CONTRAST-out

Elsevier Documentation for Electronic Warehouse Dataset Deliveries

Describing CONTRAST Journal EW Output Version 2019.1, Book EW Output Version 2019.1, Satellite EW Output Version 2019.1, Ready Messages Version 3.0

Content and Data Architecture, Elsevier B.V.

Documentation version 1.27.0

May 28, 2019

Correspondence to:
Jos Migchielsen
Content and Data Architecture, Operations
DTD Development and Maintenance
Elsevier
Radarweg 29
1043 NX Amsterdam
Netherlands
Email: j.migchielsen@elsevier.com

This document was created by Elsevier's DTD Development & Maintenance Team, the team responsible for development, maintenance and support of the Elsevier SGML and XML DTDs and XML content transport schemas. Comments about the schemas and their documentation, as well as change requests, can be sent to the team. Change requests will be considered for implementation in a future version.

The Elsevier CONTRAST standard schemas and a fully clickable PDF file of this documentation are available via `http://www.elsevier.com/locate/xml`.

This is version 1.27.0 of the documentation of the CONTRAST content transport standard. **It should be noted that the standard itself, including the collection of W3C schemas, is complete and final. This document already authoratively describes the CONTRAST standard.** The authors welcome comments, suggestions for improvement.

This document was typeset using pdfTEX and the MiKTEX 2.9 distribution.

Contents

Chapter 1
Introduction

This is the documentation of Elsevier's CONTRAST standard (*con*tent *tra*nsport *s*tandard) for deliveries from Elsevier's Electronic Warehouse (EW).

A *dataset* is the name for any delivery of Elsevier content. Traditionally, datasets have been delivered to and from the Electronic Warehouse in a format called EFFECT. This format defined a dataset directory structure, and an accompanying file describing the dataset called the `dataset.toc`.

EFFECT (Exchange Format For Electronic Components And Texts) is a standard to enable large-scale deliveries of electronic files. This standard was initially developed by Elsevier to support comprehensive electronic journal/article distribution from production systems at the publisher to distribution servers either at local libraries or at a remote host organization. The EFFECT standard describes how large amounts of electronic files can be structured and encased in datasets, and how the "packing list" (the file dataset.toc), which comes with the dataset, is structured. The standard was developed in the course of the TULIP project, a five-year research project (1991–1995) on digital libraries by (then) Elsevier Science and nine major universities in the USA. The material provided by Elsevier was used to create local current awareness and article delivery database systems.

The EFFECT standard has served us well while the CAP and PreCAP workflows evolved into a smooth process for journals. It was even (mis)used for certain varieties of books, that were forced into the journal model. EFFECT's limits were reached.

The introduction of new intrastructure, based on XML, was the incentive to develop a new XML-based content transport standard for journal and book content. CONTRAST (*con*tent *tra*nsport *s*tandard) is the name of that standard. It consists of agreements about how datasets are organized and of W3C schemas that define a dataset description format.

Dataset creation is largely done by Elsevier's external suppliers. The suppliers validate the dataset with Elsevier's own validation tools before they deliver content to the Electronic Warehouse. For journals, the workflow is very automated and driven by XML orders from Elsevier's workflow system PTS.

The Electronic Warehouse receives deliveries from the supplier and sends them on to online content repositories and other users of Elsevier's electronic content. For this, the EW can use EFFECT as well as CONTRAST. Outbound CONTRAST, or " CONTRAST-out" is equal to inbound CONTRAST in almost all respects, and it is described in this document.

Another document, " CONTRAST-in", describes the Elsevier–supplier interaction: it specifies the dataset requirements and documents the XML orders that pass between the suppliers and Elsevier.

Originally, the plan was to develop different versions of CONTRAST-out for different purposes. The first and to date only version is called `ew-xcr`, a version that has become widespread for a variety of recipients of content from the Electronic Warehouse. Since these recipients have different requirements (some want only PDFs, others want no multimedia components, etc.) the standard has become somewhat loose, and therefore needs to be read in conjuction with the recipients *product format*, sometimes incorrectly equated with their *output profile*.

Chapter 2
Technical aspects

An important part of the CONTRAST transport standard are the W3C XML schemas that define the transport format.

This chapter contains some technical details about the CONTRAST-out schemas and the XML files that are structured according to these schemas.

2.1. CONTRAST versions

There are several versions of CONTRAST-out described in this document. The three major versions are CONTRAST-out for serial publications, CONTRAST-out for books and CONTRAST-out for ready messages. Each has a version number, and the version number is captured in the XML schema.

The correct way to determine the version number is by using the namespace of the `dataset` top element. For instance,

`http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2016.6/items`

is the namespace of version 2016.6 of the Electronic Warehouse output schema for serial publication items.

As is usual, this is not an actual file. The location of the XML schema file itself is to be based on the schema location.

```
<dataset
  xmlns="http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2016.6/items"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
   "http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2016.6/items
    http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2016.6/items.xsd"
  schema-version="2016.6">
```

is the typical opening of a dataset.xml file that conforms to the above-mentioned schema. The convention is used that the schema location is almost identical to the namespace. The schema location is not an existing file on the Elsevier website either. It is expected that applications deploy XML catalogs in order to retrieve an instance of the schema.

2.2. New CONTRAST versions

Recipients of output material from Elsevier's Electronic Warehouse should be aware of the following. Elsevier's current output delivery standard, CONTRAST-out, will be subject to

a version release schedule similar to that of Elsevier production systems such as PTS or the EW itself. Systems receiving CONTRAST output are expected to be able to seamlessly integrate updated versions of the CONTRAST-out schemas in their workflow by installing them in such a way that they can be replaced easily. In this respect CONTRAST is different from its predecessor, EFFECT, which has remained more or less unchanged during its lifetime.

The release schedule of CONTRAST-out will run in parallel with that of the EW, each new version of the CONTRAST-out schemas becoming available for testing purposes at approximately the same time the previous versions of EW and CONTRAST-out go live.

New versions of the CONTRAST-out schemas will retain backward compatibility with their predecessors.

Two situations may occur, (1) the tags newly introduced in CONTRAST-out are not required by the platform receiving the dataset, or (2) the newly introduced tags are required. In the first case the platform should be able to simply ignore the new tags, in the second case the receiving platform will be expecting their presence as part of an end-to-end implementation.

In case a resupply of a dataset takes place it is very well possible that in the period between receipt of the original dataset and the resupply a new CONTRAST-out version was introduced. All receiving platforms are expected to be able to handle this situation.

2.3. Version history

2.3.1. Journal 1.0

The Journal 1.0 schemas were released on 2 May 2005.

2.3.2. Journal 1.1

The Journal 1.1 schemas were released on 1 September 2005.

- The elements `pre-isbn`, `isbn` and `journal-item-properties` were changed to support delivery of ISBN-13s.
- Added attribute `type` to element `journal-item` with values in the new list journal-item-types-list to support "Subitems Mk III".
- Element `version-number` was changed to include versions up to 6.n.
- Added new PDF version 1.4.
- Added new stages Q200 and Q300.
- Changed elements `vol-first` and `iss-first` to nonnegative integers.

2.3.3. Journal 1.2

The Journal 1.2 schemas were released on 12 January 2005.

- Various lists of patterns and lists of values were introduced. Some elements are no longer declared in the mother schema.
- Element `embargo` was added to the content of `journal-item` in the items schema and to the content of `journal-issue` in the issues schema.
- PROJECT was added to production-processes-list.

The schemas were later patched (10 April 2005) to allow for delivery of issues with web-PDF files. In a second patch (3 May 2005) an error in element `jid` in the items schemas was corrected.

2.3.4. Journal 1.3

The Journal 1.3 schemas were released on 18 October 2006.

- Element `embargo` was changed slightly and moved to `journal-item-properties`.
- To support "sponsored access" element `sponsored-access` with subelement `type` (taking values in sponsored-access-types-list) were added to `journal-item-properties`.
- Stage S5 was added.
- Web-PDFs were made optional in the items schema.
- Web-PDF purpose MAIN-ABRIDGED was added and can be used in journal items.
- Stage F300 was added to the stages that can be used in issues.

In a patch on 22 December 2006 the support of F300 deliveries was changed.

2.3.5. Journal 1.4

The Journal 1.4 schemas were released on 6 March 2007.

- Value IMAGE-HIGH-RES was added to asset-types-list.
- JA 5.0.2 values were added to ml-versions-list.
- Three tombstone PITs were added to pit-list.
- Element `filesize` was added to every item in the dataset package file.

In a patch on 19 June 2007 the `filesize` elements were made optional.

2.3.6. Journal 1.5

The Journal 1.5 schemas were released on 2 August 2007.

- To support delivery of RadCon metadata files value `AUXILIARY` was added to `ml-purposes-list` and value `INFOPATH` was added to `ml-versions-list` and `ml-versions-list-journal-item`. These values can only occur on item level in issue deliveries.
- Element `ml` was made optional for items in issue deliveries.

In a patch on 24 September 2007 the correct version number (1.5) was used.

2.3.7. Journal 1.6

The Journal 1.6 schemas were released on 7 December 2007.

- To prepare for stage-based version numbering, the allowed version numbers were adapted.
- The `jid` element may start with `BS:` in capital letters.
- The XML weight `HEAD-ONLY` was reinstated.
- The element `parent-item`, not used, was removed.

In a patch on 19 December 2007 it was made possible to deliver items (not in an issue) with stage S300 or S350.

2.3.8. Journal 1.7

The Journal 1.7 schemas were released on 5 April 2008 and patched on 21 April 2008.

- Web PDF files conformant to the new web PDF specifications 6.0 will have a `pdf-version` of `1.4 6.0`.
- To prepare for thumbnails for MMCs, three new asset types were created: `IMAGE-MMC`, `IMAGE-MMC-DOWNSAMPLED` and `IMAGE-MMC-THUMBNAIL`.
- A new optional element `online-publication-date` will in future hold the online publication date as reported to the Electronic Warehouse.
- In the items schema (for articles in press), the `raw-text` was added. The EW will add these manifestations on demand.
- In the item schema the XML and PDF manifestations were made optional. The EW will omit these manifestations on demand.
- Item-based Q300 deliveries are made possible.

2.3.9. Journal 1.8

The Journal 1.8 schemas were released on 5 September 2008.

- Support for new journal article DTD 5.1.0 was added.
- The `filesize` element is now mandatory.
- In item deliveries more web-PDF properties were made possible.

2.3.10. Journal 1.9

The Journal 1.9 schemas were released on 11 December 2008.

- To support partial deliveries the value `PARTIAL-RELOAD` for element `dataset-action` was added.
- Further support for partial deliveries was added in the form of the new attribute `omitted` of element `journal-item`.
- The occurrence indicator of `web-pdf` on issue level was changed to “unbounded”.
- Value `VIDEO-FLASH` was added as possible value for element `asset/type`.

2.3.11. Journal 1.10

The Journal 1.10 schemas (v1.10) were released on 7 May 2009.

- The pattern for AIDs was changed to allow AIDs starting with 0.

2.3.12. Journal 1.11

The Journal 1.11 schemas (v1.11) were released on 25 August 2009.

- Stage values S250 and H200 were added to the mother schema. The patterns for version numbers were adapted as well.
- Web PDF files conformant to the new web PDF specifications 6.1 will have a `pdf-version` of `1.6 6.1` or `1.7 6.1`.

2.3.13. Journal 2010.2

The Journal 2010.2 schemas were released on 28 June 2010. With this release the naming convention is changed to match the EWII release names.

- Changed the pattern for element `jid` to remove “`bs:`” as an allowed prefix.
- Added new asset type `IMAGE-MMC-HIGH-RES` to the list of possible value for element `type` to support deliveries of non-downsampled images from EW.

- Removed `1.6 6.1` from the list of possible value for element `pdf-version`.
- Removed pattern restriction on element `aid` for items.
- Added new empty elements `journal-item-properties/delayed-sponsored-article` and `delayed-restricted-article`, both with attribute `release-date` and new element `funding-body-id`

2.3.14. Journal 2010.3

The Journal 2010.3 schemas were released on 26 November 2010.

- Value `1.7 6.2` was added to the list of possible values for element `pdf-version`.

2.3.15. Journal 2011.2

The Journal 2011.2 schemas were released on 18 May 2011. There were no changes with respect to the previous version.

In a patch on 2 August 2011 the value `print 1.1` for element `schema-version` was added.

2.3.16. Journal 2012.1

The Journal 2012.1 schemas were released on 10 May 2012.

- Added `print 1.2` to the list of allowed values for element `journal-issue/files-info/ml/schema-version`.

2.3.17. Journal 2012.2

The Journal 2012.2 schemas were released on 11 May 2012.

- Value `CRP` was added to the list of possible values for `pit`.
- Support for JA DTD 5.2.0 was added.
- Support for SI DTD 5.2.0 was added.
- Value `1.7 6.3` was added to the list of possible values for elements `pdf-version`.
- A new attribute named `cross-mark` was added to element `journal-item`.

2.3.18. Journal 2013.1

The Journal 2013.1 schemas were released on 18 October 2012.

- Value `1.7 6.4` was added to the list of possible values for elements `pdf-version`.

2.3.19. Journal 2013.3

The Journal 2013.3 schemas were released on 17 October 2013.

- Value was added to the list of possible values for elements `ml-versions-list` and `ml-versions-list-journal-item`.
- Value was added to the list of possible values of element `ml-purposes-list`.

2.3.20. Journal 2014.5

The Journal 2014.5 schema was released on 04 July 2014. No changes made to this version.

2.3.21. Journal 2014.6

The Journal 2014.6 schemas were released on 7 July 2014.

- Introduced a new element `collection-title` and modified the `journal-item-properties` and `journal-issue-properties` content model sequence to support `collection-title`.
- Introduced a new element `ucs-locator` and modified the `asset`, `ml`, `web-pdf`, `epub` and `raw-text` content model sequence to support `ucs-locator`.
- The type attribute of element `online-publication-date` is changed to `date-timeUTC` to capture zulu format.

The schemas were later patched (2 October 2014) to enable delivery of items with a DOI with a longer prefix as usual.

2.3.22. Journal 2015.1

The Journal 2015.1 schemas were released on 19 January 2015.

- Updated elements `ml-versions-list`, `ml-versions-list-journal-item` to support new DTD 5.4.0.
- Updated `journal-item-pits-list` to support new PITs.
- Update elements `stages-list` and `version-number-patterns` to support S280.
- Removed older version number pattern from `version-number-patterns`.

2.3.23. Journal 2015.3

The Journal 2015.3 schemas were released on 9 March 2015.

- Updated element `asset-types-list` with new values IMAGE-STRIPIN and IMAGE-PREVIEW.
- Introduced two new child elements, `height` and `width` to the element `asset`.
- The max occurrence of `ucs-locator` is set to 2.
- Introduced a new child element `pdf-pages-web` of type xs:positiveInteger to the elements `web-pdf` and `print-pdf`.
- Introduced a child element `asset` within `web-pdf`.
- Introduced support to capture file name.

2.3.24. Journal 2015.4

The Journal 2015.4 schemas were released on 1 July 2015.

- Updated element `web-pdf-versions-list` with a new value .
- Updated element `schema-version` with a new value .

2.3.25. Journal 2016.2

The Journal 2016.2 schemas were released on 11 March 2016.

