**CONFIDENTIAL AND PROPRIETARY**

**Aries Systems Corporation**



**Technical Specification**

13.0-03 v2

checkCIF Integration

**Status: DRAFT**

Copyright © 2015, Aries Systems Corporation

This document is the confidential and proprietary property of Aries Systems Corporation. The contents hereof are for use only by recipients specifically identified by Aries, and may not be further disclosed to any other third party. Specifications, designs and drawings contained herein are subject to withdrawal or change by Aries, without notice. Aries shall have no liability resulting from such withdrawal or change. Editorial Manager is a registered trademark and the property of Aries Systems Corporation.

Aries Systems reserves the right to share this specification with other individuals or organizations that have entered into confidentiality agreements with Aries Systems, in which case the specification will be appropriately blinded so that no identifying or proprietary information will be exposed.

Contributors: 9

Document Revision History: 9

Summary: 9

References: 9

Design: 10

Spec Support 10

Data 10

+ CIF\_RESULT\_FILES (table) 10

+ TRG\_CIF\_RESULT\_FILES\_maintainFileCatalogueReferenceCount(trigger on CIF\_RESULT\_FILES) 10

+ CIF\_CHECK\_STATUS (table) 11

usp\_GetCIFResultFiles 11

Code 13

+ cifCheckFunctions.js 13

docFilesUtil.js 13

+ Aries.EditorialManager.Framework.Journal.FileObjects.CIFResultFile 13

+ Aries.EditorialManager.Framework.Journal.CIFObjects.CIFCheckStatus 14

Aries.EditorialManager.Framework.Journal.FileObjects.CompanionFile 14

Aries.EditorialManager.Framework.Journal.FileObjects.SubmissionFile 14

Action Links (2.13.1) 15

Data 15

usp\_SelectSearchProposalResults (used only in results from proposal search) 15

Code 15

Aries.EditorialManager.ActionLinkManager 15

ActionLinkManager.cs 15

Aries.EditorialManager.Facade.Utility 16

ActionLinkManager.cs 16

Aries.EditorialManager.Framework.Journal.LayoutItems 16

SectionItemDetails.cs 16

Folders.SearchSubmissions(2.8) 16

Web\common\App\_Code\Folders\SearchResults\Controller.cs 16

Folders.AllAssignedSubmissions (2.13.1) 16

Web\common\App\_Code\Folders\AllAssignedSubmissions\Controller.cs 16

Folders.NewSubmissions (2.13.1) 16

Web\common\App\_Code\Folders\NewSubmissions\Controller.cs 16

Folders.RevisedSubmissions (2.13.1) 17

Web\common\App\_Code\Folders\RevisedSubmissions\Controller.cs 17

Folders.RevisedSubmissionsReqAssign (2.13.1) 17

Web\common\App\_Code\Folders\RevisedSubmissionsReqAssign\Controller.cs 17

Pages 18

EMDetails.aspx/cs (2.5) 18

TechnicalCheck.aspx/cs (2.6) 19

downloadSubFilesList.asp (2.7.2) 21

editor\_invitation.asp (2.13.1) 21

editor\_new.asp (2.13.1) 21

editorSubByAssign.asp (2.13.1) 21

editorSubByCurEd.asp (2.13.1) 22

editorSubByStatus.asp (2.13.1) 22

editorOutForRevision.asp (2.13.1) 22

editor\_assign\_with\_fd.asp (2.13.1) 22

editor\_docs\_with\_fd.asp (2.13.1) 22

productionInitiatedNoTasksAssigned.asp (2.13.1) 22

viewAllSubmissionsInProduction.asp (2.13.1) 22

publisherAcceptedManuscripts.asp (2.13.1) 22

Archiving 23

Data 23

Code 23

Asynch 24

Data 24

CIF\_FILE\_TRAFFIC\_LOG (new) 24

usp\_ProcessCIFFileResultRequest (new) 24

REQUEST\_TYPES (new record) 24

REQUEST\_PRIORITIES (new records) 24

MASTER\_CONFIG (New Records) 25

URL 25

Timeout 25

User Name 25

Password 25

Code 26

Asynch Executable 26

AsynchObject\_CIF\_Check.cpp/h, 26

ServicesHelper.cpp/h – 26

Core/Source/Wrappers/ServicesHelperWrapper.cpp 26

+ Aries.EditorialManager.CIFChecker 27

+ CIFCheckTask.cs 27

Aries.EditorialManager.TransferRecorders 27

+ CIFTransferRecorder.cs 27

+ Aries.EditorialManager.AsynchServices.CIFCheck 27

+ CIFCheck.cs 27

+ CIFCheckProcessingRequest.cs 27

+ CIFCheckRequestingApplication.cs 27

Aries.EditorialManager.AsynchServices 27

RequestType.cs 27

ProcessingRequest.cs 27

RequestSubmitter.cs 27

Aries.EditorialManager 29

SiteManagement.cs 29

CIF Results (2.13.3) 30

Pages 30

Web/Common/CIFResults.aspx/cs 30

Configuration 31

Code 31

Aries.EditorialManager.Framework.Journal.ConfigObjects.ConfigItem 31

CheckCIFSettings.cs (new) 31

ConfigItemEnum.cs (additions to enum) 31

Aries.EditorialManager.Framework.Journal.ConfigObjects.CustomInstructions 31

CustomInstructionGroupType.cs (additions to enum) 31

CustomInstructionObjectType.cs (additions to enum) 31

Aries.EditorialManager.Facade.Utility 31

HTMLUtilityCIFCheck.cs (new) 31

Pages 32

CheckCIFSettings.aspx.cs (new) (2.3.1.1) 32

Data 32

MASTER\_CONFIG (New Records) 32

ADMINACCESSFUNCTIONS (Policy Manager - New Records) 34

CUSTOM\_INSTRUCTIONS (New Records) 35

Details Page Layouts 36

Data 36

DETAILS\_PAGE\_LAYOUT\_ITEMS (new record) 36

EMIA (2.17) 37

Pages 37

qdStats72.asp 37

newSubs72.asp 37

Manager 38

Code 38

+ Aries.EditorialManager.CIFChecker 38

CIFCheckManager.cs (new) 38

+ Aries.EditorialManager.Facade.BusinessLayer 39

CIFManager.cs (new) 39

Merge Fields (7) 40

Data 40

Code 40

Aries.EditorialManager.Framework.Journal.DeepLinkObjects 40

DeepLinkType.cs enum 40

Aries.EditorialManager.LetterHandling.MailMerge 40

MergeRuleset.cs 40

Aries.EditorialManager.LetterHandling.MailMerge.MergeCodes.DeepLinks. 40

+ ViewCIFResultsPDFDeepLink.cs 40

Pages 40

Web/common/linkval.asp 40

Code 41

AsynchObject\_PDFBuild\_Process.cpp: 41

Web/Common/docFilesUtil.js 41

Web/Common/AsynchFunctions.js 41

Data 41

prGetPDFBuildFileList 41

Pages 42

pdfBuildSetup.asp 42

Permissions 43

Data 43

AUTHORROLE 43

EDITORROLE 43

PUBLISHERROLE 43

REVIEWERROLE 43

usp\_InsertAuthorRoleSettings 43

usp\_UpdateAuthorRoleSettings 43

usp\_SelectAuthorRoleSettings 43

usp\_InsertEditorRoleSettings 43

usp\_UpdateEditorRoleSettings 43

usp\_SelectEditorRoleSettings 44

usp\_SelectAllEditorRoleSettings 44

usp\_InsertPublisherRoleSettings 44

usp\_UpdatePublisherRoleSettings 44

usp\_SelectPublisherRoleSettings 44

usp\_SelectAllPublisherRoleSettings 44

usp\_InsertReviewerRoleSettings 44

usp\_UpdateReviewerRoleSettings 44

usp\_SelectReviewerRoleSettings 44

usp\_SelectAllReviewerRoleSettings 44

Code 45

Aries.EditorialManager.Framework.Journal.RoleManager 45

PermissionEnum.cs 45

IEditorPublisherRolePermissions.cs 45

AbstractRolePermissionsEntity.cs 45

Aries.EditorialManager.Framework.Journal.RoleManager.AuthorRolePermissions. Database 45

Mapper.cs 45

Aries.EditorialManager.Framework.Journal.RoleManager.EditorRolePermissions 46

Entity.cs 46

Aries.EditorialManager.Framework.Journal.RoleManager.EditorRolePermissions. Database 46

Mapper.cs 46

Aries.EditorialManager.Framework.Journal.RoleManager.PublisherRolePermissions 46

Entity.cs 46

Aries.EditorialManager.Framework.Journal.RoleManager.PublisherRolePermissions. Database 46

Mapper.cs 46

Aries.EditorialManager.Framework.Journal.RoleManager.ReviewerRolePermissions. Database 46

Mapper.cs 46

Site Cleanout(9) 47

File Statistics 48

Pages 48

Web/Common/Admin/getFileStatistics.asp 48

Submission Item Type 49

Data 49

DFILETYPE 49

usp\_InsertSubmissionItemType 49

usp\_UpdateSubmissionItemType 49

usp\_SelectAllSubmissionItemTypes 49

usp\_SelectItemTypeByNameAndFileGroup 49

usp\_SelectSubmissionItemByFileTypeIdAndCategoryID 49

usp\_SelectSubmissionItemTypesByIDs 49

usp\_SelectSubmissionItemTypeByID 49

usp\_SelectItemTypeByName 49

usp\_SelectFirstSubmissionItemTypeForFileGroup 49

usp\_SelectProductionTaskLimitedItemTypes 49

usp\_SelectFirstSubmissionItemTypesForFileGroups 49

Code 50

Aries.EditorialManager.Framework.Journal.ArticleTypeObjects.SubmissionItemType 50

Entity.cs 50

Aries.EditorialManager.Framework.Journal.ArticleTypeObjects.SubmissionItemType. Database 50

Mapper.cs 50

Pages 50

fileType2.asp 50

Trigger Points 51

Pages 51

Web/common/pdfBuildSetup.asp 51

Web/common/sendReviewerLetters.asp 51

Web/common/notifyAuthor2.asp 51

Aries.EditorialManager.Web.Command.Upload 51

UploadCompanionFile.cs 51

Issues and Concerns: 52

Spec Changes 53

# Contributors:

|  |  |
| --- | --- |
| Primary Engineer | Jean Gelinas |
| Senior Engineer | Andrew Pokrovski |
| Coding Engineer(s) | Jean Gelinas  Andrew Pokrovski |

# Document Revision History:

|  |  |  |
| --- | --- | --- |
| 07/16/2015 | Jean Gelinas | Start of Tech Spec – Journal Master Config |
| [Date] | [Engineer name] | Approved Technical Specification |
| [Date] | [Engineer name] | Revision note |

# Summary:

Integrates checkCIF functionality from the International Union of Crystallography to check .cif files for proper formatting and allows download and saving of check result pdf files.

# References:

# Design:

## Spec Support

Architectural elements that will be used in multiple aspects of application.

### Data

JOURNAL

+ CIF\_RESULT\_FILES (table)

This table contains the result files received from the CIF Checker web interface. An entry is made in this table when the CheckCIF web interface returns a results file.

