update, methods now require a DOI instead of a tableRow & DSpaceObject
- findTableRow renamed to resolveToDOI and now returns a DOI instead of tableRow

IdentifierService

- lookup() method now returns a list instead of an array

license package

CreativeCommons refactored into: CreativeCommonsService

**Service Usage:** LicenseServiceFactory.getInstance().getCreativeCommonsService();
- `getCCField()` method now returns a `LicenseMetadataValue` instead of an internal class `MdField`
- `getLicenseURL()`, `getLicenseText()`, `getLicenseRDF()` now require a context

**rdf package**

**RDFUtil**
- `generateIdentifier, delete` methods now use a UUID as dso identifier & a list of identifiers

**rdf.conversion package**

**SimpleDSORelationsConverterPlugin**
- `bitstreamURI` method also requires a context

**rdf.storage package**

**DOIURIGenerator**
- `generateIdentifier` now requires a list of identifiers instead of array and a UUID instead of an integer as DSpaceObject identifier

**HandleURIGenerator**
- `generateIdentifier` now requires a list of identifiers instead of array and a UUID instead of an integer as DSpaceObject identifier

**LocalURIGenerator**
- `generateIdentifier` now requires a list of identifiers instead of array and a UUID instead of an integer as DSpaceObject identifier

**URIGenerator**
- `generateIdentifier` now requires a list of identifiers instead of array and a UUID instead of an integer as DSpaceObject identifier

**statistics package**

**ElasticSearchLogger** refactored into: **ElasticSearchLoggerService**

*Factory usage:* `StatisticsServiceFactory.getInstance().getElasticSearchLoggerService();`
- `getInstance()` method removed, use the factory method from above

**SolrLogger** refactored into: **SolrLoggerService**

*Factory usage:* `StatisticsServiceFactory.getInstance().getSolrLoggerService();`

**storage.bitstore package**

**BitstreamStorageManager** refactored into **BitstreamStorageService**

*Factory Usage:* `StorageServiceFactory.getInstance().getBitstreamStorageService();`
- `store() & register` methods now require a bitstream and returns a UUID
- `retrieve` now requires a bitstream instead of a bitstream identifier
- `delete(context, bitstreamid)` has been removed, use `bitstreamService.delete()` instead.
requestitem package

RequestItem

* Factory usage: RequestItemServiceFactory.getInstance().getRequestItemService()

  - public constructor removed, use factory method from above
  - bitstreamId getter & setters now require bitstream instead of bitstream identifier
  - itemId getter & setters now require item instead of item identifier
  - update method now requires requestItem object
  - getToken moved to the new create method, requires same parameters as the old constructor

version package

Version

* Interface replaced by Database entity

VersionDAO

* Has moved to the DAO sub package and been split into VersionDAO (interface) & VersionDAOImpl (the new DAO implementation)

VersionHistory

* Interface replaced by Database entity

VersionHistoryDAO

* Has moved to the DAO sub package and been split into VersionHistoryDAO (interface) & VersionHistoryDAOImpl (the new DAO implementation)

VersionHistoryImpl refactored into: VersionHistoryService

* Factory usage: DSpaceVersionServiceFactory.getInstance().getVersionHistoryService();

  - getLatestVersion(), getFirstVersion(), getPrevious(), getNext(), hasNext(), add(), getVersion(), isFirstVersion(), isLastVersion(), remove(), now require a VersionHistory object

VersionImpl, VersioningService, VersioningServiceImpl refactored into: VersioningService

* Factory usage: DSpaceVersionServiceFactory.getInstance().getVersioningService();

  - createNewVersion(), findVersionHistory(), updateVersion(), methods require item object instead of item identifier
  - removeVersion(), restoreVersion() methods require a version object instead of identifier

workflow package

WorkflowItem refactored into: Interface (old usage moved to: BasicWorkflowItem)

* Has now become an interface, to use the original WorkflowItem class check BasicWorkflowItem (this is an implementation of workflowItem).
  - findByEPerson() method has been renamed to findBySubmitter & is also available from the WorkflowItemService

WorkflowManager refactored into: WorkflowService (old usage moved to BasicWorkflowService)

* Renamed to WorkflowService and has become an interface, to use the old WorkflowManager look at BasicWorkflowService & BasicWorkflowServiceImpl

workflowbasic package (new)

TaskListItem, TaskListItemService
Factory usage: BasicWorkflowServiceFactory.getInstance().getTaskListItemService();

- A new class “TaskListItem” has been created with a service & dao interface + implentation. Querying this table was done in the old workflowManager, but this has been extracted from it.