- Updated pattern of elements `suppl-pattern` and `suppl-pattern-with-spinoff` to handle lengthy digits.

2.3.26. Journal 2016.6

The Journal 2016.6 schemas were released on 13 January 2017.

- Updated elements `ml-versions-list`, `ml-versions-list-journal-item` and element `dtd-version` to support new DTD 5.5.0.
- Updated `journal-item-pits-list` to support new PITs.

2.3.27. Journal 2018.1

The Journal 2018.1 schemas were released on 23 January 2018.

- Updated element `schema-version` with a new value `print 1.4`.

2.3.28. Journal 2018.4

The Journal 2018.4 schemas were released on 9 May 2018.

- Introduced two new (cover) asset types, `IMAGE-COVER-H400` and `IMAGE-COVER-H768`.

The schemas were later patched (22 August 2018) with changes to the input schemas.

2.3.29. Journal 2018.6

The Journal 2018.6 schemas were released on 26 September 2018.

- Updated elements `ml-versions-list`, `ml-versions-list-journal-item` and element `dtd-version` to support the new DTD 5.6.0.
- Updated `journal-item-pits-list` with new PITs introduced in DTD 5.6.0.
- Added optional element `jid-aid/article-number`.

The schemas were later patched (14 January 2019) with changes to the input schemas.

2.3.30. Journal 2019.1

The Journal 2019.1 schemas were released on 10 May 2019. With this release the version names no longer match the EWII release names.

- Value `1.7 7.0` was added to the list of possible values for element `web-pdf/pdf-version`.
- A list of possible values for element `print-pdf/pdf-version` was added.
- Element `print-pdf/pdf-version` was added in the issues schema.

2.3.31. Book 0.2

The Book 0.2 schemas were released on 2 May 2005.

2.3.32. Book 1.0

The Book 1.0 schemas were released on 19 October 2005.

- Element `pathname` was changed to enforce 8+3 filenames.
- To allow delivery of book projects with web-PDF files on book project level, element `web-pdf` was added to `files-info` on book project level.
- Mandatory element `raw-text` was added to `files-info` on book item level.

2.3.33. Book 1.1

The Book 1.1 schemas were released on 12 January 2006.

- Value PROJECT was added to production-processes-list.
- The pattern in element `pathname` was removed.
- A basic pattern was added to element `doi`.

2.3.34. Book 1.2

The Book 1.2 schemas were released on 6 March 2007.

- Value IMAGE-HIGH-RES was added to asset-types-list.
- Element `filesize` was added to every item in the dataset package file.
- BOOK-METADATA 5.0.0 BOOK-METADATA and six MRW 5.0.0 values added to ml-versions-list.
- Value DED was added to pit-list.
- Value O300 was added to stages-list.
- Element `book-item` was made optional.
- Support for Book DTD 5.2.1 was added.

In a patch on 5 June 2007 the element `pit` was made optional to allow for delivery of O300 datasets. In a second patch on 19 June 2007 the `filesize` elements were made optional.

2.3.35. Book 1.3

The Book 1.3 schemas were released on 2 August 2007.

- Support for deliveries with pagebreak files (on issue and item level) was added: `PAGEBREAK` was added to `ml-purposes-list`, `PAGEBREAK 5.0.0` was added to `ml-versions-list-item`.
- Made element `ml` on item level optional.

In a patch on 24 September 2007 the correct version number (1.3) was used.

2.3.36. Book 1.4

The Book 1.4 schemas were released on 7 December 2007.

- To prepare for stage-based version numbering, the allowed version numbers were adapted.

2.3.37. Book 1.5

The Book 1.5 schemas were released on 5 April 2008.

- Web PDF files conformant to the new web PDF specifications 6.0 will have a `pdf-version` of `1.4 6.0`.
- To prepare for thumbnails for MMCs, three new asset types were created: `IMAGE-MMC`, `IMAGE-MMC-DOWNSAMPLED` and `IMAGE-MMC-THUMBNAIL`.
- A new optional element `online-publication-date` will in future hold the online publication date as reported to the Electronic Warehouse.

2.3.38. Book 1.6

The Book 1.6 schemas were released on 5 September 2008.

- Support for new Book-Metadata 5.0.1 DTD was added.
- Added support for delivery of multiple ISBNs.
- The `filesize` element is now mandatory.

2.3.39. Book 1.7

The Book 1.7 schemas were released on 11 December 2008.

- To support partial deliveries the value `PARTIAL-RELOAD` for element `dataset-action` was added.

- Further support for partial deliveries was added in the form of the new attribute `omitted` of element `book-item`.
- The `MRW 5.0.0 *` values were removed as possible DTD versions.
- Ten `BOOK 5.3.0 *` values were added as possible DTD versions.
- The new PIT `COP` (copyright) was added.
- Value `VIDEO-FLASH` was added as possible value for element `asset/type`.
- `web-pdf`'s subelements `purpose` and `property` were made optional.

2.3.40. Book 1.8

The Book 1.8 schemas (v1.8) were released on 7 May 2009.

- The PIT `DCT` (dictionary) was added.
- Subelement `isbn` of `book-project-unique-ids` was re-introduced.
- A list of ISBN purposes (simple type isbn-purposes-list) was added.

2.3.41. Book 1.9

The Book 1.9 schemas were released on 25 August 2009.

- Web PDF files conformant to the new web PDF specifications 6.1 will have a `pdf-version` of `1.6 6.1` or `1.7 6.1`.

2.3.42. Book 2010.2

The Book 2010.2 schemas were released on 28 June 2010.

- Added new asset type `IMAGE-MMC-HIGH-RES` to the list of possible value for element `type` to support deliveries of non-downsampled images from EW.
- Removed `1.6 6.1` from the list of possible value for element `pdf-version`.

2.3.43. Book 2010.3

The Book 2010.3 schemas were released on 26 November 2010.

- Web PDF files conformant to the new web PDF specifications 6.2 will have a `pdf-version` of`1.7 6.2`.
- Stages F300 and U300 were added to the stages-list.
- `F300` and `U300` were added to the version number patterns.

2.3.44. Book 2011.2

The Book 2011.2 schemas were released on 18 May 2011.

- Stage E300 was added to the stages-list, while U300 was removed.
- `E300` was added to the version number patterns, while `U300` was removed.
- `BOOK-EPUB 5.0.0 BOOK-EPUB` added to the list `ml-versions-list`.
- `EPUB` added to the list `production-processes-list`.
- `EPUB` added to the list `isbn-purposes-list`.
- Added a new list `epub-purposes-list`.
- In element `book-project/files-info`, `web-pdf` was replaced by an optional choice between `web-pdf+`, `print-pdf+` and `epub+`. The latter two elements are new and have the usual subelements `pathname` and `filesize` plus (for `epub`) `purpose`.
- In element `book-item/files-info`, `web-pdf` was made optional.

In a patch on 2 August 2011 the element `schema-version` was added (with one possible value `print 1.0`).

2.3.45. Book 2012.1

The Book 2012.1 schemas were released on 10 May 2012.

- Added OLBS to the lists production-processes-list (for element production-process).
- Added COMPLETE-PF and COMPLETE-CE to the lists web-pdf-purposes-list (for element web-pdf).
- Added print 1.2 to the list of allowed values for element book-project/files-info/ml/schema-version.

2.3.46. Book 2012.2

The Book 2012.2 schemas were released on 11 May 2012.

- Value 1.7 6.3 was added to the list of possible values for elements pdf-version.

2.3.47. Book 2013.1

The Book 2013.1 schemas were released on 18 October 2012.

- Values RET and OVW were added to the list of possible values for pit.
- Added support for Book DTD 5.3.1.
- Added a new stage S280 to the list of allowed values of element stages-list, and a new schema named book-items.xsd added to support the delivery of book/module items without hub.
- Value 1.7 6.4 was added to the list of possible values for elements pdf-version.

2.3.48. Book 2013.3

The Book 2013.3 schemas were released on 17 October 2013.

- Values was added to the list of possible values of schema-version.

2.3.49. Book 2014.5

The Book 2014.5 schema was released on 04 July 2014. No changes made to this version.

2.3.50. Book 2014.6

The Book 2014.6 schemas were released on 7 July 2014.

- Introduced a new element collection-title and modified the book-item-properties and book-project-properties content model sequence to support collection-title.
- Introduced a new element ucs-locator and modified the asset, ml, web-pdf, print-pdf, epub and raw-text content model sequence to support ucs-locator.
- The type attribute of element online-publication-date is changed to date-timeUTC to capture zulu format.

The schemas were later patched (2 October 2014) to enable delivery of items with a DOI with a longer prefix as usual.

2.3.51. Book 2015.1

The Book 2015.1 schemas were released on 19 January 2015.

- Updated elements ml-versions-list and dtd-version to support new DTD 5.4.0.

- Updated `pit-list` to support new PITs.
- Removed older version number pattern from `version-number-patterns`.
- Updated elements `stages-list` and `version-number-patterns` to support S200.

2.3.52. Book 2015.3

The Book 2015.3 schemas were released on 9 March 2015.

- Updated element `asset-types-list` with new values IMAGE-STRIPIN and IMAGE-PREVIEW
- Introduced two new child elements, `height` and `width` to the element `asset`
- The max occurrence of `ucs-locator` is set to 2
- Introduced a new child element `pdf-pages-web` of type xs:positiveInteger to the elements `web-pdf` and `print-pdf`
- Introduced a child element `asset` within `web-pdf`
- Introduced support to capture file name

2.3.53. Book 2015.4

The Book 2015.4 schemas were released on 1 July 2015.

- Updated element `web-pdf-versions-list` with a new value .
- Updated element `schema-version` with a new value .

2.3.54. Book 2016.2

The Book 2016.2 schemas were released on 11 March 2016. No changes made to this version.

2.3.55. Book 2016.6

The Book 2016.6 schemas were released on 13 January 2017.

- Updated elements `ml-versions-list`, `dtd-version` to support new DTD 5.5.0.
- Updated `pit-list` to support new PITs.

2.3.56. Book 2018.1

The Book 2018.1 schemas were released on 23 January 2018.

- Updated element `schema-version` with a new value `print 1.4`.

2.3.57. Book 2018.4

The Book 2018.4 schemas were released on 9 May 2018.

- Introduced two new (cover) asset types, `IMAGE-COVER-H400` and `IMAGE-COVER-H768`.

The schemas were later patched (22 August 2018) with changes to the input schemas.

2.3.58. Book 2018.6

The Book 2018.6 schemas were released on 26 September 2018.

- Updated elements `ml-versions-list` and `dtd-version` to support the new DTD 5.5.0.
- Updated `pit-list` with new PITs introduced in DTD 5.6.0.

The schemas were later patched (14 January 2019) with changes to the input schemas.

2.3.59. Book 2019.1

The Book 2019.1 schemas were released on 10 May 2019. With this release the version names no longer match the EWII release names.

- Value `1.7 7.0` was added to the list of possible values for element `web-pdf/pdf-version`.
- A list of possible values for element `print-pdf/pdf-version` was added.
- Element `print-pdf/pdf-version` was added in the book-projects schema.

2.3.60. Satellite 2011.2

The new Satellite v2011.2 schema was released on 18 May 2011.

2.3.61. Satellite 2012.1

The new Satellite v2012.1 schema was released on 10 May 2012. There were no changes with respect to the previous version.

2.3.62. Satellite 2012.2

The new Satellite v2012.2 schema was released on 11 May 2012. There were no changes with respect to the previous version.

2.3.63. Satellite 2013.1

The new Satellite v2013.1 schema was released on 18 October 2012. No changes were made to this version.

2.3.64. Satellite 2013.3

The new Satellite v2013.3 schema was released on 17 October 2013. No changes were made to this version.

2.3.65. Satellite 2014.5

The new Satellite v2014.5 schema was released on 04 July 2014. No changes made to this version.

2.3.66. Satellite 2014.6

The Satellite 2014.6 schemas were released on 7 July 2014.

- Introduced a new element `ucs-locator` and modified the `ml` and `asset` content model sequence to support `ucs-locator`.
- Introduced a new pattern to support K-PII identifiers for the element `pii-patterns-general`

The Satellite 2014.6p1 schemas were released on 2 October 2014. There were no changes made to this version.

2.3.67. Satellite 2015.1

The Satellite 2015.1 schemas were released on 19 January 2015.

- Removed PII pattern declaration `pii-patterns-general`

2.3.68. Satellite 2015.3

The Satellite 2015.3 schemas were released on 9 March 2015.

- Introduced two new child elements, `height` and `width` to the element `asset`
- The max occurrence of `ucs-locator` is set to 2
- Introduced support to capture file name

2.3.69. Satellite 2015.4

The Satellite 2015.4 schemas were released on 1 July 2015. No changes were made to this version.

2.3.70. Satellite 2016.2

The Satellite 2016.2 schemas were released on 11 March 2016. No changes were made to this version.

2.3.71. Satellite 2016.6

The Satellite 2016.6 schemas were released on 13 January 2017. No changes were made to this version.

2.3.72. Satellite 2018.1

The Satellite 2018.1 schemas were released on 23 January 2017. No changes were made to this version.

2.3.73. Satellite 2018.4

The Satellite 2018.4 schemas were released on 9 May 2017. No changes were made to this version.

The schemas were later patched (22 August 2018) with changes to the input schemas.

2.3.74. Satellite 2018.6

The Satellite 2018.6 schemas were released on 26 September 2018. No changes were made to this version. The schemas were later patched (14 January 2019) with changes to the input schemas.

2.3.75. Satellite 2019.1

The Satellite 2019.1 schemas were released on 10 May 2019. No changes were made to this version. With this release the version names no longer match the EWII release names.

Chapter 3
Deliverables of CAP and PreCAP

This chapter deals with the *deliverables* of the CAP and PreCAP workflows.

The purpose of the CAP (Computer-Aided Production) and PreCAP workflows is to produce a number of products at various stages of the lifetime of these products. Together, these products are used to build a publication, be it in print, online, or in any other media. These products are called the deliverables of the CAP and PreCAP workflows.

The CAP workflow for journals was implemented from 1996 onwards. When it began, the only deliverable was "S300plus", only for a limited number of full-length-article-like publication item types. S300plus meant that the PDF file of these issues, as well as the full-text SGML, was available electronically — very advanced at the time. This was subsequently scaled up to many more article types, delivered at various stages of the lifetime of the articles. It has by now evolved in a smooth operation, embedded in Elsevier's Global Production.

The PreCAP workflow was set up as a "quick win" beside CAP, in order to get content online fast. Printed journal issues were scanned and by means of OCR technology, SGML files of heads and, later, tails were created, until the journal was ready to move over to CAP. PreCAP still survives today, for journals that are produced outside regular Production workflows, e.g. camera-ready journals and when back volumes need to be brought on line.

A "CAP for books" did not start until 2002, but is now a major component of CAP.

This chapter defines which stages are recognised in the workflow and which version numbering applies to it. It also explains which items are delivered in electronic form.

3.1. Stages

3.1.1. Item stages

Items are the core content of journals and books. Journal articles, book chapters, editorials, indexes, glossaries, advertisements — all these are examples of items. Items are the smallest units that are, or can be, tracked in the workflow.

S5, S100, S200, S250, S280, P100, Q300 and S300 are the existing CAP deliverables for items, and S350 the PreCAP deliverable.

- S5 – The author's input material, accepted by the editorial board.
- S100 – The uncorrected proof.
- S200 – The final, corrected article; still "in press".