SOURCE\_FILE\_CATALOGUE\_ID (BIGINT, NOT NULL)

* + - this is the ID of the file catalogue entry of the file that was submitted to CheckCIF.
    - Foreign key to FILE\_CATALOGUE. Cascade delete.

RESULT\_FILE\_CATALOGUE\_ID (BIGINT, NOT NULL)

* this is the ID of the file catalogue entry of the file that was returned by CheckCIF.
* Foreign key to FILE\_CATALOGUE. Cascade delete.

+ TRG\_CIF\_RESULT\_FILES\_maintainFileCatalogueReferenceCount(trigger on CIF\_RESULT\_FILES)

* modeled after trig\_AMENDED\_FILES\_maintainFileCatalogueReferenceCount for the AMENDED\_FILES table.

+ CIF\_CHECK\_STATUS (table)

This table will contain the current status of all pending and completed CIF Checker requests. Once a cif file is uploaded to the submission, a new record will be added to this table.

SOURCE\_FILE\_CATALOGUE\_ID (BIGINT, NOT NULL)

* the file catalogue ID of the file being checked.
* Foreign key to FILE\_CATALOGUE, Cascade delete.

STATUS (SMALLINT, NOT NULL)

* the current status of the request
  + 0 – Not sent out yet (most likely due to not having hit the trigger point yet)
  + 1 – Sent out; no results
  + 2 – Error
  + 3 – Completed

STATUS\_DETAILS (nvarchar(max), NULL)

* the text of any mechanical problems encountered during the CIF Check job.

DATE\_TESTED (DATETIME, NOT NULL)

* set when the CIF request was initiated

DOCUMENTID (INT, NULL)

* foreign key to DOCUMENT. Cascade delete.
* The document ID (if one exists) associated with the file being CIF-checked.

SCHEDULE\_GROUP\_ID (INT, NULL)

* foreign key to SCHEDULE\_GROUPS. Cascade delete.
* The ID of the schedule group for which the CIF-checked file was uploaded.

usp\_GetCIFResultFiles

* Union query with SUBMISSION\_FILES and COMPANION\_FILES

@ASSIGNMENT\_TYPE INT (1-SUBMISSION, 2-COMPANION)

@CIF\_FILE\_IDS VARCHAR (comma-delimited list of CIF file\_catalogue ids)

FILE\_CATALOGUE

GUID

FILENAME

LAST\_MODIFIED

FILE\_TYPE\_ID

STATUS

EXTERNAL\_RESOURCE\_ID

ASSIGNMENT\_FILES

ASSIGNMENT\_FILE\_ID

DATE\_LAST\_DOWNLOADED

CIF\_RESULT\_FILES

RESULT\_FILE\_CATALOGUE\_ID

CIF\_CHECK\_STATUS

SOURCE\_FILE\_CATALOGUE\_ID

STATUS (3 – completed, 2 – error, 1 – sent out, no results) *CIF\_CHECK\_STATUS*

STATUS\_DETAILS *CIF\_CHECK\_STATUS\_DETAILS*

SUBMISSION\_FILES

FILETYPE

FILEDESC *DESCRIPTION*

OFFLINE

Row\_LastModified\_TimeStamp

COMPANION\_FILES

FILETYPE

DESCRIPTION

COMPANION\_FILES\_COLOR\_CODE\_ID

NO\_COLOR\_CODE\_CHOSEN

Row\_LastModified\_TimeStamp

DFILETYPE

NAME *FILETYPENAME*

### Code

##### + cifCheckFunctions.js

* createCIFResultsLink
  + - * create download link for CIF results file
* getCIFResults (array of cif file catalogue ids)
  + - * use *usp\_GetCIFResultFiles*
      * return array of *docFilesUtil.cifResultsFile*

##### docFilesUtil.js

* cifResultsFile (struct from usp\_GetCIFResultFiles)

#### + Aries.EditorialManager.Framework.Journal.FileObjects.CIFResultFile

* We will need to create standard supporting framework objects for the CIF\_RESULT\_FILES table. The entity will inherit from [FWJ].FileObjects.FileEntity.
* The SourceFileCatalogueID will need to be an IdentityReference to a FileCatalogue object.
* The CatalogueEntry property of the base class will map to the RESULT\_FILE\_CATALOGUE\_ID field in the database.
* We will need to create a FindBySourceFileCatalogueIDs method for the mapper, which will take a list of file catalogue IDs and return CIFResultFileEntities with SourceFileCatalogueIDs that are in that list.
* As standard, we will need to add the new mapper to the DatabaseMapperManager

#### + Aries.EditorialManager.Framework.Journal.CIFObjects.CIFCheckStatus

* We will need to create standard supporting framework objects for the CIF\_REQUEST\_STATUS table.
* We will need to create a FindByFileCatalogueIDs method and supporting stored proc, which will take a list of file catalogue IDs and return CIFCheckStatus records with SourceFileCatalogueIDs that are in that list.
* We will need to create a FindByDocumentID method and supporting stored proc, which will take a document ID (int) and return CIFCheckStatus records associated with that document.
* We will need to create a FindByScheduleGRoupID method and supporting stored proc, which will take a schedule group ID (int) and return CIFCheckStatus records associated with that schedule group.
* Add a function and supporting stored proc that determines whether any documents in the list of passed-in document IDs have CIF-Checker result files.
* Name: DocumentIDsWithCIFCheckerResults
* Input: List of document IDs. If empty, consider set of all document IDs.
* Output: Dictionary mapping documentID to count of CIF Checker result files. If a document is not contained in the dictionary, then it does not have any result files associated with it. (CIF\_CHECK\_STATUS where DOCUMENTID in list and STATUS = Completed)
* Add a function and supporting stored proc that determines whether any schedule groups in the list of passed-in schedule group IDs have CIF-Checker result files.
* Name: ScheduleGroupIDsWithCIFCheckerResults
* Input: List of schedule group IDs. If empty, consider set of all schedule group IDs.
* Output: List of schedule group IDs with CIF Checker Result Files (CIF\_CHECK\_STATUS where SCHEDULE\_GROUP\_ID in list and STATUS = Completed)
* Add the new mapper to the DatabaseMapperManager
* Add a CIFCheckState enum to this namespace to enumerate the possible values of STATUS.

#### Aries.EditorialManager.Framework.Journal.FileObjects.CompanionFile

* Mapper methods to fetch a list of all companion files that have CIF Checking turned on.
* FindCIFEnabledFiles(int documentID, int revision).
* Map new FILE\_IS\_CIF column.

#### Aries.EditorialManager.Framework.Journal.FileObjects.SubmissionFile

* Mapper methods to fetch a list of all companion files that have CIF Checking turned on.
* FindCIFEnabledFiles(int documentID, int revision).
* Map new FILE\_IS\_CIF column.

## Action Links (2.13.1)

Note: Showing the results link in the search proposals results (SearchProposalResults.aspx) is kind of a bear. The determination if an action link will be shown or not is determined in the data populating stored procedure (usp\_SelectSearchProposalResults.sql). Going to be a bit difficult to determine if link should be shown through query, although possible. Revisit after everything else is done.

### Data

TRANSLATION

+ Common.ActionLinks.CIFCheckResults

“checkCIF Results”

JOURNAL

#### usp\_SelectSearchProposalResults (used only in results from proposal search)

* Add parameter to indicate role family
* add viewCIFResults (BIT DEFAULT 0) to #searchProposalResultsPreSort table, after viewHistory.
* Create an update of viewCIFResults after table population based on passed in role family.

#### ACTION\_LINKS

Add new record ID = 25

#### ACTION\_LINK\_PAGE

PAGE\_ID = 234,

ACTION\_LINK\_ID = 25,

RANK = 2

### Code

#### Aries.EditorialManager.Framework.Journal.ActionLinkPage

##### ActionLinkEnum.cs

+ CIFCheckResults = 25

##### AjaxActionLink.ashx

+ Add case to get cif check results link

#### Aries.EditorialManager.ActionLinkManager

##### ActionLinkManager.cs

* Add a lazy-loaded CIFManager instance.
* Reset the CIFManager instance in the DocumentIDsList property’s setter.
* Add a GetCIFCheckResultsLink(documentID) method.
  + Call to CIFManager.DisplayCIFResultsLink(documentID) to determine if links can be shown.
  + If link cannot be shown, return empty string.
  + If there is just one related cif file and there is a cif results pdf file, return a download link to the pdf file.
  + If there are more than one related cif files, return a link to the cif check results page.
  + Use Common.ActionLinks.CIFCheckResults for text of link.
* If CIFManager.GetResultFileCount(documentID) == 1, then we display a link directly to the one CIF result file (ID obtained via CIFManager.GetOnlyResultFile(documentID).

#### Aries.EditorialManager.Facade.Utility

##### ActionLinkManager.cs

* string GetCIFCheckResultsLink(int documentID, bool collapsedActionLink) to interface.
* string GetCIFCheckResultsLink (int documentID, bool collapsedActionLink) to class.
  + Call ActionLinkManager.GetCIFCheckResultsLink

#### Aries.EditorialManager.Framework.Journal.LayoutItems

##### SectionItemDetails.cs

Add *CIFResults* to enumeration.

#### Folders.SearchSubmissions(2.8)

##### Web\common\App\_Code\Folders\SearchResults\Controller.cs

* Create var to hold link from call to ActionLinkManager.GetCIFCheckResultsLink
* Check if returned link is not an empty string
* If link is not empty, write to the actionLinkWriter after cross check results link is written (line 1065).

#### Folders.AllAssignedSubmissions (2.13.1)

##### Web\common\App\_Code\Folders\AllAssignedSubmissions\Controller.cs

* Create var to hold link from call to ActionLinkManager.GetCIFCheckResultsLink
* Check if returned link is not an empty string
* If link is not empty, write to the actionLinkWriter after cross check results link is written (line 1065).

#### Folders.NewSubmissions (2.13.1)

##### Web\common\App\_Code\Folders\NewSubmissions\Controller.cs

* Create var to hold link from call to ActionLinkManager.GetCIFCheckResultsLink
* Check if returned link is not an empty string
* If link is not empty, write to the actionLinkWriter after cross check results link is written (line 1065).

#### Folders.RevisedSubmissions (2.13.1)

##### Web\common\App\_Code\Folders\RevisedSubmissions\Controller.cs

* Create var to hold link from call to ActionLinkManager.GetCIFCheckResultsLink
* Check if returned link is not an empty string
* If link is not empty, write to the actionLinkWriter after cross check results link is written (line 1065).