**BasicWorkflowItem (previously WorkflowItem)**

Factory usage: WorkflowServiceFactory.getInstance().getWorkflowItemService(); OR BasicWorkflowServiceFactory.getInstance().getBasicWorkflowItemService();

- The old WorkflowItem class now extends WorkflowItem (which in turn extends the InProgressSubmission)
- findAll(), findByCollection(), findByIdentifier() methods now return a list instead of an array
- findByIdentifierEPerson has been renamed to findByIdentifierSubmitter and returns a list instead of an array
- deleteWrapper(), update() now require a context and a BasicWorkflowItem

**BasicWorkflowService (previously WorkflowManager)**

Factory usage: WorkflowServiceFactory.getInstance().getWorkflowService(); OR BasicWorkflowServiceFactory.getInstance().getBasicWorkflowService();

- abort now returns a workspace item (used to be void)
- notifyOfCuration() now requires a list of EPeople instead of an array
- archive() method is now public
- reject has been renamed to sendWorkflowItemBackSubmission and requires a “provenancePrefix” (not used in the basic workflow)

**xmlWorkflow package**

**ClaimedTask refactored into: ClaimedTask, ClaimedTaskService**

Factory usage: XmlWorkflowServiceFactory.getInstance().getClaimedTaskService();

- findByWorkflowId renamed to findByWorkflowItem & now requires an XmlWorkflowItem instead of id
- findByWorkflowIdAndEPerson renamed to findByWorkflowAndEPerson & now requires XmlWorkflowItem & EPerson objects instead of an id
- findByEperson requires an EPerson object
- All "find" methods now require objects instead of identifiers
- delete(), update() now require a context & a claimedTask object
- setOwnerID/getOwnerID renamed to getOwner/setOwner and requires/returns an EPerson
- setWorkflowItemID/getWorkflowItemID renamed to setWorkflowItem/setWorkflowItem and returns/requires an XmlWorkflowItem

**CollectionRole refactored into: CollectionRole, CollectionRoleService**

Factory usage: XmlWorkflowServiceFactory.getInstance().getCollectionRoleService();

- findAll() methods now require a collection object instead of an identifier
- findById() method now returns a list instead of an array
- create method now also requires a collection, roleid & group
- delete(), update() now require a context & a collectionRole object
- setCollectionId/getCollectionId renamed to setCollection/getCollection an requires/returns a collection object
- setGroupId renamed to setGroup

**InProgressUser refactored into: InProgressSubmission, InProgressUserService**

Factory usage: XmlWorkflowServiceFactory.getInstance().getInProgressUserService();

- findByWorkflowItemAndEPerson now requires an XmlWorkflow item & an EPerson object instead of two identifiers
- findByEperson requires an EPerson object instead of an identifier
- update() now require a context & an InProgressUserService
- setUserID/getUserID renamed to setUser/getUser and now require/return an EPerson object
- setWorkflowItemID/getWorkflowItemID renamed to setWorkflowItem/setWorkflowItem and now require/return an XmlWorkflowItem object
- setGroupID/getGroupID renamed to setGroup/getGroup and now require/return a Group object
**setWorkflowItemID/getWorkflowItemID** renamed to **setWorkflowItem/setWorkflowItem** and now require/return an **XmlWorkflowItem** object

**RoleMembers**
- removeEperson now requires an **EPerson** object instead of an identifier

**WorkflowItemRole** refactored into: **WorkflowItemRole, WorkflowItemRoleService**

**Factory usage:** `XmlWorkflowServiceFactory.getInstance().getWorkflowItemRoleService()`
- **find(), findAllForItem()** methods now require an **XmlWorkflowItem** instead of an identifier & returns a list instead of an array.
- **delete(), update()** now require a context & a **WorkflowItemRole** object
- **setWorkflowItemID/getWorkflowItemID** renamed to **setWorkflowItem/setWorkflowItem** and now require/return an **XmlWorkflowItem** object

**WorkflowRequirementsManager** refactored into: **WorkflowRequirementsService**

**Factory usage:** `XmlWorkflowServiceFactory.getInstance().getWorkflowRequirementsService();`

**WorkflowUtils**
- **createCollectionWorkflowRole(), getRoleGroup()** methods now require a collection instead of an identifier

**XmlWorkflowItem** refactored into: **XmlWorkflowItem, XmlWorkflowItemService**

**Factory usage:** `WorkflowServiceFactory.getInstance().getWorkflowItemService(); OR XmlWorkflowServiceFactory.getInstance().getXmlWorkflowItemService();`
- **create** now requires an item & a collection.
- **findAll(), findByCollection** methods now return a list instead of an array
- **findAllInCollection()** method now returns a list instead of an array and requires a collection object instead of an identifier
- **countAllInCollection()** method now request a collection instead of an identifier
- **findByEperson** has been renamed to **findBySubmitter** and returns a list instead of an array
- **deleteWrapper(), update()** now require a context and an **XmlWorkflowItem**

**WorkflowFactory** refactored into: **XmlWorkflowFactory**

**Factory Usage:** `XmlWorkflowServiceFactory.getInstance().getWorkflowFactory()`
- **XmlWorkflowFactory** has been moved to the factory sub package and is now an interface

**XmlWorkflowManager** refactored into: **XmlWorkflowService**

**Factory usage:** `WorkflowServiceFactory.getInstance().getWorkflowService(); OR BasicWorkflowServiceFactory.getInstance().getXmlWorkflowService();`