- S250 – The final, corrected article, with final publication details (of an issue in progress).
- S280 – The final, corrected article, to support virtual collection (VSI).
- P100 – Proofs of items that are not suitable for online publication before S300, such as indexes and editorial boards.
- Q300 – Proof of the article with final publication details.
- S300 – The article with final publication details.
- S350 – A scanned copy of the printed article.

The letter "S" stands for "stage", "Q" for quality checks, "P" for proof. In order to decide what is an item, the rules of *contents entries* applies, described in Section 3.6.

Each deliverable can be delivered more than once with different version numbers. The precise composition of each deliverable is described in Chapters 5 and 6.

3.1.2. Issue stages

It is common to see a journal issue or a book series volume as a number of items packaged together. For CAP and PreCAP deliverables, however, we use this term also in a somewhat more abstract sense. An issue is the *information* needed to make up the issue. This consists of the issue's properties on the one hand, such as cover date and possible title and editors' names, and the hierarchy of the items that appear in the issue on the other hand.

Each issue that is published is a deliverable of either the CAP or the PreCAP workflow. We distinguish the deliverables H300, H350 and F300, and the deliverables P100 and Q300.

- H300 – The complete issue for electronic publication.
- H350 – The complete issue for electronic publication, derived (scanned) after the fact from the printed issue.
- F300 – The issue cover-to-cover for print publication.
- P100 – Proofs of issue items that are not contents entries (see Section 3.6), as well as items that are not desirable as S100 or S200, such as indexes and editorial boards.
- Q300 – Proof of the complete issue for electronic publication.

The letter "H" refers to "hub", as the main component of the deliverable is the issue hub, which connects the issue with its items in the proper hierarchy and contains the issue data. The letter "F" is associated with "fat" PDF files, i.e., high-quality PDF files suitable for print publication.

Implementation Note. 1. At present, S300 and S350 items will always be delivered as part of a complete issue delivery. Therefore, S300-H300 and S350-H350 deliveries have been defined that contain all the issue hub and all the items. In the future, we will allow item-based resupplies in which items are omitted that have not changed.

Remark. 1. In old terminology, S300 was used to denote the delivery of a complete issue. Formally, this is the deliverable S300-H300: the final and complete issue hub together with all its items.

3.1.3. Book project

The terminology in the books world is more ambiguous than in the journals area. There are volumes, parts, sections, and these may have multiple meanings. When these books (parts, volumes) are scheduled for production, it is decided how they will be produced. A major reference work consisting of three physical volumes can be published at the moment when

one volume is ready, but also as one whole (which is currently the way). A *book project* is the term used for such a deliverable; it can, therefore, comprise one or more physical books, or it can be a continuation of an earlier book project, etc.

The only CAP deliverables for book projects defined today are Q300, H300, H350, O300 and F300.

- Q300 – The book project's main "hub" file with optional PDF proofs of pages of the printed book, all for proofing purposes in Production.
- H300 – The complete book project for electronic publication.
- H350 – The complete book project for electronic publication, derived (scanned) after fact from the printed book(s).
- F300 – The complete book project for print publication.
- O300 – The complete book project for electronic publication as "e-book", i.e. as one PDF file for third-party booksellers.
- E300 – The complete book project in ePub format.

3.1.4. Satellites

Satellites contain information about articles, issues or book chapters, or indeed about images, videos, etc. There are various types of satellites (e.g. annotations, enhancement fragments) which are all delivered with production stage A300.

- A300 – A collection of satellites for electronic publication.

Implementation Note. 1. For now enhancement fragments will be delivered with one satellite per dataset.

3.2. Versions

In the regular workflow, each deliverable (e.g., S100, S200, S250, S280, S300, H300 will ideally be delivered only once, but the CONTRAST standard allows for the possibility of redeliveries.

Each item, therefore, not only possesses a stage, but also a version number.

The version number consists of two components. As of April 2008, the first is the stage of the deliverable, the second is the sequence number of the delivery within that stage, for example `H300.3`.

The convention is subject to change.

Implementation Note. 1. Within every stage every delivery except the first one can be accompanied by a changes-with-respect-to XML file. This changes-with-respect-to file, describing the changes that took place, will be implemented at a later date.

2. In the first implementation of CONTRAST the items' and issue's version within an S300-H300 dataset was identical. With the introduction of stage-based version numbers, the version numbers may vary within the dataset.

Remark. Recipients of content from the Electronic Warehouse should take note of the version numbers in the dataset. Due to hickups in the delivery mechanisms, or at ingestion

at the receiving end, lower versions might arrive after higher versions. The version number decides whether the content should be overwritten.

3.3. File and asset types

CAP and PreCAP deliverables consist of a variety of files. Which files belong to which deliverable depends on the stage. In the dataset.xml, these files are listed under the appropriate subelement of `files-info`.

3.3.1. SGML/XML components (ml)

The main content of items and issue and book hubs is contained in the XML (or SGML) file. However, not all items are fully captured in XML: for some less important items only the head and tail or even only the title are captured. This is called the *weight* of the XML files, see Section 3.5.

`ml` is used for XML files (or older SGML files). The specifications for XML files are divided over various documents. The *Tag by Tag* documentation [4, 5, 6] is the starting point.

XML files are not complete without the external files that they reference, such as strip-ins, marker files, images and electronic components. External files declared in the entity declaration of the XML files are called *assets*. See below for the definition of asset types.

3.3.2. Web PDF files (web-pdf)

All items possess a "web" PDF file that is published online. By definition, the PDF file of an item contains all the pages that contain a portion of the item. If pages contain portions of other material as well, then these are not suppressed: they are also visible. Note that the page ranges may well be non-contiguous if the item is spread over the (printed) issue — in online products, they appear only once at their first occurrence. A special case is when colour images are collected in a colour plate section. The colour plate section itself is rarely a contents entry; the pages with plates belonging to the item are included in the item's PDF file per the definition above.

`web-pdf` is used for PDF files that satisfy the CAP criteria for "web PDF" files. For the web PDF specifications, see [2].

3.3.3. ePub files

`epub` is used for the ePub version of a complete book project.

3.3.4. Print PDF files (print-pdf)

`print-pdf` is used for PDF files that satisfy the CAP criteria for print publication (these files are also known as "fat PDF").

3.3.5. Raw text files (raw-text)

`raw-text` is used for ASCII files with the text of the document. This is a component of a PreCAP delivery.

3.3.6. Asset types

Assets are the external files associated with an XML file declared in the entity declaration, e.g. image files or electronic components, but not strip-ins or markers. Exactly which kind of files may be associated with an item or issue/book hub and which criteria they need to satisfy is described in [7, 8, 9].

The following types of assets are defined.

- `APPLICATION` is used for files belonging to computer applications. It includes Microsoft Word, Microsoft Excel, Adobe PDF files.
- `AUDIO` is used for audio files, such as MP3 or WAV files.
- `IMAGE-CAP` is used for files that satisfy the CAP input specifications for artwork, with the exception of cover images, [1].
- `IMAGE-COVER` is used for images of issue or book covers according to the CAP input specifications.
- `IMAGE-COVER-H150` is used for images of issue or book covers downsized to 150 pixels in height.
- `IMAGE-COVER-H200` is used for images of issue or book covers downsized to 200 pixels in height.
- `IMAGE-COVER-H400` is used for images of issue or book covers downsized to 400 pixels in height.
- `IMAGE-COVER-H768` is used for images of issue or book covers downsized to 768 pixels in height.
- `IMAGE-DOWNSAMPLED` is used for images (of type `IMAGE-CAP`) that were downsampled at the Electronic Warehouse. The downsampling specifications are described in Section 3.7.
- `IMAGE-HIGH-RES` is used for the original artwork, converted to JPEG or GIF format.
- `IMAGE-MMC` is used for the original artwork (e.g. a movie still) that is converted to a thumbnail for an MMC, i.e., an asset which has a `type` not starting with `IMAGE-`.
- `IMAGE-MMC-HIGH-RES` is used to deliver the non-downsampled images from EW.
- `IMAGE-MMC-DOWNSAMPLED` is similar to `IMAGE-DOWNSAMPLED` but it is created from an `IMAGE-MMC`.
- `IMAGE-MMC-THUMBNAIL` is used for the thumbnail images created from images of type `IMAGE-MMC`.
- `IMAGE-NONCAP` is used for artwork files that do not satisfy the specifications for artwork. Images of this type are not touched by the Electronic Warehouse.
- `IMAGE-THUMBNAIL` is used for the thumbnail images created from images of type `IMAGE-CAP`. The specifications for thumbnails are described in Section 3.7.
- `IMAGE-STRIPIN` is used for the stripin image of XML file (i.e. //files-info/ml/purpose = MAIN).
- `IMAGE-PREVIEW` is used for the preview image of web PDF file.
- `VIDEO` is used for movie files, such as MPEG files.
- `VIDEO-FLASH` is used for Flash movie files. These files are the result of a conversion of a movie file that is also present in the dataset.
- `XML` is used for XML assets, such as SVG files.

See also Section 3.7 (p. 24).

3.4. Batches

A batch is a set of items. Some types of batches have an item that acts as the representative of the batch.

Typical applications of batches are add-on items such as short commentaries or questions and answers that follow an item. In this case the item that represents the batch is delivered. The type of this journal item is "with-add-ons": `<journal-item type="with-add-ons">`. The add-on items do not possess a type attribute, similar to any other stand-alone item, but they do have the element `journal-item/journal-item-properties/batch` populated with the PII and DOI of the main item. The item directories are located at the same level as the item directory of the main item.

Another typical application of batches is a section of abstracts (i.e., items of publication item type `ABS`). In Production these are represented by a special item created for this purpose, the batch placeholder item. These placeholders will not be delivered to customers and the batch items will be delivered as regular items. The batch items do not possess a type attribute, but they do have the element `journal-item/journal-item-properties/batch` populated with the PII and DOI of the batch placeholder item. That PII and DOI have to be regarded not as PII and DOI of a regular item; instead, they are identifiers of a "batch" of items. This allows online content repositories receiving the items to group them together, if they would so desire.

Implementation Note. At present, it is required that a batch is delivered complete, i.e. the batch representative together with all the batch items.

Since batch items are in fact items in their own right, they possess an XML file and a web PDF file just like any item, as described in Section 3.3. Hence, the PDF file consists of all item pages that possess a portion of the batch item (portions of the batch representative and/or other batch items may be visible as well).

An item is a batch item if and only if the dataset description file, dataset.xml, contains a `batch` element for the batch item pointing to the representative item. Hence, `batch` must be present for all batch items. Note that the representative item may not be present.

When an item is a batch representative, its "type" (this attribute of `journal-item` is explained in Chapter 5) must be different from the default value `stand-alone`. In the case of add-on items the value must be `with-add-ons`; other values are presently not defined.

The batch representative item possesses a web PDF file that includes the item *and* all batch items.

Note about future expansion. The specification supports batch items whose batch representative is itself a batch item for future expansion. This is to support advanced hierarchies of items.

3.5. Weights of CAP deliverables

CAP item deliverables exist in a number of weights: `FULL-TEXT`, `HEAD-AND-TAIL` (known as CAPLitePlus), and `CONTENTS-ENTRY-ONLY` (known as Ultralight).

The weight indicates to what extent the text of the item is captured in XML. The majority of Elsevier's electronic products is captured in full-text XML, a small number as CAPLite or CAPLitePlus.

It is not true that an XML file with only a head and a tail is necessarily of weight HEAD-AND-TAIL. This can only be verified by comparing the XML file with the PDF file of the item.

The *Tag by Tag* [4] describes which elements in the XML file belong to which weight.

3.5.1. Default serial item weight assignment

The default assignment for weights for items of serial publications (journals and book series) depends on the publication item type (PIT) and the production type and is as given in Table 1. PITs are described in [3].

Table 1: PIT + production type gives default weight

PIT	Production type	Default weight
ADD, BRV, CNF, COR, DIS, EDI, ERR, EXM, FLA, PRP, PRV, REQ, REV, SCO, SSU	NON-CRC	FULL-TEXT or HEAD-AND-TAIL[a]
ADD, BRV, CNF, COR, DIS, EDI, ERR, EXM, FLA, PRP, PRV, REQ, REV, SCO, SSU	CRC	CONTENTS-ENTRY-ONLY or HEAD-AND-TAIL[a]
ABS, ADV, ANN, CAL, CON, EDB, IND, LIT, MIS, NWS, OCN, PNT, PUB	NON-CRC	CONTENTS-ENTRY-ONLY
ABS, ADV, ANN, CAL, CON, EDB, IND, LIT, MIS, NWS, OCN, PNT, PUB	CRC	CONTENTS-ENTRY-ONLY

[a] Depending on the product specification.

In the next section, the concept of contents entries is described. Parts of the issue that are not contents entries, called ancillary material, are not part of any delivery. Weights do not apply to ancillary material, and therefore the rule that heavier-than-default is allowed does not apply either.

3.6. Contents entries

The printed journal and the electronic journal differ in that the printed journal contains cover pages, preliminary pages, etc., that do not belong to the online version. For instance, a web version of the journal may have a homepage for the journal with the Aims and Scope and the Instruction for Authors — in a way, the homepage and the other pages around the content are the online alternatives of the cover and preliminary pages.

To decide which items require electronic delivery, a simple rule of thumb applies: If the item appears in the (paper) table of contents, then it belongs to the electronic delivery. If it does not, it is not delivered. The editorial board, when it appears in the issue, is an exception to this rule: it is always delivered with publication item type EDB whether or not it is listed in the table of contents.

Items that appear in the electronic delivery are called *contents entries*. The remainder of the journal issue consists of *ancillary material*.

For books, the situation is entirely similar: only contents entries are part of the electronic

book. Here, too, there are exceptions: the editorial board and the copyright page are always delivered as part of the book project.

3.7. Downsampled artwork and thumbnails

The Electronic Warehouse delivers downsampled and thumbnail versions of the CAP images it receives from suppliers as part of the deliveries. For downsampled full-size images the format is either GIF or JPEG depending on the input format. For downsampled thumbnail images the format is GIF. The aspect ratio of an output image is the same as that of the original image.

3.7.1. Notations

- DefRes: Default resolution (113 dpi)
- ImXRes: Image resolution in the X direction of the original image
- ImXDim: Image size (in pixels) in the X direction of the original image
- ImYDim: Image size (in pixels) in the Y direction of the original image
- LIW: LargeImageWidth – the width (i.e. the number of pixels in the X direction) of the downsampled full-size image
- W_{th}: Width of thumbnail image (in pixels)
- H_{th}: Height of thumbnail image (in pixels)

3.7.2. Downsampled full-size images

In a CONTRAST-out dataset a downsampled full-size image has asset type `IMAGE-DOWNSAMPLED`. The asset type `IMAGE-MMC-DOWNSAMPLED` is created in the same way.

The width (LIW) of the downsampled full-size image is determined as follows:

$$\mathrm{LIW} = \begin{cases} \mathrm{ImXDim} \times (\mathrm{DefRes}/\mathrm{ImXRes}) & \text{if } \mathrm{ImXRes} \geqslant \mathrm{DefRes}, \\ \mathrm{ImXDim} & \text{if } \mathrm{ImXRes} < \mathrm{DefRes}. \end{cases} \tag{1}$$

The calculated values are rounded down except when this results in a width (LIW) or height of 0, in which case a larger value of LIW is used.