#### Folders.RevisedSubmissionsReqAssign (2.13.1)

##### Web\common\App\_Code\Folders\RevisedSubmissionsReqAssign\Controller.cs

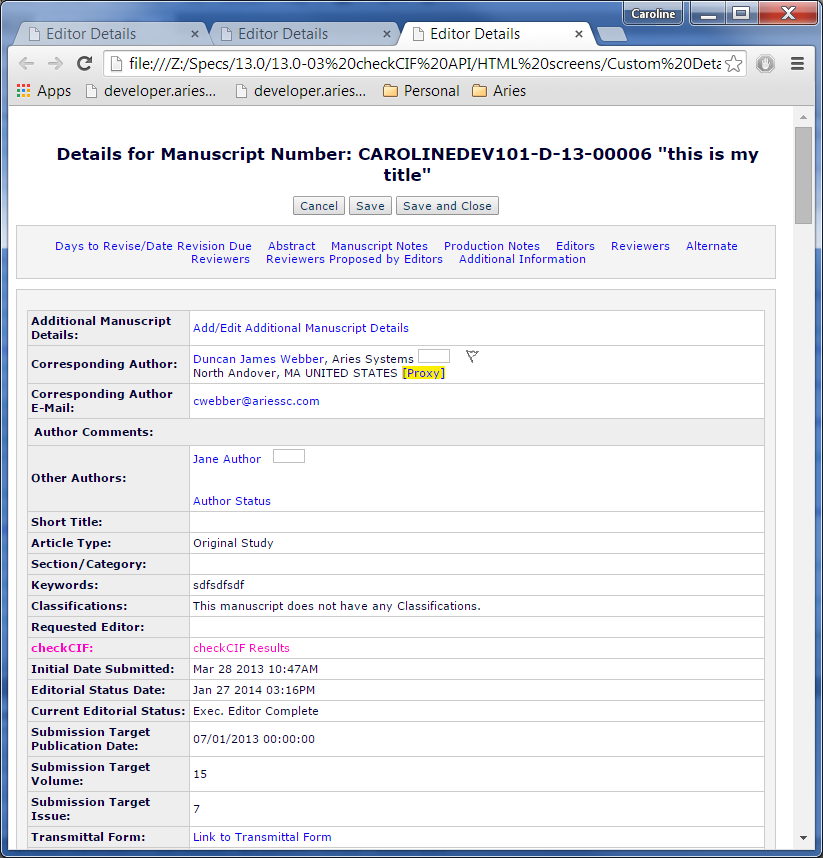
* Create var to hold link from call to ActionLinkManager.GetCIFCheckResultsLink
* Check if returned link is not an empty string
* If link is not empty, write to the actionLinkWriter after cross check results link is written (line 1065).

### Pages

#### General

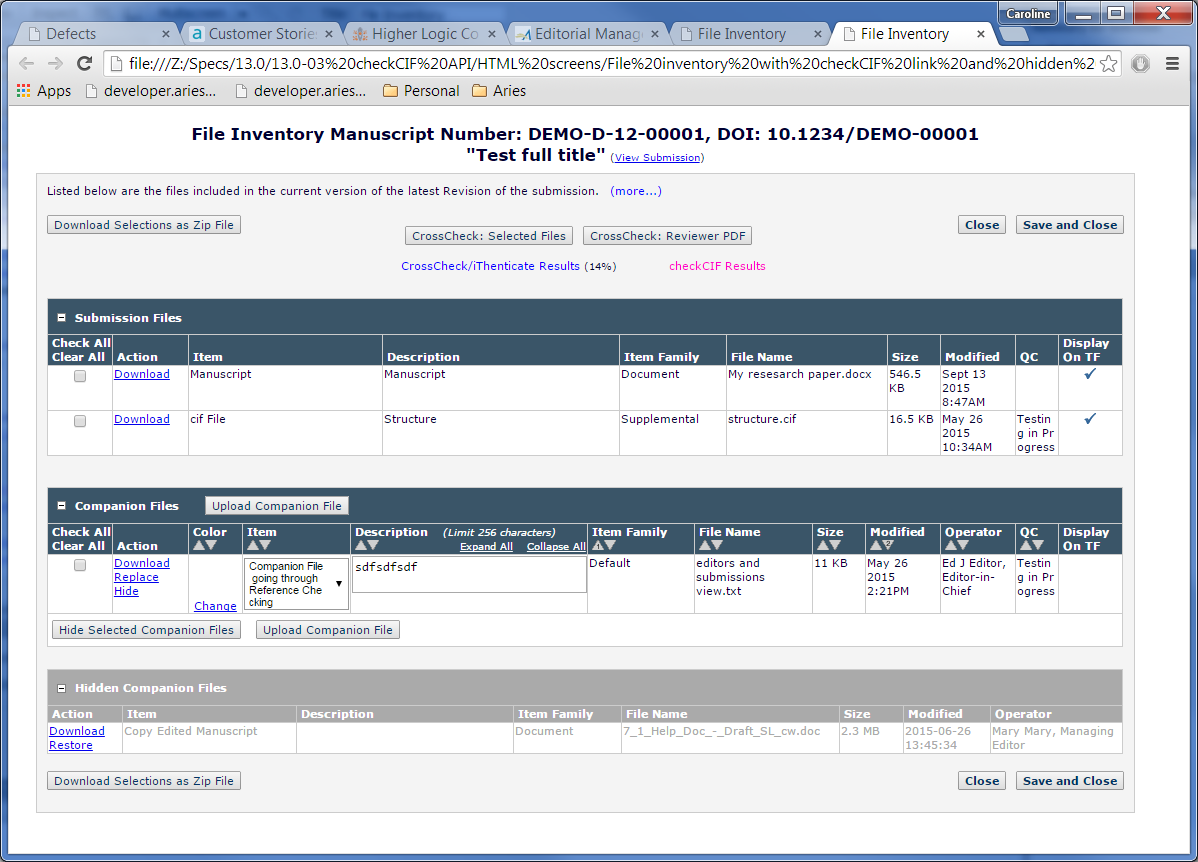
##### EMDetails.aspx/cs (2.5)

* New function, CreateCIFResultsItem, that will create table row for cif results action link. Get link html from ActionLinkManager.GetCIFCheckResultsLink.
* Add CreatCIFResultsLink function pointer in FillDictionary function, using new SectionItemDetails enum (CIFResult).



##### downloadSubFilesList.asp (2.7.2)

* Create action link manager through HTMLUtility.InitializeActionLinkManager
* After code for creation of cross check results link, write to response stream the results of *ActionLinkManager.GetCIFCheckResultsLink*.



#### Author Facing

##### authorSubmissionsSentBack.asp

##### auth\_ndauthApprove.asp

##### auth\_pendSubmissions.asp

##### authorRevisionsSentBack.asp

##### auth\_RevisionsNeedingApproval.asp

##### auth\_RevisionsBeingProcessed.asp

##### process\_submission\_decision.asp

#### Reviewer Facing

##### reviewer\_current.asp

##### reviewer\_complete.asp

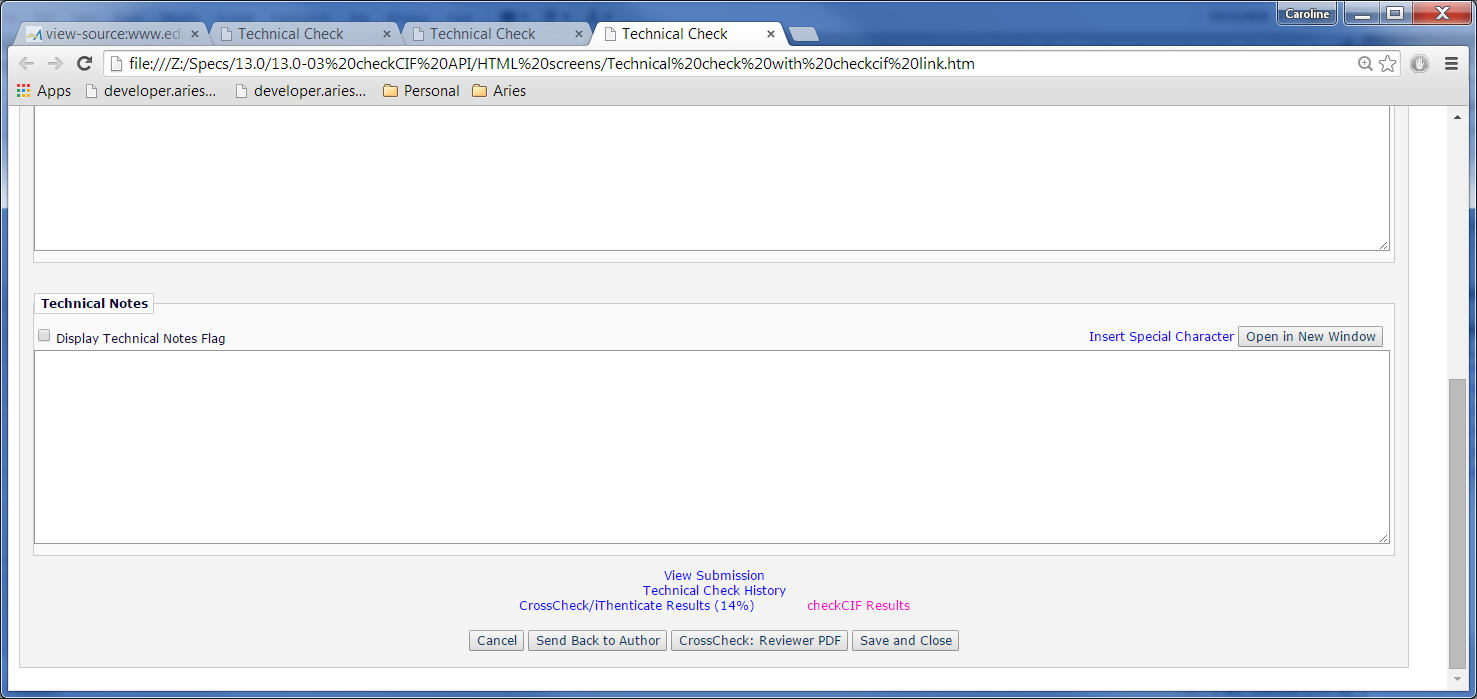
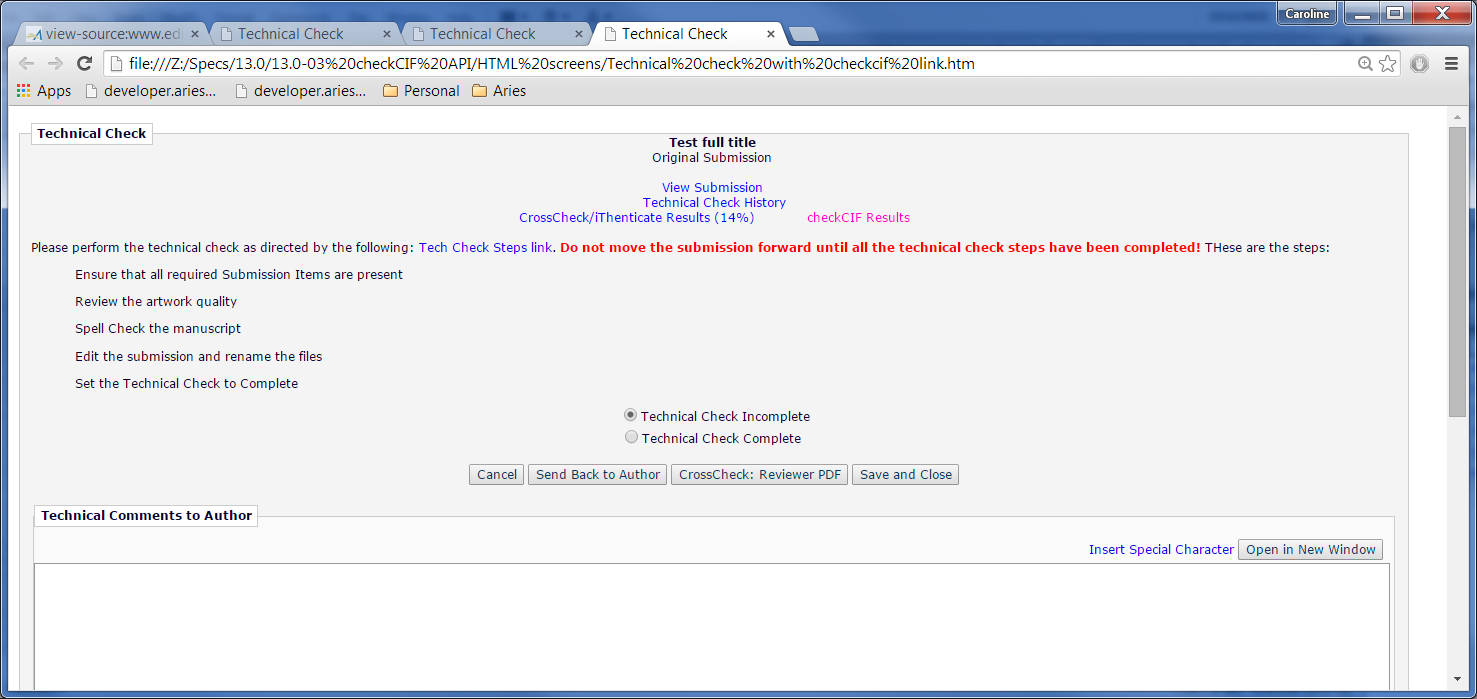
#### Editor Facing