Exception: Images with resolution smaller than the default resolution are not downsampled.

For a common resolution of 300 dpi, the size in pixels will be about 37% of the original. In view of the screen resolution this will result in images that fit on screen quite well.

3.7.3. Downsampled thumbnail images

In a CONTRAST-out dataset a thumbnail image has asset type `IMAGE-THUMBNAIL`. Assets of type `IMAGE-MMC-THUMBNAIL` are created in the same way.

Thumbnail images, or poststamp images, are downsampled from the original images. The width and height (W_{th} and H_{th}) of the downsampled thumbnail image depend on whether the image is a portrait or a landscape image.

A thumbnail of a portrait image (i.e. when $\mathrm{ImXDim} \leqslant \mathrm{ImYDim}$) is 165 pixels high: $H_{\mathrm{th}} = 165$.

A thumbnail of a landscape image (i.e. when $\mathrm{ImXDim} > \mathrm{ImYDim}$) is 219 pixels wide: $W_{\mathrm{th}} = 219$.

Exception: Images with width (ImXDim) less than 219 are not downsampled.

3.7.4. Downsampled cover images

In a CONTRAST-out dataset a cover image has type `IMAGE-COVER-H150` or `IMAGE-COVER-H200`.

A cover image is converted to two GIF files, one of height 200 pixels (`cov200h.gif`) and one of height 150 pixels (`cov150h.gif`). These GIF files do not contain a black border.

3.7.5. Original and high-resolution images

In a CONTRAST-out dataset it is possible to include the original artwork as received from the supplier untouched by any artwork conversion. Such artwork has asset type `IMAGE-CAP`. The original cover image has asset type `IMAGE-COVER` and an original used as an alternative for an electronic component has asset type `IMAGE-MMC`.

Since original artwork comes in formats that are not always suitable for easy access in web browsers, the asset type `IMAGE-HIGH-RES` was introduced. This retains the high quality of the original artwork but converted to GIF or JPEG.

Chapter 4
CONTRAST datasets

CONTRAST (*con*tent *tra*nsport *st*andard) is Elsevier's standard for dataset deliveries to and from the Electronic Warehouse — the successor of EFFECT. A dataset is the name given to a collection of electronic content, transported from one place to the other, in particular from the Electronic Warehouse to online content repositories.

CONTRAST has the following three components:

- The dataset directory structure. The structure and naming conventions are different compared to EFFECT datasets.
- The dataset delivery protocol.
- The accompanying file describing the dataset, called dataset.xml.

The CONTRAST dataset.xml file is an XML file, that validates against a W3C schema. There are separate CONTRAST-out schemas for items and for issues in serial publications as well as for book projects.

4.1. CONTRAST directory structure

Content in a CONTRAST dataset is organized in a directory structure. In this section conventions for the directory structure are described.

4.1.1. Rules versus conventions

It is of utmost importance to bear in mind that these conventions are only there to help human inspection of a dataset. Systems must not draw any conclusion whatsoever from the directory names and the directory structure. The *only* reliable source of information is the dataset.xml file that is part of every dataset.

For instance, the convention may describe that items are stored within a directory based on the PII. The *rule*, however, is to look up the PII in the dataset.xml, and then read the `pathname` of the item, and use the file found there. The system should work equally well if the item resides in a directory `foo`.

Another example: the convention may be that the main XML file is called `main.xml`. Systems must, however, never go to the directory of the item and search for a file with that name. Instead, they must inspect the dataset.xml file and look for the appropriate manifestation. The `pathname` will then lead to the main XML file, even if it is called `rob.xml`.

This small programming overhead makes content transport extensible and leaves room for future expansion or revision.

4.1.2. Which files are listed in the dataset.xml file?

The following files are never listed in the dataset.xml file:

- the dataset.xml file itself, that must be present within the top-level directory of each dataset;
- the strip-in files called in by XML files. From v2015.3, it is allowed to capture strip-in file details as part of CONTRAST OUT dataset.xml.

All other files are mentioned in the dataset.xml.

4.1.3. File and directory names

With *file name* we mean the file name inclusive extension, excluding any directory names. With *path name* we mean the full path name of the file within the dataset (that is, relative to the top-level directory of the dataset).

File and directory names follow standard Unix rules. File and directory names are case sensitive. If it is specified below that a directory is called `main.stripin` then it consists entirely of lowercase characters. If it is specified that the directory name is a PII, then the first letter and the possible `X`s must be uppercase.

While the files must be named with the right case, it should be noted that datasets need to be valid on any computer platform. For this reason it is not allowed to have files or directories within a directory whose names differ only in the case of one of more letters.

4.1.4. Dataset package file

A dataset consists of a directory tree with files on different directory levels. The whole dataset, when delivered to and from the Electronic Warehouse is packed into a ZIP file or a (possibly gzipped) tar file, or, for very large datasets, it can be split over more than one of such files. The package file follows an 8+3 filename convention, because it may have to be transported via restrictive channels, e.g. on an ISO 9660 CDROM. None of the files *within* the dataset need to follow that convention.

Convention. The dataset is named with the last eight characters of the content of `profile-dataset-id`.

No directory within the dataset may be empty.

4.1.5. Dataset base directory

Each dataset must be contained within a single top-level directory, called the *base directory*. (All pathnames are relative to this directory.) The dataset.xml file is present within that directory and is called `dataset.xml`. Hence, if the ZIP or tar file is unpacked, the result is a single directory (with files and subdirectories).

Convention. The top-level directory of the dataset has the same name as the content of `profile-dataset-id`.

4.1.6. XML files and their assets and strip-ins

XML files structured with any one of the DTD 5 family of DTDs come with zero or more strip-ins and zero or more *assets*. Assets are external files (graphic files or electronic components) declared in the entity declaration of the XML file.

Rule. The strip-ins belonging to a file called *file*`.xml` are located in a subdirectory called *file*`.stripin`.

Rule. The assets belonging to a file called *file*.xml are located in a subdirectory called *file*.assets.

If there are no assets or strip-ins, then these directories are not present.

4.1.7. S5, S100, S200, S250 and S280 serial items

Convention. If items in serial publications are part of an S5, S100, S200, S250 or S280 delivery, then they reside in a directory, called the *item directory*, within the base directory. The name of that directory is the PII of the item without dashes and parentheses.

Therefore, a typical item dataset looks like this:

```
jmi00434/
    dataset.xml
    S0022404903002780/main.xml
                      main.pdf
    S0022404903002780/main.stripin/si1.gif
                                    ...
                                    si137.gif
    S0022404903002780/main.assets/gr1.tif
                                   ...
                                   gr17.jpg
```

Convention. If the item is a *subitem* (see Section 3.4) then one of two possibilities may exist. The first possibility is that the subitem travels independently. Then the (sub)item directory resides within the base. The second possibility is that the subitem travels with the item and fellow subitems. In that case the subitem's directory resides within the parent item directory. In all these cases, the conventions for naming by PII and structure still apply.

Implementation Note. The option to let subitems travel alone is reserved for future expansion. Current implementations require the item and its subitems to be despatched in batch and hence only the second possibility mentioned above is allowed.

4.1.8. P100, S300 and S350 items, and serial issues

Convention. In case of an issue in a serial publication, i.e., a journal issue or a volume of a book series, the base directory contains a subdirectory with the ISSN of the serial without punctuation. Within that directory there is a directory, called the *issue directory* with the volume/issue number in VIS format. In this format, a directory name is created from the volume/issue number by prepending the name with a v, replacing the slash by i, and introducing an s before the suppl.

```
"v" vol-first [ "-" vol-last ]
  [ "i" iss-first [ "-" iss-last ] ]
  [ "s" suppl ]
```

Examples:

Volume/issue	Directory name
37C	`v37sC`
37/2	`v37i2`
37I2	`v37sI2`
37/1–3	`v37i1-3`
37S2	`v37sS2`
37–39C	`v37-39sC`
37/1S	`v37i1sS`
37PA	`v37sPA`

P100, S300 and S350 items reside within the issue directory. They are named with the item's PII number without punctuation and follow the conventions for item directories.

The issue hub file (H200, H300 or H350) also resides in the issue directory. The issue hub file may or may not contain (inline) graphics or strip-in structures, so there may be asset and strip-in directories belonging to the hub file (usually called `issue.assets` and `issue.stripin`).

Hence, a typical S300-H300 dataset looks like this (we have left out the subdirectories for brevity):

```
jmi00435/
    dataset.xml
    00224049/v188sC/issue.xml
    00224049/v188sC/issue.assets/cover.tif
    00224049/v188sC/S0022404903002159/...
    00224049/v188sC/S0022404903002068/...
    00224049/v188sC/S0022404903002172/...
    00224049/v188sC/S0022404903002160/...
    00224049/v188sC/S0022404903002081/...
    ...
```

Figure 1 shows the directory structure in an example S300-H300 dataset.

4.1.9. Book projects

Convention. The base directory contains a directory for the book project, named using the unformatted ISBN. Within that directory, the following can be found.

- `main.xml`, the hub file for the book project;
- `changes.xml`, an optional file describing changes with respect to an earlier version after a correction;
- `main.stripin`, the strip-in directory belonging to the hub file;
- `main.assets`, the assets directory belonging to the hub file;
- `changes.stripin`, the strip-in directory belonging to the changes XML file;
- `changes.assets`, the assets directory belonging to the changes XML file;
- `front`, a subdirectory containing all the book item directories within the frontmatter of the book project, named and structured as described above;
- `body`, a subdirectory containing all the book item directories in the body of the book project, named and structured as described above;
- `rear`, a subdirectory containing all the book item directories in the rear of the book project, such as any back-of-the-book index, named and structured as described above;

- `repository`, a subdirectory, with possible subdirectories, containing unlisted additional components, such as PDF files, typeset files and Word files, that may be requested by the Book Production departments.

Hence, a typical Books dataset looks like this (we have left out the subdirectories for brevity):

```
jmi00436/
    dataset.xml
    9780721639505/main.xml
    9780721639505/front/B9780721639505100276/...
    9780721639505/body/B978072163950510001X/...
    9780721639505/body/B9780721639505100021/...
    9780721639505/body/B9780721639505100033/...
    ...
    9780721639505/body/B9780721639505100227/...
    9780721639505/rear/B9780721639505100239/...
    9780721639505/rear/B9780721639505100240/...
    9780721639505/rear/B9780721639505100252/...
    9780721639505/rear/B9780721639505100264/...
    9780721639505/repository/...
```

Figure 1 shows the directory structure in an example Books dataset.

4.2. The dataset.xml file

Each CONTRAST dataset has a dataset.xml file in its base directory. It validates against a schema. There are different CONTRAST-out schemas for items and for issues in serial publications, as well as for book projects.

The dataset.xml file has `dataset` as its top element, with three children: `dataset-unique-ids`, `dataset-properties` and `dataset-content`. The latter element is described in the subsequent chapters, it is different for each deliverable.

Figure 2 shows the top structure of the dataset.xml.

4.2.1. Elements related to dataset identification

dataset-unique-ids

All datasets can be uniquely identified by a set of identifiers, which are contained within the element `dataset-unique-ids`. Alone, the three mandatory subelements `profile-code`, `profile-dataset-id`, `timestamp`, do not uniquely identify the dataset, but together they do.

dataset-unique-ids/profile-code

The element `profile-code` contains the unique code of the Electronic Warehouse profile that the dataset is for.

`profile-code` is an unrestricted W3C schema token, `xs:token`.

dataset-unique-ids/profile-dataset-id

Figure 1: Left – An example of an S300-H300 serial issue dataset directory structure. Right – An example of a Books dataset directory structure.

The element `profile-dataset-id` is the identifier of the dataset. It is an unrestricted W3C schema token, `xs:token`. The identifier must be unique per `profile-code`.

dataset-unique-ids/timestamp

The timestamp when the creation of the dataset was completed is captured with `time-stamp`. Its content is in W3C schema DateTime format, `xs:dateTime` and should be accurate to the second. If the time zone is absent, UTC is assumed. Local time with offset from UTC is also allowed.

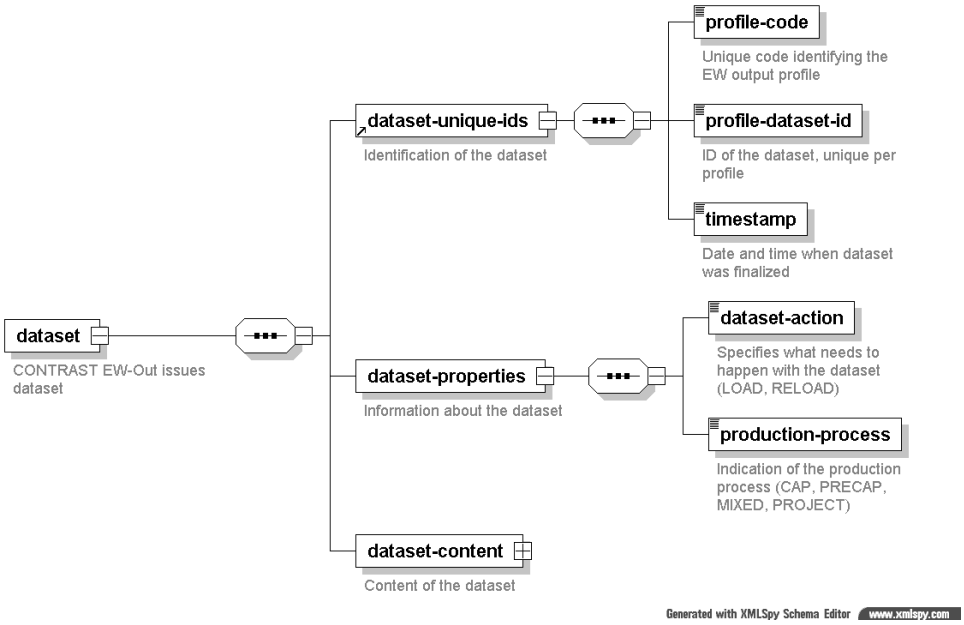

Figure 2: Top structure of the dataset.xml file.

4.2.2. Dataset properties

When a dataset arrives, a certain action must be performed by the recipient. Note that the action relates to the *whole* dataset. It is not possible to have different actions performed on different parts of the dataset.

Deliveries go by *stage*, e.g. S200 or S300. The recipient may decide what to do with existing content of an earlier stage. (The word "stage" is used not just to indicate a point in time, it has become the term for a certain CAP delivery, see Section 3.1.) Within each stage new versions may be despatched. Such a new version is called a *correction*. The correction retains the same stage and same PII, but has, of course, an increased version number.

An *update* takes place when, e.g., a chapter of a book is rewritten. Such an update requires a *correction* of the hub file and a new delivery of a chapter.

dataset-properties/dataset-action

The element `dataset-action` indicates the action to be performed. The value `LOAD` is used under ordinary circumstances. If the dataset is the result of a redelivery request, then the value `RELOAD` is used.

In case of a partial delivery of an issue or book, the value `PARTIAL-RELOAD` is used. Partial deliveries are not possible for S5, S100(-project), S200(-project), P100 and O300 deliveries. Items that are not delivered are still present in the dataset.xml file as usual, except that element `journal-item` or `book-item` has an attribute `omitted` with value "`true`". Files for those items are omitted from the dataset. The hub file is always delivered.

dataset-properties/production-process

The element `production-process` indicates the workflow used for the document or issue.

In datasets that do not contain issues the only possible value is `CAP`. In datasets that do contain issues it can take the following five values:

- `CAP`
- `PRECAP`: used if and only if the delivery is S350 and H350
- `MIXED`: used if the dataset contains items from different workflows
- `CONVERSION`: used if the dataset contains items from a conversion workflow
- `PROJECT`: used if the dataset contains items from a project workflow (which will replace the conversion workflows in the future)
- `EPUB`: used if and only if the delivery is E300
- `OLBS`: used if and only if the delivery is O300

Presently, all CONTRAST-out datasets are delivered with value `CAP`.

4.2.3. Example

XML

```
<dataset
  xmlns="http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2012.1/items"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2012.1/items
     http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-2012.1/items.xsd"
  schema-version="2012.1">

  <dataset-unique-ids>
    <profile-code>SDX</profile-code>
    <profile-dataset-id>SDX01237</profile-dataset-id>
    <timestamp>2012-01-17T12:33:00+02:00</timestamp>
  </dataset-unique-ids>
  <dataset-properties>
    <dataset-action>LOAD</dataset-action>
    <production-process>CAP</production-process>
  </dataset-properties>
  <dataset-content>
    <journal-item>
    .
    .
    .
    </journal-item>
  </dataset-content>
</dataset>
```

Chapter 5
Serial issues and serial items

A serial publication is a journal or a book series. It is divided into "issues": journal issues or book series volumes.

This chapter describes the interaction between Elsevier's Electronic Warehouse and the online content repositories that receive journal content of serial publications. This includes descriptions of the dataset deliveries for each stage for all CAP and PreCAP journals and book series.

There are two CONTRAST-out schemas for journals and book series: one for items (`items.xsd`, see Section 5.1) and one for issues (`issues.xsd`, see Section 5.3).

5.1. Item-only deliveries

The serial item dataset directory structure is defined in Section 4.1.7. In this section we describe what is expected in the dataset and in the dataset.xml.

Below, we traverse the item schema for serial publications starting from the element `dataset-content`. The elements `dataset-unique-ids` and `dataset-properties` are explained in Section 4.2 (p. 30). The element `dataset-content` contains an unlimited number of subelements `journal-item`. Items in the dataset may belong to different journals or book series, but they must be restricted to one content type.

journal-item

The element `journal-item` contains all the information pertaining to an item in a serial publication. It can have two attributes, `type` and `cross-mark`.

The first one, `type`, takes the values `stand-alone` (default), `with-add-ons` and `batch-placeholder`. In Section 3.4 (p. 21) an explanation is given when `type` is used.

The second one, `cross-mark`, takes the values `false` and `true`.

journal-item/version

journal-item/version/version-number

The element `version-number` contains the version number of the item, as described in Section 3.2. The version number is assigned by Elsevier's Electronic Warehouse.

journal-item/version/stage

```
<?xml version="1.0" encoding="UTF-8" ?>
<dataset
  xmlns="http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.6/items"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.6/items
     http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.6/items.xsd"
  schema-version="1.6">

  <dataset-unique-ids>
    <profile-code>SDX</profile-code>
    <profile-dataset-id>SDX68477</profile-dataset-id>
    <timestamp>2006-12-28T09:30:47+02:00</timestamp>
  </dataset-unique-ids>
  <dataset-properties>
    <dataset-action>LOAD</dataset-action>
    <production-process>CAP</production-process>
  </dataset-properties>
  <dataset-content>
    <journal-item>
      <version>
        <version-number>S100.2</version-number>
        <stage>S100</stage>
      </version>
      <journal-item-unique-ids>
        <pii>S0040-4020(03)01057-3</pii>
        <doi>10.1016/S0040-4020(03)01057-3</doi>
        <jid-aid>
          <jid>TET</jid><issn>0040-4020</issn><aid>11699</aid>
        </jid-aid>
      </journal-item-unique-ids>
      <journal-item-properties>
        <pit>FLA</pit>
        <production-type>NON-CRC</production-type>
      </journal-item-properties>
      <files-info>
        <ml>
          <pathname>S0040402003010573/main.xml</pathname>
          <filesize>158554</filesize>
          <purpose>MAIN</purpose>
          <dtd-version>Out-JA 5.0.1 ARTICLE</dtd-version>
          <weight>FULL-TEXT</weight>
          <asset>
            <pathname>S0040402003010573/main.assets/fx1.gif</pathname>
            <filesize>5167</filesize>
            <type>IMAGE-DOWNSAMPLED</type>
          </asset>
          <asset>
            <pathname>S0040402003010573/main.assets/fx1.sml</pathname>
            <filesize>1197</filesize>
            <type>IMAGE-THUMBNAIL</type>
          </asset>
          <asset>
            <pathname>S0040402003010573/main.assets/mmc1.doc</pathname>
            <filesize>374784</filesize>
            <type>APPLICATION</type>
          </asset>
        </ml>
        <web-pdf>
          <pathname>S0040402003010573/main.pdf</pathname>
          <filesize>280050</filesize>
          <purpose>MAIN</purpose>
          <pdf-version>1.4 6.0</pdf-version>
          <pdf-property>DISTILLED OPTIMIZED BOOKMARKED</pdf-property>
        </web-pdf>
      </files-info>
    </journal-item>
  </dataset-content>
</dataset>
```

Figure 3: Sample S100 dataset.xml.

The value of `stage` is one of `S5`, `S100`, `S200`, `S250`, `S280`, `Q300`, `S300`, `S350`. Note that it is possible to receive items of the latter three stages without the accompanying issue hub.

journal-item/journal-item-unique-ids

journal-item/journal-item-unique-ids/pii

The element `pii` contains the (formatted) PII of the item. It must be equal to the PII in the XML file and the PII in the document properties of the main PDF file (if appropriate). Note that there is a variety of patterns that a PII can have.

journal-item/journal-item-unique-ids/doi

The element `doi` contains the DOI of the item, if any. Only items that will appear online may have a DOI. It must be equal to the DOI in the XML file and the DOI in the document properties of the main PDF file (if appropriate).

journal-item/journal-item-unique-ids/jid-aid

The element `jid-aid` contains up to four subelements: `jid` is the Elsevier system code and `issn` the (formatted) ISSN of the serial publication to which the item belongs; `aid` is the Elsevier system article ID and `article-number` is the item's article number.

The first three elements are mandatory while the last one is optional. The `jid`, `aid` and `article-number` must be equal to the same elements in the item's XML file(s). Book series JIDs should begin with "BS:".

Note: Element `article-number` is for now only used in S250 deliveries.

journal-item/journal-item-properties

journal-item/journal-item-properties/pit

The element `pit` contains the publication item type (PIT) of the item. The value of `pit` must be identical to the value of the top-level attribute `docsubtype` in the XML file, except for the case: PITs are always written uppercase in dataset.xml files and lowercase in DTD 5.x files.

journal-item/journal-item-properties/production-type

The element `production-type` can take the values `NON-CRC` and `CRC`.

journal-item/journal-item-properties/collection-title

The element `collection-title` contains the journal title of a journal.

journal-item/journal-item-properties/embargo

The optional element `embargo` contains a date/time before which the item is not allowed to be published. Its content is in W3C schema DateTime format, `xs:dateTime`, and is accurate to the minute. To avoid confusion the date/time is in UTC, signified by a "Z". That is, the embargo date/time is in the following format: "*yyyy*-*mm*-*dd*T*hh*:*mm*:00Z". For more information about embargoes, please contact the Electronic Warehouse.

journal-item/journal-item-properties/sponsored-access

This element, when present, signifies that the item is sponsored, that is, the online version will be available to non-subscribers. It has a subelement `type` which has values in sponsored-access-types-list. The only value currently allowed is `UNLIMITED`. For more information about the sponsored access flag, please contact the Electronic Warehouse.

journal-item/journal-item-properties/delayed-sponsored-article

This optional empty element has an attribute `release-date` and it signifies that the date on which the online platform is to make the item universally accessible.

journal-item/journal-item-properties/delayed-restricted-article

This optional empty element has an attribute `release-date` and it signifies that the date on which the online platform must make a read-only PDF available for downloading.

journal-item/journal-item-properties/funding-body-id

This element is introduced to capture the funding body identifier. If an author chooses for sponsored access he or she has to choose a funding body from the list (see Section 5.2, p. 41). If no funding body is chosen then the value is NONE.

journal-item/journal-item-properties/online-publication-date

This optional element contains the online publication date as reported to the Electronic Warehouse. It is added for future expansion.

journal-item/journal-item-properties/batch

If a journal item is part of a batch, then element `batch` must be present. It contains `pii`, the PII of the journal item that represents the batch, and `doi`, its optional DOI. Note that the item that is referred to, itself an independent item, has the attribute `type` set to a non-default value (see Section 3.4, p. 21).

journal-item/files-info

The element `files-info` contains all the information needed to process the files belonging to the item.

journal-item/files-info/ml

One or two XML files can be associated with the item. These are listed under `ml`, see also Section 3.3 (p. 20). The element is optional; the sole purpose of this is to omit XML files in deliveries for recipients who do not wish to receive them.

journal-item/files-info/ml/pathname

`pathname` is the pathname, relative to dataset.xml (that is, relative to the base directory), of the XML file.

journal-item/files-info/ml/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/ml/filename

filename contains the filename of the XML file.

journal-item/files-info/ml/filesize

Mandatory filesize contains the filesize in bytes of the XML file.

journal-item/files-info/ml/purpose

purpose indicates what the XML file is for. There must always be one XML file with purpose MAIN. The optional second XML file is a changes-with-respect-to file with purpose CHANGES. A third optional XML file is a file containing pagebreak information, the purpose is PAGEBREAK.

journal-item/files-info/ml/dtd-version

The version of the DTD and the top-level element (doctype) used to capture the item is contained in dtd-version. This must, of course, be identical to the declaration in the XML file. The following values may appear: JA 4.5.2 CONVERTED-ARTICLE, JA 5.0.1 ARTICLE, JA 5.0.1 SIMPLE-ARTICLE, JA 5.0.1 BOOK-REVIEW, JA 5.0.1 EXAM, JA 5.0.2 ARTICLE, JA 5.0.2 SIMPLE-ARTICLE, JA 5.0.2 BOOK-REVIEW, JA 5.0.2 EXAM, JA 5.1.0 ARTICLE, JA 5.1.0 SIMPLE-ARTICLE, JA 5.1.0 BOOK-REVIEW, JA 5.1.0 EXAM, JA 5.2.0 ARTICLE, JA 5.2.0 SIMPLE-ARTICLE, JA 5.2.0 BOOK-REVIEW, JA 5.2.0 EXAM, JA 5.4.0 ARTICLE, JA 5.4.0 SIMPLE-ARTICLE, JA 5.4.0 BOOK-REVIEW, JA 5.4.0 EXAM, JA 5.5.0 ARTICLE, JA 5.5.0 SIMPLE-ARTICLE, JA 5.5.0 BOOK-REVIEW, JA 5.5.0 EXAM.

journal-item/files-info/ml/weight

The weight of the XML file can be FULL-TEXT, HEAD-AND-TAIL, HEAD-ONLY or CONTENTS-ENTRY-ONLY. The weight of the item indicates which parts of the text are captured in XML, see Section 3.5 (p. 22).

journal-item/files-info/ml/asset

All assets belonging to the item are listed under asset. This includes all external files referred to by the XML file; it does not include any markers.

journal-item/files-info/ml/asset/pathname

pathname is the pathname, relative to dataset.xml, of the asset file.

journal-item/files-info/ml/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/ml/asset/filename

Mandatory `filename` contains the filename of the asset file.

journal-item/files-info/ml/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

journal-item/files-info/ml/asset/type

The `type` of the asset can have the values `APPLICATION`, `AUDIO`, `IMAGE-CAP`, `IMAGE-DOWNSAMPLED`, `IMAGE-HIGH-RES`, `IMAGE-MMC`, `IMAGE-MMC-HIGH-RES`, `IMAGE-MMC-DOWNSAMPLED`, `IMAGE-MMC-THUMBNAIL`, `IMAGE-NONCAP`, `IMAGE-THUMBNAIL`, `IMAGE-STRIPIN`, `IMAGE-PREVIEW`, `VIDEO`, `VIDEO-FLASH`, `XML`. The values `IMAGE-COVER`, `IMAGE-COVER-H150`, `IMAGE-COVER-H200`, `IMAGE-COVER-H400`, `IMAGE-COVER-H768` will not be used for items. These types are defined in Section 3.3 (p. 20). The choice of assets recipients may receive from the Electronic Warehouse is determined by their output product format.

journal-item/files-info/ml/asset/height

Optional `height` contains the height in bytes of the asset file.

journal-item/files-info/ml/asset/width

Optional `width` contains the width in bytes of the asset file.

journal-item/files-info/web-pdf

An item can have at most two web PDF files associated with it.

journal-item/files-info/web-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the web PDF file.

journal-item/files-info/web-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/web-pdf/filename

`filename` contains the filename of the PDF file.

journal-item/files-info/web-pdf/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

journal-item/files-info/web-pdf/purpose

`purpose` indicates what the web PDF file is for.

If the recipient's product format includes PDF files (it is possible to omit these from the product format) there is always one PDF file with purpose `MAIN`. This contains the PDF file with the content of the item.

The other optional web PDF file can have value MAIN-ABRIDGED. This contains the abridged print version. For more information, please refer to documentation about "e-extra" from the Electronic Warehouse.

journal-item/files-info/web-pdf/pdf-version

pdf-version is the version of PDF. The following versions may appear in current production: 1.7 6.1, 1.7 6.2, 1.7 6.3, 1.7 6.4 and 1.7 6.5 Datasets with legacy content may also contain: 1.1, 1.2, 1.3, 1.4 and 1.4 6.0.

journal-item/files-info/web-pdf/pdf-property

The element pdf-property describes the nature of the web PDF file and has one of the following values: WRAPPED, WRAPPED OPTIMIZED, DISTILLED, DISTILLED BOOKMARKED, DISTILLED OPTIMIZED, DISTILLED OPTIMIZED BOOKMARKED, INTERPRETED BOOKMARKED, INTERPRETED OPTIMIZED, INTERPRETED OPTIMIZED BOOKMARKED.

The values WRAPPED and WRAPPED OPTIMIZED are only allowed if the item's production type is CRC.

If the purpose of the PDF file is not MAIN, then the value DISTILLED OPTIMIZED must be used.

journal-item/files-info/web-pdf/pdf-pages-web

pdf-pages-web contains the total page count of web-pdf.

journal-item/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

journal-item/files-info/web-pdf/asset/pathname

pathname is the pathname, relative to dataset.xml, of the asset file.

journal-item/files-info/web-pdf/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/web-pdf/asset/filename

Mandatory filename contains the filename of the preview image of web-pdf file.

journal-item/files-info/web-pdf/asset/filesize

Mandatory filesize contains the filesize in bytes of the asset file.

journal-item/files-info/web-pdf/asset/type

The type of the asset should have IMAGE-PREVIEW.

journal-item/files-info/web-pdf/asset/height

Optional height contains the height in bytes of the asset file.

journal-item/files-info/web-pdf/asset/width

Optional `width` contains the width in bytes of the asset file.

journal-item/files-info/raw-text

An S350 item also possesses a raw text manifestation, which is obtained by optical character recognition from the scanned PDF file.

journal-item/files-info/raw-text/pathname

`pathname` is the pathname, relative to dataset.xml, of the raw text file.

journal-item/files-info/raw-text/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the raw text file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/raw-text/filename

`filename` contains the filename of the raw text file.

journal-item/files-info/raw-text/filesize

Mandatory `filesize` contains the filesize in bytes of the raw text file.