##### EditorNewRequiresAssignment.aspx.cs

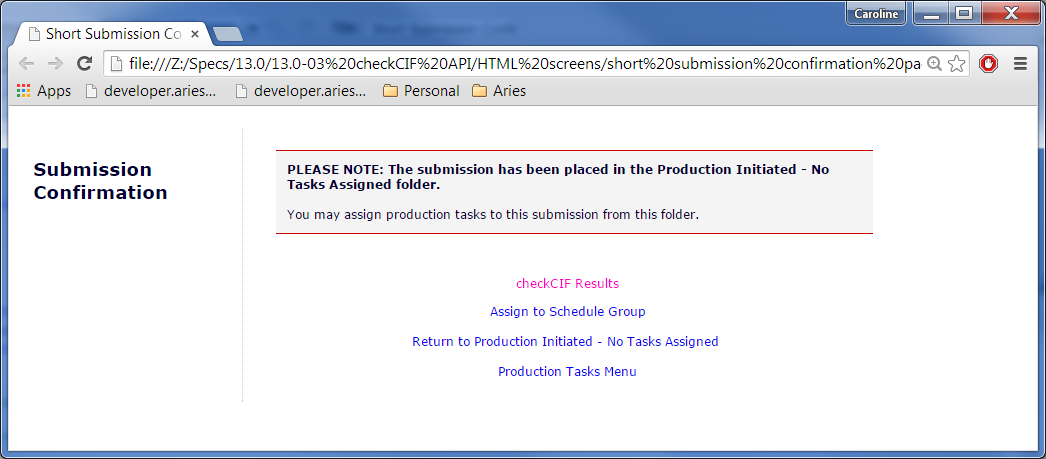
Add link in BuildActionLinkMenu function.

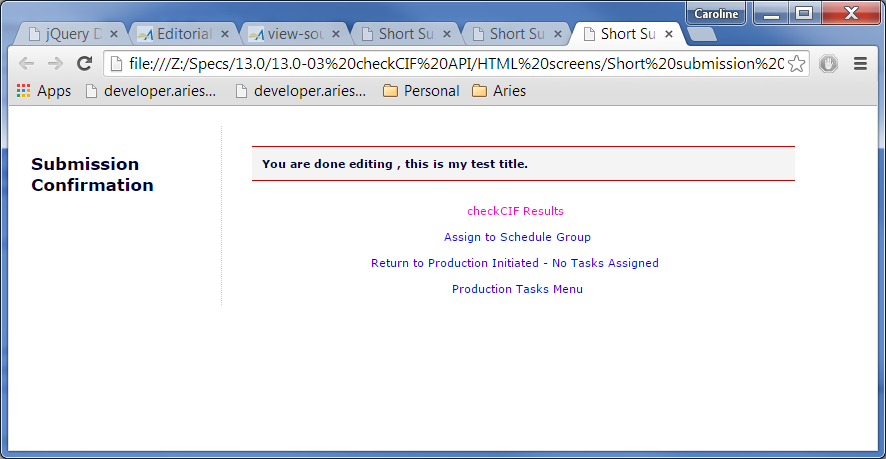
##### TechnicalCheck.aspx/cs (2.6)

* New function *ViewCIFCheckResultsURL*, which will call new *ActionLinkManager.GetCIFCheckResultsLink*.
* New TranslationLiteral to hold cifCheck results link and using *ViewCIFCheckResultsURL* as text attribute.

ShortSubmissionConfirmation.aspx/cs (2.7.1)

* New Literal control, id=litCIFCheckResults.
* Add logic in function BuildNavLinks to call A*ctionLinksManager*.*GetCIFCheckResultsLink* and place resulting text into *litCIFCheckResults*.





##### editor\_invitation.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editor\_new.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorSubByAssign.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorSubByCurEd.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorSubByStatus.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorOutForRevision.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editor\_assign\_with\_fd.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editor\_docs\_with\_fd.asp (2.13.1)

after call to create CrossCheck Results Link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorRevisedSubmissionsTechCheck.asp

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editorRevisedSubmissions.asp

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### editor\_assigned\_docs.asp

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

#### Publisher Facing

##### publisherAcceptedManuscripts.asp (2.13.1)

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### viewAllSubmissionsInProduction.asp (2.13.1)

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

##### productionInitiatedNoTasksAssigned.asp (2.13.1)

within function *outputTableBody()*, after history link

* Create var to hold result of call to HTMLObject.GetCIFCheckResultsLink.
* If var has value, write value to response stream.

## Archiving

### Data

Implement usp\_SelectAllCIFResultFilesForArchiving, which should be a union of two joins: FILEVIEW\_CIF\_RESULT\_FILES with SUBMISSION and COMPANION\_FILES, making sure to capture DOCUMENTID.

### Code

Add an invocation of usp\_SelectAllCIFResultFilesForArchiving to the FileArchivingEligibilityWeb constructor, passing the results to LoadFileList.

## Asynch

### Data

JOURNAL

CIF\_FILE\_TRAFFIC\_LOG (new)

TRAFFIC\_LOG\_ID (PK)

FILE\_CATALOGUE\_ID (FK to FILE\_CATALOGUE)

START\_TIME (datetime) – time asynch server sent cifCheck request

END\_TIME (datetime) – time asynch server received cifCheck result

SENT\_BYTES (float)

RECEIVE\_BYTES (float)

usp\_ProcessCIFFileResultRequest (new)

Updates/Inserts to CIF\_FILE\_TRAFFIC\_LOG

* @trafficLogID (BIGINT)
* @fileCatalogueID (BIGINT)
* @startTime (DATETIME)
* @endTime (DATETIME)
* @sentBytes (FLOAT)
* @receiveBytes (FLOAT)

PDFBUILDER

REQUEST\_TYPES (new record)

Description:

Request type for CIF checks.

Data:

NAME: CheckCIF Task

ACTION\_NAME: Validating CIF File

MAX\_ATTEMPTS: 3

WAIT\_INTERVAL\_BASE: 0

DESTABILIZATION\_SCORE: 0

ERROR\_INTERVAL: 120

DELAY\_TIME: 0

RELEASE\_INTERVAL: 0

FOLLOWUP\_REQUEST\_ID: 0

REQUEST\_PRIORITIES (new records)

Add one record for each record in PDF\_BUILDERS, for the request type id returned when adding the new record to REQUEST\_TYPES.

EM\_SITE\_MGNT

MASTER\_CONFIG (New Records)

**URL**

Description:

URL to call to request CIF check.

Data:

DESCRIPTION: cifCheck URL

VALUE: [http://publ.cif.co.uk/checkCIF.html](http://publcif.co.uk/checkcif.html)

**Timeout**

Description:

Seconds to wait for a response from chkCIF url.

Data:

DESCRIPTION: cifCheck Timeout

VALUE: ?

**User Name**

Description:

User name provided to log into cifCheck.

Data:

DESCRIPTION: cifCheck User Name

VALUE: ?

**Password**

Description:

Password provided to log into cifCheck.

Data:

DESCRIPTION: cifCheck Password

VALUE: ?

### Code

#### Asynch Executable

##### AsynchObject\_CIF\_Check.cpp/h,

Create a new class, modeled after AsynchObject\_Sanitize. This class will handle the submission of a single CIF file to the CIF checker by invoking .NET code via wrapper. The class will inherit from AsynchObject. It will need to implement the following functions and variables:

* + fileCatalogueID - int
  + SetFileCatalogueID() – calls GetValueFromCommandXML(“requesting-app/cifcheck/catalogueID/@value”, fileCatalogueID) to extract the file catalogue ID of the target file from the asynch request XML.
  + doParseCommand() – needs to invoke SetJournalCode(), SetJournalVersion() and SetFileCatalogueID(), loading data from asynch request XML into member variables. SetJournalCode() and SetJournalVersion() are implemented by the base class.
  + storeMetadataVariable – model after AsynchObject\_Sanitize::storeMetadataVariable, but substitute fileCatalogueID for fileGUID and smptServer. Stores current asynch request metadata within the database.
  + doExecute() – invokes ServicesHelper()->submitFileForCIFCheck(m\_sJournalCode, fileCatalogueID, m\_sTempDir, m\_iAttemptNum, m\_iMaxAttempts) within a try/catch block similar to AsynchObject\_Sanitize::doExecute(). TempDir, AttemptNum and MaxAttempts are loaded by base class functionality.

##### ServicesHelper.cpp/h –

* ServicesHelper::submitFileForCIFCheck(wchar\_t\* journal, int fileCatalogueID, wchar\_t\* directory, int attemptNum, int maxAttempts)
* method that invokes ServicesHelperFactory::Instance().submitFileForCIFCheck (same parameter list)
* ServicesHelperFactory::submitFileForCIFCheck() follows the generateSanitizedFile model to invoke the SubmitFileForCIFCheck function in the wrapper.

##### Core/Source/Wrappers/ServicesHelperWrapper.cpp

* Add a SubmitFileForCIFCheck method following existing examples (such as the ProcessExport method). The interior of the function simply needs to invoke CIFCheckTask::ProcessCIFCheck(journalCode, fileCatalogueID)

#### + Aries.EditorialManager.CIFChecker

##### + CIFCheckTask.cs

* This class will do the actual work of submitting a file to the CIFChecker, waiting for and processing a response. This also includes recording the outgoing file size and incoming file size in the CIF\_FILE\_TRAFFIC\_LOG table.

#### Aries.EditorialManager.TransferRecorders

##### + CIFTransferRecorder.cs

* public int LogCIFResultRequest(int? trafficLogID, int?

fileCatalogueId, DateTime? startTime,

DateTime? endTime,float? sentBytes,float? receiveBytes)

* private CIFTransferRecorder(string journalConnectionString)
* protected const string **\_storedProcName** = "usp\_ProcessCIFFileResultRequest";

#### + Aries.EditorialManager.AsynchServices.CIFCheck

##### + CIFCheck.cs

This class is modeled on [AEM].AsynchServices.Transfer.Transfer.cs. It contains three auto-properties: JournalCode, Version and CatalogueID (mapped to “catalogueID” so the asynch processor can extract it).