5.2. Funding Body Identifiers

Table 2 lists the funding body IDs and a short description.

5.3. Issue deliveries

Serial issue deliveries can occur in the following stages: Q300, S300-H300, S350-H350 and F300. The dataset directory structure for serials is defined in Section 4.1.8.

Below, we traverse the "issues" schema for serials starting from the element `dataset-content`. The elements `dataset-unique-ids` and `dataset-properties` are explained in Section 4.2.

An issue schema contains an unlimited number of issues, accompanied by zero or more items. The items must belong to the issue. If the dataset action is `LOAD`, then the issue must be accompanied by *all* its items. An F300 dataset does not have any items. For the other stages, items will only be omitted if the recipient explicitly requested issue hub files only.

journal-issue

The element `journal-issue` contains all the information pertaining to the issue in the dataset. The files associated with the issue are the issue hub with possible assets.

journal-issue/version

Table 2: Funding Body IDs and Description

Code	Description
AHRC	Arts and Humanities Research Council
AHRCPP	Arts and Humanities Research Council
APA	Australian Physiotherapy Association
ARC	Arthritis Research UK
BBSRC	Biotechnology and Biological Sciences Research Council
BBSRCPP	Biotechnology and Biological Sciences Research Council
BCC	Breast Cancer Campaign
BHF	British Heart Foundation
BZG	Brazilian Government
CRUK	Cancer Research UK
CSO	Chief Scientist Office
DFG	Leibniz Publik
DFIDUK	Department for International Development
DHUK	Department of Health UK
DUNHILL	Dunhill Medical Trust
ELS	
EPSRC	Engineering and Physical Sciences Research Council
EPSRCPP	Engineering and Physical Sciences Research Council
ERC	European Research Council
ERCPMC	European Research Council
ESRC	Economic and Social Research Council
ESRCPP	Economic and Social Research Council
EUROB	European Observatory on Health Systems and Policies
FWF	Austrian Science Fund (FWF)
HHMI	
HHMIA	Howard Hughes Medical Institute
MNDA	Motor Neurone Disease Association
MRCUK	Medical Research Council
MRCUKA	Medical Research Council
MRCUKPP	Medical Research Council
NERC	Natural Environment Research Council
NERCPP	Natural Environment Research Council
NIH	
NIHA	National Institutes of Health
NIHGOLD	National Institutes of Health
NONE	
PATINF	
PUK	Parkinson's UK
STFC	Science and Technology Facilities Council
STFCPP	Science and Technology Facilities Council
TCF	The Conservation Fund
TEL	Telethon (Italy)
UCLWT	Wellcome Trust for University College London (UCL) authors
UKPP	
UNU	UNU-WIDER
VSNU	Dutch Universities
WB	
WHO	
WHOGOLD	World Health Organization
WT	Wellcome Trust
WTA	Wellcome Trust

```
<?xml version="1.0" encoding="UTF-8" ?>
<dataset
  xmlns="http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.5/issues"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.5/issues
     http://www.elsevier.com/xml/schema/transport/ew-xcr/journal-1.5/issues.xsd"
  schema-version="1.5">

  <dataset-unique-ids>
    <profile-code>SDYY</profile-code>
    <profile-dataset-id>SDYY04391</profile-dataset-id>
    <timestamp>2006-12-28T13:04:26</timestamp>
  </dataset-unique-ids>
  <dataset-properties>
    <dataset-action>LOAD</dataset-action>
    <production-process>CAP</production-process>
  </dataset-properties>
  <dataset-content>
    <journal-issue>
      <version>
        <version-number>H300.5</version-number>
        <stage>H300</stage>
      </version>
      <journal-issue-unique-ids>
        <pii>S9999-9994(03)X7607-2</pii>
      </journal-issue-unique-ids>
      <journal-issue-properties>
        <jid>MANSC</jid>
        <issn>0276-8976</issn>
        <volume-issue-number>
          <vol-first>11</vol-first>
          <iss-first>1</iss-first>
        </volume-issue-number>
        <isbn>0-7623-1095-2</isbn>
      </journal-issue-properties>
      <files-info>
        <ml>
          <pathname>02768976/v11i1/issue.xml</pathname>
          <filesize>5217</filesize>
          <purpose>MAIN</purpose>
          <dtd-version>SI 5.1.0</dtd-version>
          <asset>
            <pathname>02768976/v11i1/issue.assets/cover150h.gif</pathname>
            <filesize>4389</filesize>
            <type>IMAGE-COVER-H150</type>
          </asset>
          <asset>
            <pathname>02768976/v11i1/issue.assets/cover200h.gif</pathname>
            <filesize>6298</filesize>
            <type>IMAGE-COVER-H200</type>
          </asset>
        </ml>
      </files-info>
    </journal-issue>
    <journal-item>
      <version>
        <version-number>S300.2</version-number>
        <stage>S300</stage>
      </version>
      <journal-item-unique-ids>
        <pii>S0276-8976(04)11001-8</pii>
        <doi>10.1016/S0276-8976(04)11001-8</doi>
        <jid-aid>
          <jid>MANSC</jid>
          <issn>0276-8976</issn>
          <aid>11001</aid>
        </jid-aid>
      </journal-item-unique-ids>
```

Figure 4: Sample issue dataset.xml.

```
    <journal-item-properties>
      <pit>REV</pit>
      <production-type>NON-CRC</production-type>
    </journal-item-properties>
    <files-info>
      <ml>
        <pathname>02768976/v11i1/S0276897604110018/main.xml</pathname>
        <filesize>13478</filesize>
        <purpose>MAIN</purpose>
        <dtd-version>JA 5.0.2 ARTICLE</dtd-version>
        <weight>FULL-TEXT</weight>
        <asset>
          <pathname>02768976/v11i1/S0276897604110018/main.assets/gr1.gif</pathname>
          <filesize>9968</filesize>
          <type>IMAGE-DOWNSAMPLED</type>
        </asset>
        <asset>
          <pathname>02768976/v11i1/S0276897604110018/main.assets/gr1.sml</pathname>
          <filesize>1155</filesize>
          <type>IMAGE-THUMBNAIL</type>
        </asset>
        <asset>
          <pathname>02768976/v11i1/S0276897604110018/main.assets/gr1_lrg.gif</pathname>
          <filesize>23588</filesize>
          <type>IMAGE-HIGH-RES</type>
        </asset>
      </ml>
      <web-pdf>
        <pathname>02768976/v11i1/S0276897604110018/main.pdf</pathname>
        <filesize>882511</filesize>
        <purpose>MAIN</purpose>
        <pdf-version>1.4 6.0</pdf-version>
        <pdf-property>DISTILLED OPTIMIZED BOOKMARKED</pdf-property>
      </web-pdf>
    </files-info>
  </journal-item>
  <journal-item>
      <version>
          <version-number>S300.5</version-number>
          <stage>S300</stage>
      </version>
      <journal-item-unique-ids>
          <pii>S0276-8976(04)11002-X</pii>
          <doi>10.1016/S0276-8976(04)11002-X</doi>
          <jid-aid>
              <jid>MANSC</jid>
              <issn>0276-8976</issn>
              <aid>11002</aid>
          </jid-aid>
      </journal-item-unique-ids>
      <journal-item-properties>
          <pit>REV</pit>
          <production-type>NON-CRC</production-type>
      </journal-item-properties>
      <files-info>
          <ml>
              <pathname>02768976/v11i1/S027689760411002X/main.xml</pathname>
              <filesize>51360</filesize>
              <purpose>MAIN</purpose>
              <dtd-version>Out-JA 5.0.1 ARTICLE</dtd-version>
              <weight>FULL-TEXT</weight>
              <asset>
                  <pathname>02768976/v11i1/S027689760411002X/main.assets/gr1.gif</pathname>
                  <filesize>38859</filesize>
                  <type>IMAGE-DOWNSAMPLED</type>
              </asset>
              <asset>
                  <pathname>02768976/v11i1/S027689760411002X/main.assets/gr1.sml</pathname>
                  <filesize>2672</filesize>
                  <type>IMAGE-THUMBNAIL</type>
              </asset>
```

Figure 5: Sample issue dataset.xml (continued).

```
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr1_lrg.gif</pathname>
                    <filesize>123764</filesize>
                    <type>IMAGE-HIGH-RES</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr2.jpg</pathname>
                    <filesize>103546</filesize>
                    <type>IMAGE-DOWNSAMPLED</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr2.sml</pathname>
                    <filesize>12787</filesize>
                    <type>IMAGE-THUMBNAIL</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr2_lrg.jpg</pathname>
                    <filesize>330546</filesize>
                    <type>IMAGE-HIGH-RES</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr3.jpg</pathname>
                    <filesize>142780</filesize>
                    <type>IMAGE-DOWNSAMPLED</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr3.sml</pathname>
                    <filesize>5780</filesize>
                    <type>IMAGE-THUMBNAIL</type>
                </asset>
                <asset>
                    <pathname>02768976/v11i1/S027689760411002X/main.assets/gr3_lrg.jpg</pathname>
                    <filesize>348365</filesize>
                    <type>IMAGE-HIGH-RES</type>
                </asset>
            </ml>
            <web-pdf>
                <pathname>02768976/v11i1/S027689760411002X/main.pdf</pathname>
                <filesize>231478</filesize>
                <purpose>MAIN</purpose>
                <pdf-version>1.4 6.0</pdf-version>
                <pdf-property>DISTILLED OPTIMIZED BOOKMARKED</pdf-property>
            </web-pdf>
        </files-info>
    </journal-item>
    .
    .
    .
  </dataset-content>
</dataset>
```

Figure 5: Sample S300-H300 dataset.xml (continued).

journal-issue/version/version-number

The element `version-number` contains the version number of the issue, as described in Section 3.2. The version number is assigned by Elsevier's Electronic Warehouse.

journal-issue/version/stage

The value of `stage` is one of `Q300`, `H300`, `H350`, `F300`.

journal-issue/journal-issue-unique-ids

journal-issue/journal-issue-unique-ids/pii

The element `pii` contains the PII of the issue. It must be equal to the PII in the XML hub file. Note that there is a variety of formats for PIIs.

journal-issue/journal-issue-unique-ids/doi

The element `doi` contains the DOI of the issue, if any. Only issues that will appear online may have a DOI. It must be equal to the DOI in the XML hub file.

journal-issue/journal-issue-properties

journal-issue/journal-issue-properties/jid

The element `jid` contains the Elsevier system code of the serial publication to which the issue belongs.

journal-issue/journal-issue-properties/issn

The element `issn` contains the ISSN of the serial publication to which the issue belongs.

journal-issue/journal-issue-properties/volume-issue-number

The volume/issue number of the issue is captured in `volume-issue-number`. It consists of `vol-first`, `vol-last`, `iss-first`, `iss-last` and `suppl`.

journal-issue/journal-issue-properties/collection-title

The element `collection-title` contains the journal title of a journal.

journal-issue/journal-issue-properties/embargo

The optional element `embargo` contains a date/time before which the issue is not allowed to be published. Its content is in W3C schema DateTime format, `xs:dateTime`, and is accurate to the minute. To avoid confusion the date/time is in UTC, signified by a "Z". That is, the embargo date/time is in the following format: "*yyyy*-*mm*-*dd*T*hh*:*mm*:00Z". For more information about embargoes, please contact the Electronic Warehouse.

journal-issue/journal-issue-properties/online-publication-date

This optional element contains the online publication date as reported to the Electronic Warehouse. It is added for future expansion.

journal-issue/journal-issue-properties/isbn

The serial issue's optional ISBN is captured with `isbn`. There can be two `isbn` elements for delivery of both ISBN-10 and ISBN-13; the latter is always present.

journal-issue/files-info

The element `files-info` contains all the information needed to process the files belonging to the issue. It consists of an `ml` element and an optional list of `web-pdf` elements, or, in case of an F300 delivery, an `ml` element optionally followed by a list of `print-pdf` elements.

journal-issue/files-info/ml

There can be one or two XML files associated with the issue, distinguished by their `purpose`. Their details are listed under `ml`, which consists of either a `pathname` followed by `purpose`, `dtd-version` and an optional `asset`, or, in case of an F300 delivery, of a `pathname` followed by `schema-version` and an optional `asset`.

journal-issue/files-info/ml/pathname

`pathname` is the pathname, relative to dataset.xml (that is, relative to the base directory), of the hub XML file.

journal-issue/files-info/ml/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the hub XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-issue/files-info/ml/filename

`filename` contains the filename of the XML file.

journal-issue/files-info/ml/filesize

Mandatory `filesize` contains the filesize in bytes of the XML file.

journal-issue/files-info/ml/purpose

The element `purpose` contains the value `MAIN` (this is the issue hub), or `AUXILIARY` (optional).

journal-issue/files-info/ml/dtd-version

The version of the DTD of the hub file is contained in `dtd-version`. The version must, of course, be identical to the declaration in the XML file. It has values `SI 5.1.0`, `SI 5.2.0`, `SI 5.4.0` and `SI 5.5.0`.

journal-issue/files-info/ml/schema-version

Element `schema-version` is used in datasets that contain a delivery for a printer. It contains the schema version of the print hub (print.xml), i.e. the file describing the "fat" PDFs meant for the printer. The version must, of course, be identical to the declaration in the XML file. The allowed values are `print 1.0`, `print 1.1`, `print 1.2`, `print 1.3` and `print 1.4`.

journal-issue/files-info/ml/asset

All assets belonging to the item are listed under `asset`. This includes all external files declared in the XML file;

journal-issue/files-info/ml/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

journal-issue/files-info/ml/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-issue/files-info/ml/asset/filename

filename contains the filename of the asset file.

journal-issue/files-info/ml/asset/filesize

Mandatory filesize contains the filesize in bytes of the asset file.

journal-issue/files-info/ml/asset/type

The type of the asset can have the values APPLICATION, AUDIO, IMAGE-CAP, IMAGE-COVER, IMAGE-COVER-H150, IMAGE-COVER-H200, IMAGE-COVER-H400, IMAGE-COVER-H768, IMAGE-DOWNSAMPLED, IMAGE-HIGH-RES, IMAGE-MMC, IMAGE-MMC-HIGH-RES, IMAGE-MMC-DOWNSAMPLED, IMAGE-MMC-THUMBNAIL, IMAGE-NONCAP, IMAGE-THUMBNAIL, IMAGE-STRIPIN, IMAGE-PREVIEW, VIDEO, VIDEO-FLASH, XML. These types are defined in Section 3.3 (p. 20).

journal-issue/files-info/ml/asset/height

Optional height contains the height in bytes of the asset file.

journal-issue/files-info/ml/asset/width

Optional width contains the width in bytes of the asset file.

journal-issue/files-info/web-pdf

Beside the hub XML file, there can be either an unlimited number of issue-level PDF files or an unlimited number of print PDF files. If there are web PDF files then they are captured under this element. These web PDF files are untyped, i.e. there is no subelement purpose. This is restricted to Q300 deliveries.

journal-issue/files-info/web-pdf/pathname

pathname is the pathname, relative to dataset.xml, of the web PDF file.

journal-issue/files-info/web-pdf/ucs-locator

ucs-locator is the VTWs Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-issue/files-info/web-pdf/filename

filename contains the filename of the PDF file.

journal-issue/files-info/web-pdf/filesize

Mandatory filesize contains the filesize in bytes of the PDF file.

journal-issue/files-info/web-pdf/pdf-version

The untyped issue-level PDFs possess a version, `pdf-version`, whose allowed values are `1.1`, `1.2`, `1.3`, `1.4`, `1.4 6.0`, `1.7 6.1`, `1.7 6.2`, `1.7 6.3`, `1.7 6.4` and `1.7 6.5`

journal-issue/files-info/web-pdf/pdf-pages-web

`pdf-pages-web` contains the total page count of web-pdf.

journal-issue/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

journal-issue/files-info/web-pdf/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

journal-issue/files-info/web-pdf/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-issue/files-info/web-pdf/asset/filename

Mandatory `filename` contains the filename of the preview image of web-pdf file.

journal-issue/files-info/web-pdf/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

journal-issue/files-info/web-pdf/asset/type

The `type` of the asset should have `IMAGE-PREVIEW`.

journal-issue/files-info/web-pdf/asset/height

Optional `height` contains the height in bytes of the asset file.

journal-issue/files-info/web-pdf/asset/width

Optional `width` contains the width in bytes of the asset file.

journal-issue/files-info/print-pdf

Alternatively, in F300 datasets, instead of web PDF files there is an unlimited number of print PDFs ("fat PDFs"). Print PDF files and their accompanying XML file are described elsewhere.

journal-issue/files-info/print-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the print PDF file.

journal-issue/files-info/print-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the print PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-issue/files-info/print-pdf/filename

`filename` contains the filename of the PDF file.

journal-issue/files-info/print-pdf/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

journal-issue/files-info/print-pdf/purpose

The element `purpose` contains the value `SUITABILITY-DIGITAL`, or `SUITABILITY-NONE` or `SUITABILITY-OFFSET-AND-DIGITAL`.

journal-issue/files-info/print-pdf/pdf-version

`pdf-version` is the version of the print-PDF file, the allowed values are `1.3 1.0`, and `1.6 2.0`.

journal-issue/files-info/print-pdf/pdf-property

The element `pdf-property` contains the value `SCANNED`, or `NOT SCANNED`.

journal-issue/files-info/print-pdf/pdf-pages-web

`pdf-pages-web` contains the total page count of print-pdf.

journal-item

The element `journal-item` contains all the information pertaining to an item in a serial publication. It can have three attributes, `type`, `omitted` and `cross-mark`.

The first one, `type`, takes the values `stand-alone` (default), `with-add-ons` and `batch-placeholder`. In Section 3.4 (p. 21) an explanation is given when `type` is used.

The second one, `omitted`, takes the values `false` (default) and `true`. (See also p. 32 for information on partial deliveries.)

The third one, `cross-mark`, takes the values `false` and `true`.

journal-item/version

journal-item/version/version-number

The element `version-number` contains the version number of the item, as described in Section 3.2. The version number is assigned by Elsevier and is included in the order.

journal-item/version/stage

The value of `stage` is `Q300`, `S300` or `S350`.

journal-item/journal-item-unique-ids

journal-item/journal-item-unique-ids/pii

The element `pii` contains the PII of the item. It must be equal to the PII in the XML file and the PII in the document properties of the main PDF file (if appropriate). Note that PIIs can have a variety of formats.

journal-item/journal-item-unique-ids/doi

The element `doi` contains the DOI of the item, if any. Only items that will appear online may have a DOI. It must be equal to the DOI in the XML file and the DOI in the document properties of the main PDF file (if appropriate).

journal-item/journal-item-unique-ids/jid-aid

The element `jid-aid` contains up to four subelements: `jid` is the Elsevier system code and `issn` the (formatted) ISSN of the serial publication to which the item belongs; `aid` is the Elsevier system article ID and `article-number` is the item's article number.

The first three elements are mandatory while the last one is optional. The `jid`, `aid` and `article-number` must be equal to the same elements in the item's XML file(s). Book series JIDs should begin with "BS:".

Note: Element `article-number` is for now only used in S250 deliveries.

journal-item/journal-item-properties

journal-item/journal-item-properties/pit

The element `pit` contains the publication item type of the item. The value of `pit` must be identical to the value of the top-level attribute `docsubtype` in the XML file, except for the case: PITs are always written uppercase in dataset.xml files and lowercase in DTD 5.x files.

journal-item/journal-item-properties/production-type

The element `production-type` can take the values `NON-CRC` and `CRC`.

journal-item/journal-item-properties/embargo

The optional element `embargo` contains a date/time before which the issue is not allowed to be published. Its content is in W3C schema DateTime format, `xs:dateTime`, and is accurate to the minute. To avoid confusion the date/time is in UTC, signified by a "Z". That is, the embargo date/time is in the following format: "*yyyy-mm-dd*T*hh*:*mm*:00Z". For more information about embargoes, contact the Electronic Warehouse.

journal-item/journal-item-properties/sponsored-access

This element signifies that the item is sponsored, that is, the online version will be available to non-subscribers. It has a subelement `type` which has values in sponsored-access-types-list. The only value currently allowed is "UNLIMITED".

journal-item/journal-item-properties/delayed-sponsored-article

This optional empty element has an attribute `release-date` and it signifies that the date on which the online platform is to make the item universally accessible.

journal-item/journal-item-properties/delayed-restricted-article

This optional empty element has an attribute `release-date` and it signifies that the date on which the online platform must make a read-only PDF available for downloading.

journal-item/journal-item-properties/funding-body-id

This element is introduced to capture the funding body identifier. If an author chooses for sponsored access he or she has to choose a funding body from the list (see Section 5.2, p. 41). If no funding body is chosen then the value is NONE.

journal-item/journal-item-properties/online-publication-date

This optional element contains the online publication date as reported to the Electronic Warehouse. It is added for future expansion.

journal-item/journal-item-properties/batch

If a journal item is part of a batch, then element `batch` must be present. It contains `pii`, the PII of the journal item that represents the batch, and `doi`, its optional DOI. Note that the item that is referred to, itself an independent item, has the attribute `type` set to a non-default value (see Section 3.4, p. 21).

journal-item/files-info

The element `files-info` contains all the information needed to process the files belonging to the item.

journal-item/files-info/ml

Up to 32 XML files can be associated with the item. When present these are listed under `ml`. In practice only few XML files will be delivered. Always present is the “main” XML file and optional changes-with-respect-to and auxiliary files.

journal-item/files-info/ml/pathname

`pathname` is the pathname, relative to dataset.xml, of the XML file.

journal-item/files-info/ml/ucs-locator

`ucs-locator` is the VTW’s Unified Cloud Storage URL of the XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/ml/filename

`filename` contains the filename of the XML file.

journal-item/files-info/ml/filesize

Mandatory `filesize` contains the filesize in bytes of the XML file.

journal-item/files-info/ml/purpose

purpose indicates what the XML file is for. There must always be one XML file with purpose MAIN. This can be accompanied by a changes-with-respect-to file with purpose CHANGES. A third optional XML file is a file containing pagebreak information, the purpose is PAGEBREAK.

RadCon metadata files are delivered with purpose AUXILIARY.

The purposes ORIGINAL, LINKED-TEXTBOX and ELAN are for internal EW use only.

journal-item/files-info/ml/dtd-version

The version of the DTD and the top-level element (doctype) used to capture the item is contained in dtd-version. This must, of course, be identical to the declaration in the XML file. The following values may appear (other versions are for internal EW use only): JA 4.5.2 CONVERTED-ARTICLE, JA 5.0.1 ARTICLE, JA 5.0.1 SIMPLE-ARTICLE, JA 5.0.1 BOOK-REVIEW, JA 5.0.1 EXAM, JA 5.0.2 ARTICLE, JA 5.0.2 SIMPLE-ARTICLE, JA 5.0.2 BOOK-REVIEW, JA 5.0.2 EXAM, JA 5.1.0 ARTICLE, JA 5.1.0 SIMPLE-ARTICLE, JA 5.1.0 BOOK-REVIEW, JA 5.1.0 EXAM, JA 5.2.0 ARTICLE, JA 5.2.0 SIMPLE-ARTICLE, JA 5.2.0 BOOK-REVIEW, JA 5.2.0 EXAM, JA 5.4.0 ARTICLE, JA 5.4.0 SIMPLE-ARTICLE, JA 5.4.0 BOOK-REVIEW, JA 5.4.0 EXAM, JA 5.5.0 ARTICLE, JA 5.5.0 SIMPLE-ARTICLE, JA 5.5.0 BOOK-REVIEW, JA 5.5.0 EXAM.

This is not the case for RadCon metadata files. These files are structured according to a MicroSoft Infopath schema and the dtd-version is INFOPATH.

journal-item/files-info/ml/weight

The weight of the XML file can be FULL-TEXT, HEAD-AND-TAIL, HEAD-ONLY or CONTENTS-ENTRY-ONLY. The weight of the item indicates which parts of the text are captured in XML.

journal-item/files-info/ml/asset

All assets belonging to the item are listed under asset. This includes all external files declared in the XML file.

journal-item/files-info/ml/asset/pathname

pathname is the pathname, relative to dataset.xml, of the asset file.

journal-item/files-info/ml/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/ml/asset/filename

Mandatory filename contains the filename of the asset file.

journal-item/files-info/ml/asset/filesize

Mandatory filesize contains the filesize in bytes of the asset file.

journal-item/files-info/ml/asset/type

The type of the asset can have the values APPLICATION, AUDIO, IMAGE-CAP, IMAGE-DOWNSAMPLED, IMAGE-HIGH-RES, IMAGE-MMC, IMAGE-MMC-HIGH-RES, IMAGE-MMC-DOWNSAMPLED, IMAGE-MMC-THUMBNAIL, IMAGE-NONCAP, IMAGE-THUMBNAIL, IMAGE-STRIPIN, IMAGE-PREVIEW, VIDEO, VIDEO-FLASH, XML. The values IMAGE-COVER, IMAGE-COVER-H150, IMAGE-COVER-H200, IMAGE-COVER-H400, IMAGE-COVER-H768 will not be used for items. These types are defined in Section 3.3.

journal-item/files-info/ml/asset/height

Optional height contains the height in bytes of the asset file.

journal-item/files-info/ml/asset/width

Optional width contains the width in bytes of the asset file.

journal-item/files-info/web-pdf

An item can have at most two web PDF files associated with it.

journal-item/files-info/web-pdf/pathname

pathname is the pathname, relative to dataset.xml, of the web PDF file.

journal-item/files-info/web-pdf/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/web-pdf/filename

filename contains the filename of the PDF file.

journal-item/files-info/web-pdf/filesize

Mandatory filesize contains the filesize in bytes of the PDF file.

journal-item/files-info/web-pdf/purpose

purpose indicates what the web PDF file is for.

There is always one PDF file with purpose MAIN. This contains the PDF file of the item.

The other optional web PDF file can have value MAIN-ABRIDGED. This contains the abridged print version. For more information, please refer to documentation about "e-extra" avaiable from the Electronic Warehouse.

journal-item/files-info/web-pdf/pdf-version

pdf-version is the version of PDF, the allowed values are 1.1, 1.2, 1.3, 1.4, 1.4 6.0, 1.7 6.1, 1.7 6.2, 1.7 6.3, 1.7 6.4 and 1.7 6.5

journal-item/files-info/web-pdf/pdf-property

The element pdf-property describes the nature of the web PDF file and has one of the following nine values: WRAPPED, WRAPPED OPTIMIZED, DISTILLED, DISTILLED BOOKMARKED. DISTILLED OPTIMIZED, DISTILLED OPTIMIZED BOOKMARKED, INTERPRETED, INTERPRETED OPTIMIZED, INTERPRETED OPTIMIZED BOOKMARKED.

journal-item/files-info/web-pdf/pdf-pages-web

pdf-pages-web contains the total page count of web-pdf.

journal-item/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

journal-item/files-info/web-pdf/asset/pathname

pathname is the pathname, relative to dataset.xml, of the asset file.

journal-item/files-info/web-pdf/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/web-pdf/asset/filename

Mandatory filename contains the filename of the preview image of web-pdf file.

journal-item/files-info/web-pdf/asset/filesize

Mandatory filesize contains the filesize in bytes of the asset file.

journal-item/files-info/web-pdf/asset/type

The type of the asset should have IMAGE-PREVIEW.

journal-item/files-info/web-pdf/asset/height

Optional height contains the height in bytes of the asset file.

journal-item/files-info/web-pdf/asset/width

Optional width contains the width in bytes of the asset file.

journal-item/files-info/raw-text

An S350 item also possesses a raw text manifestation, which is obtained by optical character recognition from the scanned PDF file.

journal-item/files-info/raw-text/pathname

pathname is the pathname, relative to dataset.xml, of the raw text file.

journal-item/files-info/raw-text/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the raw text file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/raw-text/filename

`filename` contains the filename of the raw text file.

journal-item/files-info/raw-text/filesize

Mandatory `filesize` contains the filesize in bytes of the raw text file.

Chapter 6
Book projects and book items

This chapter describes the interaction between Elsevier's Electronic Warehouse and the online content repositories that receive book content. It describes all the elements of the transport schema for book projects and book items.

There are two CONTRAST-out schemas for book item-only and complete book : one for items (`book-items.xsd`, see Section **??**) and one for complete book (`book-projects.xsd`, see Section 6.2).

6.1. Item-only deliveries

In this chapter, we traverse the Electronic Warehouse output schema for books starting from the top element `dataset-content`.

The elements `dataset-unique-ids` and `dataset-properties` are explained in Section 4.2.

The element `dataset-content` is the container element for all of the metadata covering all of the items in the book dataset delivery. It consists of one or more book-item elements belonging to each book being delivered.

book-item

The element book-item contains all of the metadata elements needed to uniquely identify the book item as well as the location of the book item's asset files in the dataset. It contains the following elements: `version`, `book-item-unique-ids`, `book-item-properties`, and `files-info`.