##### + CIFCheckProcessingRequest.cs

This class is modeled on [AEM].AsynchServices.Transfer.TransferProcessingRequest.cs.

##### + CIFCheckRequestingApplication.cs

This class is modeled on [AEM].AsynchServices.Transfer.TransferRequestingApplication.cs. It contains an instance of the CIFCheck class (with the XMLElementAttribute “cifcheck”) and an Identifier that is set to “transfer” during the constructor”.

#### Aries.EditorialManager.AsynchServices

##### RequestType.cs

Add CIFCheck to the enum.

##### ProcessingRequest.cs

Add CIFCheckProcessingRequest to the list of XmlIncludes above the class declaration.

##### RequestSubmitter.cs

* **SubmitCIFCheckJob**(string journal, int catalogueID).
  + “journal” being the journal containing the file to check
  + catalogueID being the ID of the file being checked (not the GUID!).
  + This method will initialize a CIFCheckProcessingRequest and invoke AsynchRequestsQueue.InsertRequest on it.

#### Aries.EditorialManager

##### SiteManagement.cs

Add item to *ConfigItemEnum* for each MASTER\_CONFIG item added previously, and correspond the table assigned id in the enum.

cifCheck URL CIF\_CHECK\_URL

cifCheckTimeout CIF\_CHECK\_TIMEOUT

cifCheck User Name CIF\_CHECK\_USER\_NAME

cifCheck Password CIF\_CHECK\_PASSWORD

NOTE: Asynch Server should be able to get the values by calling SiteManagement.GetMasterConfigValue.

## CIF Results (2.13.3)

### Data

TRANSLATION

(add) Pages.Editorial.CIFResults.Instructions

“Items associated with the current revision of the Submission, and configured to be sent to the checkCIF tool, are listed below. Click the 'Completed' link to see a PDF of the results.

Items that have not yet been sent to the tool are marked as 'Not yet tested'. Items that are currently being tested are marked as 'In Progress'. If an Item has errored out, the error message can be seen by clicking the checkCIF error hyperlink.”

(add) Pages.Editorial.CIFResults.ShortInstructions

“Items associated with the current revision of the Submission, and configured to be sent to the checkCIF tool, are listed below. Click the 'Completed' link to see a PDF of the results.”

(add) Pages.Editorial.CIFResults.CheckCIFResultsText

“checkCIF Results”

(add) Pages.Editorial.CIFResults.SendFilesButton

“checkCIF : Selected Files”

(add) Pages.Editorial.CIFResults.CrystallographicFileText

“Crystallographic File”;

(add) Pages.Editorial.CIFResults.ResultNotTestedLinkText

“Not yet tested”

(add) Pages.Editorial.CIFResults.ResultErrorLinkText

“cifCheck Error”

(add) Pages.Editorial.CIFResults.ResultInProgressLinkText

“Testing in progress”

(add) Pages.Editorial.CIFResults.ResultCompletedLinkText

“Completed”

(add) Pages.Editorial.CIFResults.ReturnToResultsPage

“Return to Results Page.

(add) Pages.Editorial.CIFResults.SendFilesButton

“checkCIF : Send Files”

(add) Pages.Editorial.CIFResults.LeaveResultsPage

“Leave Results Page”

(add) Pages.Editorial.CIFResults.CloseWarning

“One or more of the cif files was not checked or is in an error status. To send these files to the checkCIF tool, click the 'Return to Results Page' button below and then select the file(s) using the checkboxes and click the 'Selected File: checkCIF' button. To close this result page, click the 'Leave Results Page' button below.”

(add) Pages.Editorial.CIFResults.SendFilesError

“Error occurred sending files to server to be checked.\n Error:”

### **Pages**

#### Web/Common/CIFResults.aspx/cs

This page displays the results of any CIF checking operations conducted for a given document. Page layout should be drawn from the functional specification. The page elements are as follows:

* Header. Contains manuscript number and article title.
* Instructions. Have an expanded and collapsed version. Text depends on user’s role and permissions (authors and reviewers see different instructions than editors and publishers), and possibly if
* Close Button – annoys users with popup confirmation when clicked if checkCIF button is visible.
* checkCIF button – uses AsynchServices.RequestSubmitter to submit CIF check jobs for each selected item in both submission and companion file grids.
* Submission Files grid – data can be retrieved by invoking [FWJ].SubmissionFile.Database.Mapper.FindByDocIdAndRevisionQuery(documentID, revision), filtering down just the ones with a SubmissionItemType that has CIF checked off. After that, we invoke CIFCheckStatusMapper.FindByCatalogueIDs and CIFCheckResultsMapper.FindByCatalogueIDs to get the results files. Columns have a 1 to 1 mapping. Select checkbox is available if status is ‘error’ or ‘not yet tested’. CheckCIF results errors are links to the error for publishers/editors.
* Compaion Files grid – repeat procedure for Submission Files grid.

### Code

#### Aries.EditorialManager

##### SubmissionFileUtility.cs

* GetCIFFiles – return status of cif files gathered and any result files – both submission files and companion files.
* New class *CIFFile* that combines status and result file.
* SubmissionCIFFiles – collection of CIFFile.
* CompanionCIFFile – collection of CIFFile.

## Configuration

### Code

#### Aries.EditorialManager.Framework.Journal.ConfigObjects.ConfigItem

##### CIFConfig.cs (new)

Inherit from AbstractConfigSettings.

Properties:

* CIFCheckEnabled (bool, default to TRUE)
* Trigger\_OriginalPDFBuildByAuthor (bool, default to FALSE)
* Trigger\_OriginalPDFBuildByEditor (bool, default to FALSE)
* Trigger\_FirstReviewer (bool, default to FALSE)
* Trigger\_RevisedPDFBuildByAuthor (bool, default to FALSE)
* Trigger\_RevisedPDFBuildByEditor (bool, default to FALSE)
* Trigger\_AuthorNotifiedOfAccept (bool, default to FALSE)
* Trigger\_AuthorNotifiedOfRevise (bool, default to FALSE)
* Trigger\_CreateSubmission (bool, default to FALSE)

##### ConfigItemEnum.cs (additions to enum)

* CIFCheckEnabled 301
* CIFCheckTriggerOriginalPDFBuildByAuthor 302
* CIFCheckTriggerOriginalPDFBuildByEditor 303
* CIFCheckTriggerFirstReviewer 304
* CIFCheckTriggerRevisedPDFBuildByAuthor 305
* CIFCheckTriggerRevisedPDFBuildByEditor 306
* CIFCheckTriggerAuthorNotifiedOfAccept 307
* CIFCheckTriggerAuthorNotifiedOfRevise 308
* CIFCheckTriggerCreateSubmission 309

#### Aries.EditorialManager.Framework.Journal.ConfigObjects.CustomInstructions

##### CustomInstructionGroupType.cs (additions to enum)

ConfigureCheckCIFInstructions 23

##### CustomInstructionObjectType.cs (additions to enum)

ViewCheckCIFResultsInstructions 51

#### Aries.EditorialManager.Facade.Utility

##### HTMLUtilityCIFCheck.cs (new)

New HTMLUtility façade object. Should expose new CIFCheckManager and properties from new Aries.EditorialManager.Framework.Journal.ConfigObjects.ConfigItem. CheckCIFSettings

### Pages

#### ConfigureCheckCIF.aspx.cs (new) (2.3.1.1)

Properties:

* CustomInstructionsMananager (Aries.EditorialManager.CustomInstructionsManager)

### Data

JOURNAL

MASTER\_CONFIG (New Records)

**Enable CIF Check**

**Description:**

Config record indicating if cifCheck is enabled for Journal.

Data:

CONFIG\_ID: 301

DESCRIPTION: Enable CIF Check

VALUE: True/False (default to True)

CIF Check Trigger – Original PDF Building By Author

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 302

DESCRIPTION: CIF Check Trigger – Original PDF Building By Author

VALUE: True/False (default to False)

CIF Check Trigger – Original PDF Building By Editor

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 303

DESCRIPTION: CIF Check Trigger – Original PDF Building By Editor

VALUE: True/False (default to False)

CIF Check Trigger – First Reviewer Invited/Assigned

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 304

DESCRIPTION: CIF Check Trigger – First Reviewer Invited/Assigned

VALUE: True/False (default to False)

CIF Check Trigger – Revised PDF Building by Author

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 305

DESCRIPTION: CIF Check Trigger – Revised PDF Building by Author

VALUE: True/False (default to False)

CIF Check Trigger – Revised PDF Building by Editor

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 306

DESCRIPTION: CIF Check Trigger – Revised PDF Building by Editor

VALUE: True/False (default to False)

CIF Check Trigger – Author Notified of Accept

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 307

DESCRIPTION: CIF Check Trigger – Author Notified of Accept

VALUE: True/False (default to False)

CIF Check Trigger – Author Notified of Revise

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 308

DESCRIPTION: CIF Check Trigger – Author Notified of Revise

VALUE: True/False (default to False)

CIF Check Trigger – Create Submission

Description:

Config record indicating that cifCheck should be triggered if configured and if.cif file is uploaded as part of the submission at this trigger point, and a cifCheck pdf file has not yet been received/sent for, send the .cif file to cifCheck now.

Data:

CONFIG\_ID: 309

DESCRIPTION: CIF Check Trigger – Create Submission

VALUE: True/False (default to False)

ADMINACCESSFUNCTIONS (Policy Manager - New Records)

Configure checkCIF – new Policy Manager Option

Description:

Brings up link to new configuration page for checkCIF settings under Submission Policies header after Configure Similarity Check.

NOTE:

1. Because item order for this grouping is inconsistent, will have to reset the item order with script to ensure that this setting is placed appropriately in the listing.
2. Need to add new resource to Translation Database.

Data:

PAGEID: 4

SECTIONID: 3

GROUPID: 3

ITEMORDER: ?

PRODUCTID: 1

ITEMTEXT: Configure checkCIF

WEBPAGENAME: CheckCIFSettings.aspx? customInstructionsGroupType= CheckCIFInstructions& policy=SubmissionPolicies

ACTIVE: 1

ADMINRESTRICTEDACTIVE: 0

ELIGIBLEADMINACCESS: 1

POLICYDISPLAYRULES: 0

RESOURCEID: Pages.Admin.PolicyManager. LinkConfigureCIFCheck (Configure checkCIF)

CUSTOM\_INSTRUCTIONS (New Records)

View Check CIF Results Instructions

Description:

Instructions to be used in ConfigureCheckCIF.aspx and displayed ?

Data:

CUSTOM\_INSTRUCTION\_ID: 51

DEFAULT\_INSTRUCTIONS: “Configure the options below.

Submit this page, then click the 'Edit Submission Item Types' link to select at least one Submission Item to be sent to the checkCIF tool. If you need to disable checkCIF, please unselect the checkbox labelled 'Enable checkCIF'”

CUSTOM\_INSTRUCTIONS: NULL

INSTRUCTION\_LABEL\_1: blank

RANK: 1

CUSTOM\_INSTRUCTION\_TYPE\_ID: 23

HAS\_DEFAULT\_INSTRUCTIONS: 1

HAS\_EXPLANATORY\_TEXT: 0

SUPPRESS\_DISPLAY: 0

TRANSLATION

(add) Pages.Admin.PolicyManager.LinkConfigureCIFCheck

“Configure checkCIF”

(add) Common.CustomInstructions.Default.ViewCheckCIFResultInstructions

“The files shown below are eligible for analysis using the checkCIF tool from the International Union of Crystallography (IUCr). If the files have already been sent to the checkCIF tool, the analysis report can be viewed by clicking the 'Completed' link.”

(add) Common.CustomInstructions.GroupDefaultHeader.CheckCIFInstructions

“The files shown below are eligible for analysis using the checkCIF tool from the International Union of Crystallography (IUCr). If the files have already been sent to the checkCIF tool, the analysis report can be viewed by clicking the 'Completed' link.”

(add) Common.CustomInstructions.GroupDefaultHeader.CheckCIFInstructions

“Default Instructions for checkCIF Result Files.”

(add) Common.CustomInstructions.GroupTarget.CheckCIFInstructions

“Target”

(add) Common.CustomInstructions.Target.CheckCIFInstructions

“View checkCIF Results”

(add) Common.CustomInstructions.GroupDefaultGuidelines.CheckCIFInstructions

“The text shown below is the default text which appears when the user views the checkCIF results. You may revert to this text by clicking "Revert to Default Instructions" under the instruction box on the page Configure checkCIF Results.”

(add) Pages.Admin.ConfigureCheckCIF.HeaderTitle

“Configure checkCIF”

(add) Pages.Admin.ConfigureCheckCIF.Instructions

“Configure the options below. Submit this page, then click the 'Edit Submission Item Types' link to select at least one Submission Item to be sent to the checkCIF tool. If you need to disable checkCIF, please unselect the checkbox labelled 'Enable checkCIF'.”

(add) Pages.Admin.ConfigureCheckCIF.EnableCheckCIF

“Enable checkCIF”

(add) Pages.Admin.ConfigureCheckCIF.TriggerGridHeading

“Select the workflow points that will trigger checkCIF:”

(add) Pages.Admin.ConfigureCheckCIF.OriginalPDFByAuthor

“Original PDF Building By Author”

(add) Pages.Admin.ConfigureCheckCIF.OriginalPDFByEditor

“Original PDF Building By Editor”

(add) Pages.Admin.ConfigureCheckCIF.FirstReviewerInvited

“First Reviewer Invited/Assigned for Original Submission”

(add) Pages.Admin.ConfigureCheckCIF.RevisedPDFByAuthor

“Revised PDF Building By Author”

(add) Pages.Admin.ConfigureCheckCIF.RevisedPDFByEditor

“Revised PDF Building By Editor”

(add) Pages.Admin.ConfigureCheckCIF.AuthorNotifiedAccept

“Author Notified of Editor Decision in the ‘Accept’ Decision Family”

(add) Pages.Admin.ConfigureCheckCIF.AuthorNotifiedRevise

“Author Notified of Editor Decision in the ‘Revise’ Decision Family”

(add) Pages.Admin.ConfigureCheckCIF.CreateSubmission

“Create Submission”

(add) Pages.Admin.ConfigureCheckCIF.ConfirmHeaderTitle

“The checkCIF configuration has been saved.”

(add) Pages.Admin.ConfigureCheckCIF.ScriptInInstructions

“SCRIPT tags are not allowed in Customized Instructions.”

(add) Pages.Admin.ConfigureCheckCIF.MustChooseATrigger

“Please select at least one workflow point that will trigger checkCIF, or, to disable the feature, clear the ‘Enable checkCIF’ checkbox and click the Submit button again.”

(add) Pages.Admin.ConfigureCheckCIF.EmptyInstructions

“Please enter customized instructions for the ‘View checkCIF Results’ page or click the Revert to Default Instructions link to use the default instructions on that page.”

## Details Page Layouts

### Data

JOURNAL

DETAILS\_PAGE\_LAYOUT\_ITEMS (new record)

Description:

New record for showing cif results link in submission details page.

Data:

HAS\_ADDITIONAL\_PERMISSIONS: 1

IS\_PRODUCTION\_ITEM: 1

IS\_SECTION\_HEADING: 0

CATEGORY\_ID: 1

DETAILS\_LABEL : checkCIF Results

ACTIVE: 1

CONTEXT\_AVAILABILITY: 3

ITEM\_RESOURCE\_ID: Common.PageLayout.ItemName. LabelCheckCIFResults

DETAILS\_LABEL\_RESOURCE\_ID: Pages.Editorial.Details. LabelCheckCIFResults

TRANSLATION

(add) Common.PageLayout.ItemName.LabelCheckCIFResults

“checkCIF Results”

(add) Pages.Editorial.Details.LabelCheckCIFResults

“checkCIF”

## EMIA (2.17)

### Pages

#### qdStats72.asp

* Add a new checkbox as specified by the functional spec to qdStats72.asp

#### newSubs72.asp

* read the value of the checkbox off the previous page. If it is true, write a new row, do the database query bounded by time and display the per-journal data, as well as the totals. For an example of the pattern, follow the variable “displaySubsTransferredOut”.

## Manager

### Code

#### + Aries.EditorialManager.CIFChecker

##### CIFCheckManager.cs (new)

This class will handle business logic related to the CIFChecker, such as determining whether a particular document needs to have the CIF Results link displayed. It needs to be able to determine CIF link display status based on

a) user’s current permissions and b) status of files associated with the document

Functions:

* **CIFCheckManager**(string journal, AbstractRolePermissionsEntity permissions, list<int> documentIDs)
  + Journal is the current working journal
  + Permissions holds role permissions for the current user
  + documentIDs may be null. It is a list of document IDs for which we should pre-determine whether we are to display a CIF Results link or not.
  + This will need to first load the ‘enable CIF check’ MASTER\_CONFIG setting. If that is to true, then it will need to invoke [FWJ].CIFObjects.CIFCheckStatus.DocumentIDsWithCIFCheckerResults(documentIDs), and store the results of that operation in a dictionary for quick lookup.
* **CIFCheckManager**(DatabaseMapperManager mapperManager, AbstractRolePermissionsEntity permissions, list<int> documentIDs)
  + Overload of the above constructor, but we give it a mapper manager directly to avoid re-initializing it.
* bool **DisplayCIFResultsLink**(int documentID)
* First check if EnableCIFCheck is true. If so, then look the document up in the dictionary generated during the constructor. If that dictionary is not initialized, then individually invoke [FWJ].CIFObjects.CIFCheckStatus.DocumentIDsWithCIFCheckerResults(), passing in the documentID. If the returned collection contains the passed-in document ID, then return true.
* int **GetResultFileCount**(int documentID)
* This method attempts to access the documentID – result file count dictionary initialized in the constructor and returns the number of result files associated with the given document, returning 0 if the document does not exist in the dictionary.
* guid **GetOnlyResultFile**(int documentID)
* This method returns the guid of the result file associated with the given document ID. We will make it throw an exception if more than one result file is associated with the given document ID.
* void **SubmitCIFCheckerJobs**(int documentID)
* This method will invoke FindCIFEnabledFiles from both the companion and submission file mappers. Using the file catalogue IDs from that list, it will then invoke CIFCheckStatus.FindByFileCatalogueIDs. Any records returned by that with an “in-process” or “completed” status will be eliminated from the list. Finally, we invoke RequestSubmitter.SubmitCIFCheckJob for each remaining file that has not been submitted for CIF checking (“not yet tested”, “error” or no status record).
* void **SubmitCIFCheckerJobs**(list<int> fileCatalogueIDs)
* A convenience method to directly invoke RequestSubmitter.SubmitCIFCheckJob for each item in the list of file catalogue IDs.
* bool **CIFCheckEnabled**
  + Auto-property that indicates whether the CIF Check feature is enabled at all.
* bool **CIFTriggerPointEnabled**(CIFTriggerPoint)
  + - Convenience method that takes a CIF Trigger Point and checks the CIF configuration settings for whether or not the given trigger point is enabled.

Properties:

* + - * Settings (Aries.EditorialManager.Framework.Journal.

ConfigObjects.ConfigItem.CheckCIFSettings)

#### + Aries.EditorialManager.Facade.BusinessLayer

##### CIFManager.cs (new)

This façade class will expose the functionality of the above-defined CIFManager class to classic asp page.

## Merge Fields (7)

### Data

Add a new entry to DEEPLINK\_TYPE. LINK\_TEXT = ‘[Text to be determined by PM]’, everything else as null. Make sure that the ID matches the value of the element added to the enum below.

### Code

#### Aries.EditorialManager.Framework.Journal.DeepLinkObjects

##### DeepLinkType.cs enum

* + Add new CheckCIF Results PDF enum value.
  + [DeepLinkFunctionName] = “viewCIFResultsPDF”
  + [DeepLinkCategory] = DeepLinkCategoryType.RegisteredRecipientRequired, DeepLinkCategoryType.SubmissionRequired.

#### Aries.EditorialManager.LetterHandling.MailMerge

##### MergeRuleset.cs

* Add the new deep link class to ‘AddDeepLinkRules()’.

#### Aries.EditorialManager.LetterHandling.MailMerge.MergeCodes.DeepLinks.

##### + ViewCIFResultsPDFDeepLink.cs

* This new class handles the generation of the “%CHECKCIF\_RESULTS\_PDF%” link. It needs to inherit from AbstractSubmissionDeepLink, and can otherwise be modeled on the ReferenceCheckingResults class in the same namespace.

### Pages

#### Web/common/linkval.asp

Add handling for the new deep link, modeled after “referenceQcResults”. Add a case with the “viewCIFResultsPDF” command to the switch(fn) block around line 887. Add a case to the switch(fn) block around line 1890 – execute a partial login, set “isPartialLogin” = true and the URL to the CIF Results page.PDF Building

### Code

#### AsynchObject\_PDFBuild\_Process.cpp:

In the ConcatenatePDFs method, there is a section of code that invokes AddHyperLink if compInfoStruct.ReferenceResultsLinkText is a non-empty string. We will want to modify compInfoStruct to add a property “CIFResultsLinkText” and “CIFResultsURLText”, and invoke AddHyperlink using those parameters as is done for the reference results link.

In order to pass this information, we will need to add extra information to the asynch request.

#### Web/Common/docFilesUtil.js

* RebuildPdf

Modify to retrieve whether CIF Check is enabled, and whether or not the current document has any CIF check results (via CIfCheckManagerFacade.**GetResultFileCount** and CifEnabled methods)

#### Web/Common/AsynchFunctions.js

Modeled after getReferenceCheckingResultsLinkForFile, we create a getCIFResultsLinkForFile function, constructing a downloadURLNode. The only difference is the translation resource used and the URL which needs to point to the CIF check results page instead.

This function will need to be invoked ~line 2051 and 2131 in the BuildSubmissionPDFXML method, checking that it’s not an external resource ID and that CIFCheck and includeCIFResultsLinks are both true. includeCIFResultsLinks is passed in externally, CIFCheck is the value of FILE\_IS\_CIF from the invocation of prGetPDFBuildFileList.

We will also need to modify createPDFJobCommand to pass the includeCIFResultsLinks parameter down the chain to BuildSubmissionPDFXML.

### Data

JOURNAL

#### prGetPDFBuildFileList

Add FILE\_IS\_CIF from DFILETYPE to the column list.

### Pages

#### pdfBuildSetup.asp

* .RebuildPdf method:

Modify these files to retrieve whether CIF Check is enabled, and whether or not the current document has any CIF check results (via CIfCheckManagerFacade.**GetResultFileCount** and CifEnabled methods)

## Permissions

### Data

JOURNAL

AUTHORROLE

+ VIEWCIFCHECKRESULT column (bit, default 0)

EDITORROLE

+ VIEWCIFCHECKRESULT column (bit, default 0)

+ INITIATECIFCHECK column (bit, default 0)

PUBLISHERROLE

+ VIEWCIFCHECKRESULT column (bit, default 0)

+ INITIATECIFCHECK column (bit, default 0)

REVIEWERROLE

+ VIEWCIFCHECKRESULT column (bit, default 0)

usp\_InsertAuthorRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to AUTHORROLE.VIEWCIFCHECKRESULT.

usp\_UpdateAuthorRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to AUTHORROLE.VIEWCIFCHECKRESULT.

usp\_SelectAuthorRoleSettings

+ AUTHORROLE.VIEWCIFCHECKRESULT to Select list.

usp\_InsertEditorRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to EDITORROLE.VIEWCIFCHECKRESULT.

New parameter *@INITIATE\_CIF\_CHECK (bit)* goes to EDITORROLE.INITIATECIFCHECK.

usp\_UpdateEditorRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to EDITORROLE.VIEWCIFCHECKRESULT.

New parameter *@INITIATE\_CIF\_CHECK (bit)* goes to EDITORROLE.INITIATECIFCHECK.

usp\_SelectEditorRoleSettings

+ EDITORROLE.VIEWCIFCHECKRESULT to Select list.

+ EDITORROLE.INITIATECIFCHECK to Select list.

usp\_SelectAllEditorRoleSettings

+ EDITORROLE.VIEWCIFCHECKRESULT to Select list.

+ EDITORROLE.INITIATECIFCHECK to Select list.

usp\_InsertPublisherRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to PUBLISHERROLE.VIEWCIFCHECKRESULT.

New parameter *@INITIATE\_CIF\_CHECK (bit)* goes to PUBLISHERROLE.INITIATECIFCHECK.

usp\_UpdatePublisherRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to PUBLISHERROLEVIEWCIFCHECKRESULT.

New parameter *@INITIATE\_CIF\_CHECK (bit)* goes to PUBLISHER.INITIATECIFCHECK.

usp\_SelectPublisherRoleSettings

+ PUBLISHERROLE.VIEWCIFCHECKRESULT to Select list.

+ PUBLISHERROLE.INITIATECIFCHECK to Select list.

usp\_SelectAllPublisherRoleSettings

+ PUBLISHERROLE.VIEWCIFCHECKRESULT to Select list.

+ PUBLISHERROLE.INITIATECIFCHECK to Select list.

usp\_InsertReviewerRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to REVIEWERROLE.VIEWCIFCHECKRESULT.

usp\_UpdateReviewerRoleSettings

New parameter *@VIEW\_CIF\_CHECK\_RESULT (bit)* goes to REVIEWERROLE.VIEWCIFCHECKRESULT.

usp\_SelectReviewerRoleSettings

+ REVIEWERROLE.VIEWCIFCHECKRESULT to Select list.

usp\_SelectAllReviewerRoleSettings

+ REVIEWERROLE.VIEWCIFCHECKRESULT to Select list.

TRANSLATION

(add) Common.Admin.RoleManager.ViewCIFCheckResult

“View checkCIF Results”

(add) Common.Admin.RoleManager.InitiateCIFCheck

“Initiate checkCIF”

### Code

#### Aries.EditorialManager.Framework.Journal.RoleManager

##### PermissionEnum.cs

AuthorPermissions Enum:

+ ViewCIFCheckResult

PublisherPermissions Enum:

+ ViewCIFCheckResult

+ InitiateCIFCheck

EditorPermissions Enum :

+ ViewCIFCheckResult

+ InitiateCIFCheck

##### IEditorPublisherRolePermissions.cs

+ InitiateCIFCheck

##### AbstractRolePermissionsEntity.cs

+ ViewCIFCheckResult property (bool, default false)

NOTE:

* + - AuthorRole and ReviewerRole entities do not need any changes because of base class AbstractRolePermissionEntity.
    - EditorRole and PublisherRole entities get *VIEWCIFCHECKRESULT* from AbstractRolePermissionEntity.

#### Aries.EditorialManager.Framework.Journal.RoleManager.AuthorRolePermissions. Database

##### Mapper.cs

Functions:

DoLoad

Add population of new Entity property *VIEWCIFCHECKRESULT*

SetPersistenceParameters

Add send of new db parameter *@VIEW\_CIF\_CHECK\_RESULT*

#### Aries.EditorialManager.Framework.Journal.RoleManager.EditorRolePermissions

##### Entity.cs

Properties:

+ Implement IEditorPublisherRolePermissions.InitiateCIFCheck.

#### Aries.EditorialManager.Framework.Journal.RoleManager.EditorRolePermissions. Database

##### Mapper.cs

Functions:

DoLoad:

Add population of entity property *ViewCIFCheckResult*

Add population of entity property *InitiateCIFCheck*

SetPersistenceParameters:

Add send of new db parameter *@VIEW\_CIF\_CHECK\_RESULT*

Add send of new db parameter @INITIATE\_CIF\_CHECK

#### Aries.EditorialManager.Framework.Journal.RoleManager.PublisherRolePermissions

##### Entity.cs

Properties:

+ Implement IEditorPublisherRolePermissions.InitiateCIFCheck.

#### Aries.EditorialManager.Framework.Journal.RoleManager.PublisherRolePermissions. Database

##### Mapper.cs

Functions:

DoLoad:

Add population of entity property *ViewCIFCheckResult*

Add population of entity property *InitiateCIFCheck*

SetPersistenceParameters:

Add send of new db parameter *@VIEW\_CIF\_CHECK\_RESULT*

Add send of new db parameter @INITIATE\_CIF\_CHECK

#### Aries.EditorialManager.Framework.Journal.RoleManager.ReviewerRolePermissions. Database

##### Mapper.cs

Functions:

DoLoad

Add population of new Entity property *VIEWCIFCHECKRESULT*

SetPersistenceParameters

Add send of new db parameter *@VIEW\_CIF\_CHECK\_RESULT*

#### Aries.EditorialManager.Facade.Role

##### ReviewerRoleManager.cs

* Add ViewCIFCheckResult property to interface.
* Add ViewCIFCheckResult property to class.

##### AuthorRoleManager.cs

* Add ViewCIFCheckResult property to interface.
* Add ViewCIFCheckResult property to class.

##### EditorRoleManager.cs

* Add ViewCIFCheckResult property to interface.
* Add InitiateCIFCheck property to interface.
* Add ViewCIFCheckResult property to class.
* Add InitiateCIFCheck property to class.

##### PublisherRoleManager.cs

* Add ViewCIFCheckResult property to interface.
* Add InitiateCIFCheck property to interface.
* Add ViewCIFCheckResult property to class.
* Add InitiateCIFCheck property to class.

#### Web/Common

##### authorRoleManager.js

* Add population of ViewCIFCheckResult to *getAuthorRoleData*.

##### reviewerRoleManager.js

* Add population of ViewCIFCheckResult to *getReviewerRoleData*.

##### editorRoleManager.js

* Add population of ViewCIFCheckResult to *getEditorRoleData*.
* Add population of InitiateCIFCheck to *getEditorRoleData.*

##### publisherRoleManager.js

* Add population of ViewCIFCheckResult to *getPublisherRoleData*.
* Add population of InitiateCIFCheck to *getPublisherRoleData.*

#### Web/Common/App\_Code

##### roleMgrEditorUpdate.cs

* Add population of ViewCIFCheckResult and InitiateCIFCheck to *LoadEntity* function.

### Pages

#### RoleManagerAuthor.aspx.cs

* Add pass of page object to RoleData constructor.
* Add to RoleData.PermissionList – ViewCIFCheckResult – get cifcheckenabled flag from passed in page.RetrieveConfigItem<CIFConfig>.

#### roleMgrReviewer2.asp

* Create HTMLUtility object
* Capture into cifCheckEnabled var HTMLUtility.GetConfigSetting
* Place new checkbox <li> after cross check checkbox (in node\_1), named *viewCIFCheckResult*. Check with HTMLUtility object first to make sure that cif is enabled.
* See bottom where added ‘hidden’ input type for *canInitiateCrossCheck*. Don’t know if this will be necessary for cif items.

#### RoleMgrReviewer3.aspx.cs

* Add population of roleEntity.ViewCIFCheckResult from request from checkbox name *viewCIFCheckResult* in LoadEntity function.

#### roleMgrEditor2.asp

* capture cif enabled property from HTMLObject into var.
* place *viewCIFCheckResult* and *initiateCIFCheck* named checkboxes after cross check boxes in node\_5 if cif enabled property is true.
* See bottom where added ‘hidden’ input type for *canInitiateCrossCheck*. Don’t know if this will be necessary for cif items.

#### roleMgrPublisher2.asp

* capture cif enabled property from HTMLObject into var.
* place *viewCIFCheckResult* and *initiateCIFCheck* named checkboxes after cross check boxes in node\_0 if cif enabled property is true.
* See bottom where added ‘hidden’ input type for *canInitiateCrossCheck*. Don’t know if this will be necessary for cif items.

#### RoleMgrPublisher3.aspx.cs

* Add population of roleEntity.ViewCIFCheckResult from request from checkbox name *viewCIFCheckResult* in LoadEntity function.
* Add population of roleEntity.InitiateCIFCheck from request from checkbox name *initiateCIFCheck* in LoadEntity function.

## Site Cleanout(9)

We will need to add a stored proc, usp\_PeopleDocCleanDeleteCIF\_FILE\_TRAFFIC\_LOG, and add it to the GetCleanOutProcsList() methods of CleanOutDocumentData.aspx.cs and CleanOutPeopleandDocumentData.aspx.cs, both located in emia/ha/dataimportservice.

## File Statistics

### Pages

#### Web/Common/Admin/getFileStatistics.asp

Make use of the CIF\_FILE\_TRAFFIC\_LOG table as the data source, and emulate existing code to display an additional table (but only if CIF check is enabled). Emulate the bExternalTransfers block at line 855.

## Submission Item Type

### Data

JOURNAL

DFILETYPE

+ FILE\_IS\_CIF (bit, default)

usp\_InsertSubmissionItemType

New parameter *@fileIsCIF (bit)* goes to DFILETYPE.FILE\_IS\_CIF.

usp\_UpdateSubmissionItemType

New parameter *@fileIsCIF (bit)* goes to DFILETYPE.FILE\_IS\_CIF.

usp\_SelectAllSubmissionItemTypes

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectItemTypeByNameAndFileGroup

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectSubmissionItemByFileTypeIdAndCategoryID

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectSubmissionItemTypesByIDs

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectSubmissionItemTypeByID

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectItemTypeByName

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectFirstSubmissionItemTypeForFileGroup

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectProductionTaskLimitedItemTypes

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_SelectFirstSubmissionItemTypesForFileGroups

Add return of DFILETYPEID.FILE\_IS\_CIF

usp\_InsertSubmissionItemType\_aspClassic

New parameter *@fileIsCIF (bit)* goes to DFILETYPE.FILE\_IS\_CIF.

usp\_UpdateSubmissionItemType\_aspClassic

New parameter *@fileIsCIF (bit)* goes to DFILETYPE.FILE\_IS\_CIF.

TRANSLATION

(add) Pages.Admin.FileType.FileIsCIF

“Include this Item Type for checkCIF”

### Code

#### Aries.EditorialManager.Framework.Journal.ArticleTypeObjects.SubmissionItemType

##### Entity.cs

Properties:

+ FileIsCIF (boolean, default false)

#### Aries.EditorialManager.Framework.Journal.ArticleTypeObjects.SubmissionItemType. Database

##### Mapper.cs

Functions:

DoLoad

Add population of new Entity property *FileIsCIF*

SetPersistenceParameters

Add send of new db parameter *@fileIsCIF*

### Pages

#### fileType2.asp

Add checkbox for new SubmissionItemType.Entity.FileIsCIF for both editing and adding new submission file types. See section 2.3.2 of function specification.

#### fileType3.asp

Save of FILE\_IS\_CIF value.

## Trigger Points

### **Pages**

#### Web/common/pdfBuildSetup.asp

* Around line 1032 (where we submitAQCJobs), we will need to insert logic that checks to see if :
  1. CIF Checking is enabled
  2. Any of the “Build PDF” trigger points are enabled
* If 1 and 2 are both true, then we invoke CIFManager.SubmitCIFCheckerJobs for the current document ID.

#### Web/common/sendReviewerLetters.asp

* Around line 308 (end of referenceCheckingEnabled section), we will need to insert logic that checks to see if
  1. CIF Checking is enabled
  2. Any of the “Build PDF” trigger points are enabled
  3. This is the first batch of assigned/invited reviewers for the first revision of this submission.
* If 1, 2 and 3 are all true, then we invoke CIFManager.SubmitCIFCheckerJobs for the current document ID.

NOTE: This is very close to the logic used by reference checking, so we will want to split out the assignment of the reviewerAssignments, assignedReviewersFlag and invitedReviewersFlag variables so that they are done only once.

#### Web/common/notifyAuthor2.asp

* Around line 426 (end of referenceCheckingEnabled section), we will need to insert logic that checks to see if

1. CIF Checking is enabled
2. Any of the “Build PDF” trigger points are enabled
3. The decision family is accept or revise

* If 1, 2 and 3 are all true, then we invoke CIFManager.SubmitCIFCheckerJobs for the current document ID.

NOTE: Again, this logic is very close to the logic for reference checking.

#### Aries.EditorialManager.Web.Command.Upload

##### UploadCompanionFile.cs

Override the PostProcessFile method, to check the provided FileType’s CIF submission setting. If that’s turned on, then we need to use RequestSubmitter.SubmitCIFCheckJob to schedule a CIF submission.

## Schedule Group checkCIF results

### Code

#### Aries.EditorialManager.CIFManagement

##### CIFCheckManager.cs

* *scheduleGroupIds* to constructor – list of schedule group ids. Populate *\_scheduleGroupIds*.
* Enum *LinkForType* – indicates who is calling for a link (Document or ScheduleGroup).
* \_*scheduleGroupOneCIFResultFile* to hold CIF result file data for passed in schedule groups that only have one associated result file.
* \_*scheduleGroupCIFFiles* to hold CIF file data for passed in schedule groups.
* *ScheduleGroupIds,* which, on reset, clears *\_scheduleGroupOneCIFResultFile*, *\_scheduleGroupCIFFiles*.
* Constants
  + downloadString
  + openPageString
* Split *GetDownloadLinkData* into *GetDownloadLinkData\_ForDocument* and *GetDownloadLinkData\_ForScheduleGroup*.
* Split *GetCIFFiles* to *GetCIFFiles\_ForDocument* and *GetCIFFiles\_ForScheduleGroup*.
* Split *GetSingleCIFResult* into *GetSingleCIFResult\_ForDocument* and *GetSingleCIFResult\_ForScheduleGroup*
* *GetDownloadLinkData\_ForScheduleGroup* –
  + private
  + Checks *CIFCheckEnabled* and *permissions*.*ViewCIFCheckResult*
    - populate \_*scheduleGroupOneCIFResultFile* for each in \_*scheduleGroupIds* if null.
  + Return result file from \_*scheduleGroupOneCIFResultFile*  if it exists for schedule group id passed in.
* *GetCIFFiles\_ForScheduleGroup* – return collection of cif files for schedule group id.
  + Checks *CIFCheckEnabled* and *permissions.ViewCIFCheckResult*
    - Populates collection of companion cif files
  + Returns collection of companion cif files
* *GetSingleCIFResult\_ForScheduleGroup*
  + private
  + populates \_*scheduleGroupCIFFiles* for schedule group id, if not found in the dictionary.
  + returns result file data if in \_*scheduleGroupCIFFiles*.
* *GetCheckCIFResultsLink\_ForScheduleGroup*
  + Here because only will be called by *ScheduleGroupFileInventory*.*aspx*.
  + Need to have a translation resource loader passed in.
  + Generates check cif results link for schedule group id

#### Aries.EditorialManager

##### ActionManager.cs

* Use cif manager constants for download and page urls.

### Pages

#### CheckCIFResults.aspx.cs

* Page\_Load – expecting a ‘showType’ query string parameter that maps to *CIFCheckManager.LinkForType*.
* Separate preparation for both a document’s cif files and a schedule group’s cif files. Primarily sets page title correctly.
* Get data based on ‘showType’ from either *CIFCheckManager* - *GetCIFFiles\_ForDocument* or *GetCIFFiles\_ForScheduleGroup*.

#### ScheduleGroupFileInventory.aspx/cs

* Add div (float right) before companion files fieldset.
* Page\_Load
  + Create cif check manager and gather link
  + Show div if necessary.

# Issues and Concerns:

# Spec Changes

* Archive CIF Result files when associated file is archived.
* “Delete” CIF Result files when associated file is “deleted”.
* Clean out CIF Result files
* ? If instructions are customizable, then the default instructions for authors/reviewers will not match the config page
* If file has results but then CIF is un-checked for that file type, we still need to count the file towards the results grid/displaying the CIF results link

# DBContent:

CIF\_RESULT\_FILES

Short Description:

CheckCif Result Files

Long Description:

This table contains files returned by the 3rd party checkCIF utility.

CIF\_RESULT\_FILES.CIF\_RESULT\_FILE\_ID

Short Description:

CIF Result File ID

Long Description:

The unique identifier of a checked file / result file pair.

CIF\_RESULT\_FILES.SOURCE\_FILE\_CATALOGUE\_ID

Short Description:

Source File Catalogue ID

Long Description:

The unique identifier of a file that was sent off to the checkCIF utility.

CIF\_RESULT\_FILES.RESULT\_FILE\_CATALOGUE\_ID

Short Description:

Result File Catalogue ID

Long Description:

The unique identifier of a CIF Check Result file that was returned by the checkCIF utility.

CIF\_CHECK\_STATUS

Short Description:

CIF Check Status

Long Description:

Table containing the current states of pending, ongoing and completed CIF Check operations.

CIF\_CHECK\_STATUS.CIF\_CHECK\_STATUS\_ID

Short Description:

CIF Check Status Unique Identifier

Long Description:

Unique identifier of a CIF Check Status record.

CIF\_CHECK\_STATUS.SOURCE\_FILE\_CATALOGUE\_ID

Short Description:

Source File Catalogue ID

Long Description:

The unique identifier of a file that was sent off to the checkCIF utility.

CIF\_CHECK\_STATUS.STATUS

Short Description:

CIF Check Status

Long Description:

The numeric code identifying the current CIF Check Status.

Hints:

0 - Not yet tested, 1 - In progress, 2 - Error, 3 – Completed.

CIF\_CHECK\_STATUS.STATUS\_DETAILS

Short Description:

CIF Check Status Details

Long Description:

The detailed text of any error condition that may have occured for this particular CIF Check Status request.

CIF\_CHECK\_STATUS.DATE\_TESTED

Short Description:

Date Tested

Long Description:

Latest date at which the source file was submitted to the checkCIF utility.

CIF\_CHECK\_STATUS.DOCUMENTID

Alias: DOCUMENT.DOCUMENTID.

CIF\_CHECK\_STATUS.SCHEDULE\_GROUP\_ID

Alias: SCHEDULE\_GROUPS.SCHEDULE\_GROUP\_ID.

CIF\_FILE\_TRAFFIC\_LOG

Short Description:

CIF File Traffic Log

Long Description:

Table storing the daily aggregate volume of data that was sent/received for CIF Check purposes.

CIF\_FILE\_TRAFFIC\_LOG.TRAFFIC\_LOG\_ID

Short Description:

CIF File Traffic Log ID

Long Description:

Unique identifier of a daily log.

CIF\_FILE\_TRAFFIC\_LOG.START\_TIME

Short Description:

CIF File Traffic Log Start Time

Long Description:

For each record in this table, this field indicates the date at which the record's statistics began to be recorded.

CIF\_FILE\_TRAFFIC\_LOG.END\_TIME

Short Description:

CIF File Traffic Log End Time

Long Description:

For each record in this table, this field indicates the date at which the record's statistics stopped being recorded.

CIF\_FILE\_TRAFFIC\_LOG.SEND\_COUNT

Short Description:

CIF File Traffic Log Send Count

Long Description:

How many files were sent to the checkCIF utility today.

CIF\_FILE\_TRAFFIC\_LOG.RECEIVE\_COUNT

Short Description:

CIF File Traffic Log Receive Count

Long Description:

How many result files were received from the checkCIF utility today.

CIF\_FILE\_TRAFFIC\_LOG.SENT\_BYTES

Short Description:

CIF File Traffic Log Sent Bytes

Long Description:

How many bytes were sent to the checkCIF utility today.

CIF\_FILE\_TRAFFIC\_LOG.RECEIVE\_BYTES

Short Description:

CIF File Traffic Log Received Bytes

Long Description:

How many bytes were received from the checkCIF utility today.

AUTHORROLE.VIEWCIFCHECKRESULT

Short Description:

View CIF Check Result

Long Description:

Indicates whether or not this role can view CIF Check result files.

REVIEWERROLE.VIEWCIFCHECKRESULT

Short Description:

View CIF Check Result

Long Description:

Indicates whether or not this role can view CIF Check result files.

EDITORROLE.VIEWCIFCHECKRESULT

Short Description:

View CIF Check Result

Long Description:

Indicates whether or not this role can view CIF Check result files.

EDITORROLE.INITIATECIFCHECK

Short Description:

Initiate CIF Check

Long Description:

Indicates whether or not this role can manually initiate CIF Checks.

PUBLISHERROLE.VIEWCIFCHECKRESULT

Short Description:

View CIF Check Result

Long Description:

Indicates whether or not this role can view CIF Check result files.

PUBLISHERROLE.INITIATECIFCHECK

Short Description:

Initiate CIF Check

Long Description:

Indicates whether or not this role can manually initiate CIF Checks.

DFILETYPE.FILE\_IS\_CIF

Short Description:

CIF Check Enabled

Long Description:

Indicates whether or not this file type is eligible for CIF Check.