It can have one attribute, `omitted`, which takes values `false` (default) and `true`. The latter value is to be used if an item is omitted from the dataset. It should only be used if the `dataset-action` has value `PARTIAL-RELOAD`. (See also p. 32 for information on partial deliveries.)

book-item/version

book-item/version/version-number

The element `version-number` contains the version number of the item, as described in Section 3.2. The version number is assigned by Elsevier.

book-item/version/stage

The possible values of the stage are S200, S280, S300 and S350.

book-item/book-item-unique-ids

The element book-item-unique-ids contains a required pii element, with an optional doi element. Each should be properly formatted with dashes, slashes and parentheses and should be identical to the same identifiers in the XML and PDF files.

book-item/book-item-unique-ids/book-project-unique-ids

The element book-project-unique-ids contains a pii element, an optional doi element, an isbn element and an isbn-list element containing ISBN information. The IDs will be formatted.

book-item/book-item-unique-ids/book-project-unique-ids/isbn

This element contains the ISBN of the book. It can be an ISBN-10 or an ISBN-13 and it will be formatted.

book-item/book-item-unique-ids/book-project-unique-ids/isbn-list

A book project can have different ISBNs attached to it, each with a purpose. This is contained in one or more isbn-info elements.

book-item/book-item-unique-ids/book-project-unique-ids/isbn-list/isbn-info

The ISBN information is stored in element isbn (containing the actual ISBN) and element purpose (containing its purpose). The ISBN will be an ISBN-13. The possible purposes are: CDROM (a CD, DVD or equivalent with data on it), HARDBOUND, NONBOOK (not a book, but a poster, free gift, calendar, etc.), ONLINE (a website of some sort), PAPERBACK, SET-COMBO (a set which contains books and something else), SET-MULTIVOL (a classic set of books, not a multivolume single book), SET-NONBOOK (e.g. a set of DVDs or slides without books) and EPUB.

book-item/book-item-properties

The element book-item-properties contains three elements: pit (optional), production-type, and online-publication-date.

book-item/book-item-properties/pit

The pit element is a list of different types of items which can appear in books, similar to journal Publishing Item Types. The list of possible values is contained in the "pit-list" in the schema book-item-project.xsd. The value of pit must be identical to the value of the top-level attribute docsubtype in the XML file. (Except for the case, that is. PITs are always written uppercase in dataset.xml files and lowercase in DTD 5.x files.)

book-item/book-item-properties/production-type

The production-type element contains one of two possible values: CRC or NON-CRC.

book-item/book-item-properties/collection-title

The element `collection-title` contains the title of the book.

book-item/book-item-properties/online-publication-date

This optional element contains the online publication date as reported to the Electronic Warehouse. It is added for future expansion.

book-item/files-info

The element `files-info` contains all the information needed to process the files belonging to the item.

book-item/files-info/ml

One or two XML files can be associated with the item. These are listed under `ml`.

book-item/files-info/ml/pathname

`pathname` is the pathname, relative to dataset.xml, of the XML file.

book-item/files-info/ml/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/ml/filename

`filename` contains the filename of the XML file.

book-item/files-info/ml/filesize

Mandatory `filesize` contains the filesize in bytes of the XML file.

book-item/files-info/ml/purpose

`purpose` indicates what the XML file is for. There must always be one XML file with purpose `MAIN`. An optional second XML file is a changes-with-respect-to file with purpose `CHANGES`. An optional third file is a file containing pagebreak information, the purpose is `PAGEBREAK`.

book-item/files-info/ml/dtd-version

The version of the DTD and the top-level element (doctype) used to capture the item is contained in `dtd-version`. This must, of course, be identical to the declaration in the XML file.

book-item/files-info/ml/weight

The weight of the XML file can be `FULL-TEXT`, `HEAD-AND-TAIL` or `CONTENTS-ENTRY-ONLY`. The weight of the item indicates which parts of the text are captured in XML.

book-item/files-info/ml/asset

All assets belonging to the item are listed under `asset`. This includes all external files declared in the XML file, it does not include any strip-ins.

book-item/files-info/ml/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

book-item/files-info/ml/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/ml/asset/filename

Mandatory `filename` contains the filename of the asset file.

book-item/files-info/ml/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

book-item/files-info/ml/asset/type

The `type` of the asset can have the values `APPLICATION`, `IMAGE-CAP`, `IMAGE-NONCAP`, `IMAGE-DOWNSAMPLED`, `IMAGE-THUMBNAIL`, `IMAGE-HIGH-RES`, `IMAGE-STRIPIN`, `IMAGE-PREVIEW`, `AUDIO`, `VIDEO`, `XML`. The values `IMAGE-COVER`, `IMAGE-COVER-H150`, `IMAGE-COVER-H200`, `IMAGE-COVER-H400`, `IMAGE-COVER-H768` will not be used. These types are defined in Section 3.3.

book-item/files-info/ml/asset/height

Optional `height` contains the height in bytes of the asset file.

book-item/files-info/ml/asset/width

Optional `width` contains the width in bytes of the asset file.

book-item/files-info/web-pdf

An S300 item has at most one web PDF file associated with it, the main PDF file which contains the item. The element is optional to allow for deliveries without PDF files.

book-item/files-info/web-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the web PDF file.

book-item/files-info/web-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/web-pdf/filename

`filename` contains the filename of the PDF file.

book-item/files-info/web-pdf/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

book-item/files-info/web-pdf/purpose

`purpose` indicates what the web PDF file is for, it must be `MAIN`.

book-item/files-info/web-pdf/pdf-version

`pdf-version` is the version of PDF, the allowed values are `1.1`, `1.2`, `1.3`, `1.4`, `1.4 6.0`, `1.7 6.1`, `1.7 6.2`, `1.7 6.3`, `1.7 6.4` and `1.7 6.5`.

book-item/files-info/web-pdf/pdf-property

The element `pdf-property` describes the nature of the web PDF file and has one of the following nine values: `WRAPPED`, `WRAPPED OPTIMIZED`, `DISTILLED`, `DISTILLED BOOKMARKED`. `DISTILLED OPTIMIZED`, `DISTILLED OPTIMIZED BOOKMARKED`, `INTERPRETED`, `INTERPRETED OPTIMIZED`, `INTERPRETED OPTIMIZED BOOKMARKED`.

The value `WRAPPED` is only allowed if the book item's production type is `CRC`.

If the purpose of the book item is not `MAIN`, then the value `DISTILLED OPTIMIZED` must be used.

book-item/files-info/web-pdf/pdf-pages-web

`pdf-pages-web` contains the total page count of web-pdf.

book-item/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

book-item/files-info/web-pdf/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

book-item/files-info/web-pdf/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/web-pdf/asset/filename

Mandatory `filename` contains the filename of the preview image of web-pdf file.

book-item/files-info/web-pdf/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

book-item/files-info/web-pdf/asset/type

The `type` of the asset should have `IMAGE-PREVIEW`.

book-item/files-info/web-pdf/asset/height

Optional `height` contains the height in bytes of the asset file.

book-item/files-info/web-pdf/asset/width

Optional `width` contains the width in bytes of the asset file.

book-item/files-info/raw-text

A book item can also possess a raw text manifestation, which is obtained by optical character recognition from the scanned PDF file. The element is optional.

book-item/files-info/raw-text/pathname

`pathname` is the pathname, relative to dataset.xml, of the raw text file.

book-item/files-info/raw-text/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the raw text file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/raw-text/filename

`filename` contains the filename of the raw text file.

book-item/files-info/raw-text/filesize

Mandatory `filesize` contains the filesize in bytes of the raw text file.

6.2. Book projects

In this chapter, we traverse the Electronic Warehouse output schema for books starting from the top element `dataset-content`.

The elements `dataset-unique-ids` and `dataset-properties` are explained in Section 4.2.

The element `dataset-content` is the container element for all of the metadata covering all of the items in the book dataset delivery. It consists of precisely one book-project element and one or more book-item elements belonging to each book being delivered.

book-project

A book project schema contains an unlimited number of book projects. The element `book-project` contains all of the metadata elements needed to uniquely identify the book. It also contains the location of the book's hub file in the dataset. It contains the following elements: `version`, `book-project-unique-ids`, `book-project-properties`, and `files-info`.

Sample tagging of a book project is found in Fig. 6 (p. 68).

book-project/version

book-project/version/version-number

This element contains the version number of the book-project level as described in Section 3.2 (p. 19). The version number is assigned by Elsevier's Electronic Warehouse.

book-project/version/stage

The book project can have the following stages: H300, H350, Q300, O300, F300 and E300.

book-project/book-project-unique-ids

The element `book-project-unique-ids` contains a `pii` element, an optional `doi` element, an `isbn` element and an `isbn-list` element containing ISBN information. The IDs will be formatted.

book-project/book-project-unique-ids/isbn

This element contains the ISBN of the book. It can be an ISBN-10 or an ISBN-13 and it will be formatted.

book-project/book-project-unique-ids/isbn-list

A book project can have different ISBNs attached to it, each with a purpose. This is contained in one or more `isbn-info` elements.

book-project/book-project-unique-ids/isbn-list/isbn-info

The ISBN information is stored in element `isbn` (containing the actual ISBN) and element `purpose` (containing its purpose). The ISBN will be an ISBN-13. The possible purposes are: `CDROM` (a CD, DVD or equivalent with data on it), `HARDBOUND`, `NONBOOK` (not a book, but a poster, free gift, calendar, etc.), `ONLINE` (a website of some sort), `PAPERBACK`, `SET-COMBO` (a set which contains books and something else), `SET-MULTIVOL` (a classic set of books, not a multivolume single book), `SET-NONBOOK` (e.g. a set of DVDs or slides without books) and `EPUB`.

book-project/book-project-properties

The element `book-project-properties` contains mandatory `working-title` and `edition` elements, with optional `prim-auth-surname` and `book-parent` elements.

book-project/book-project-properties/working-title

The `working-title` element contains the title of the book (present to assist human readability and processing of the dataset.xml file). This version of the title is not to be confused with the official title for the book, which is in the hub file and may already be present in the EW as basedata.

book-project/book-project-properties/collection-title

The element `collection-title` contains the title of the book.

book-project/book-project-properties/edition

The edition element contains a numerical value and not text version of the edition number for the book. Textual representations of the edition will live only in the hub file.

book-project/book-project-properties/prim-auth-surname

The prim-auth-surname element contains the surname of the book's primary author or editor. Within editorial and book production, books are often referred to in this manner (e.g., the "Watson" book).

book-project/book-project-properties/book-parent

The book-parent element (with child elements either isbn or issn) is only used when a book project belongs to a larger collection of books.

book-project/book-project-properties/online-publication-date

This optional element contains the online publication date as reported to the Electronic Warehouse. It is added for future expansion.

book-project/files-info

The element files-info contains one or two ml subelements and a number of web-pdf, print-pdf or epub subelements.

book-project/files-info/ml

The element ml contains mandatory pathname, purpose, and dtd-version elements, with optional and repeatable asset elements. This element describes the hub XML file or a pagebreak file, depending on the purpose.

book-project/files-info/ml/pathname

pathname is the pathname, relative to dataset.xml (that is, relative to the base directory), of the book project hub file.

book-project/files-info/ml/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the book project hub XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/ml/filename

filename contains the filename of the XML file.

book-project/files-info/ml/filesize

Mandatory filesize contains the filesize in bytes of the XML file.

book-project/files-info/ml/purpose

The purpose element contains the purpose of the item. There must be always one XML file with purpose MAIN. An optional second file is a file containing pagebreak information, the purpose is PAGEBREAK.

book-project/files-info/ml/dtd-version

The version of the DTD of the hub file is contained in `dtd-version`. The version must, of course, be identical to the declaration in the XML file.

Main XML files can have the following versions: `EHS-BOOKS 5.1.0 EHS-BOOK`, `EHS-BOOKS 5.1.1 EHS-BOOK`, `BOOK 5.2.0 BOOK` (not used), `BOOK 5.2.1 BOOK`, `BOOK 5.3.0 BOOK`, `BOOK 5.3.1 BOOK`, `BOOK 5.4.0 BOOK`, `BOOK 5.5.0 BOOK`, `BOOK-METADATA 5.0.0 BOOK-METADATA`, `BOOK-METADATA 5.0.1 BOOK-METADATA` and `BOOK-EPUB 5.0.0 BOOK-EPUB`.

The Book-Metadata versions appear if and only if the stage is O300. The Book-Epub version appears only if the stage is E300.

The pagebreak file has the following version: `PAGEBREAK 5.0.0`.

book-project/files-info/ml/schema-version

Element `schema-version` is used in datasets that contain a delivery for a printer. It contains the schema version of the print hub (print.xml), i.e. the file describing the "fat" PDFs meant for the printer. The version must, of course, be identical to the declaration in the XML file. The allowed values are `print 1.0`, `print 1.1`, `print 1.2`, `print 1.3` and `print 1.4`.

book-project/files-info/ml/asset

All assets belonging to the item are listed under `asset`. This includes all external files declared in the XML file.

book-project/files-info/ml/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

book-project/files-info/ml/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/ml/asset/filename

`filename` contains the filename of the asset file.

book-project/files-info/ml/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

book-project/files-info/ml/asset/type

The `type` of the asset can have the values `APPLICATION`, `IMAGE-CAP`, `IMAGE-NONCAP`, `IMAGE-COVER`, `IMAGE-DOWNSAMPLED`, `IMAGE-HIGH-RES`, `IMAGE-MMC`, `IMAGE-MMC-HIGH-RES`, `IMAGE-MMC-DOWNSAMPLED`, `IMAGE-MMC-THUMBNAIL`, `IMAGE-THUMBNAIL`, `IMAGE-COVER-H150`, `IMAGE-COVER-H200`, `IMAGE-COVER-H400`, `IMAGE-COVER-H768`, `IMAGE-THUMBNAIL`, `IMAGE-STRIPIN`, `AUDIO`, `VIDEO`, `XML`. These types are defined in Section 3.3.

book-project/files-info/ml/asset/height

Optional `height` contains the height in bytes of the asset file.

book-project/files-info/ml/asset/width

Optional `width` contains the width in bytes of the asset file.

book-project/files-info/web-pdf

A book project can have a number of web PDF files associated with it, ususally one PDF file containing the complete book project.

book-project/files-info/web-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the web PDF file.

book-project/files-info/web-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/web-pdf/filename

`filename` contains the filename of the PDF file.

book-project/files-info/web-pdf/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

book-project/files-info/web-pdf/purpose

`purpose` indicates what the web PDF file is for, it must be `COMPLETE`, `COMPLETE-PF` and `COMPLETE-CE`.

book-project/files-info/web-pdf/pdf-version

`pdf-version` is the version of PDF, the allowed values are `1.1`, `1.2`, `1.3`, `1.4`, `1.4 6.0`, `1.7 6.1`, `1.7 6.2`, `1.7 6.3`, `1.7 6.4` and `1.7 6.5`.

book-project/files-info/web-pdf/pdf-property

The element `pdf-property` describes the nature of the web PDF file and has one of the following values: `WRAPPED`, `WRAPPED OPTIMIZED`, `DISTILLED`, `DISTILLED BOOKMARKED`, `DISTILLED OPTIMIZED`, `DISTILLED OPTIMIZED BOOKMARKED`, `INTERPRETED`, `INTERPRETED OPTIMIZED`, `INTERPRETED OPTIMIZED BOOKMARKED`.

book-project/files-info/web-pdf/pdf-pages-web

`pdf-pages-web` contains the total page count of web-pdf.

book-project/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

book-project/files-info/web-pdf/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

book-project/files-info/web-pdf/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/web-pdf/asset/filename

Mandatory `filename` contains the filename of the preview image of web-pdf file.

book-project/files-info/web-pdf/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

book-project/files-info/web-pdf/asset/type

The `type` of the asset should have `IMAGE-PREVIEW`.

book-project/files-info/web-pdf/asset/height

Optional `height` contains the height in bytes of the asset file.

book-project/files-info/web-pdf/asset/width

Optional `width` contains the width in bytes of the asset file.

book-project/files-info/print-pdf

In an F300 delivery the book project has a number of print PDF files associated with it.

book-project/files-info/print-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the print PDF file.

book-project/files-info/print-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the print PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/print-pdf/filename

`filename` contains the filename of the PDF file.

book-project/files-info/ml/print-pdf/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

book-project/files-info/print-pdf/purpose

The element `purpose` contains the value `SUITABILITY-DIGITAL`, or `SUITABILITY-NONE` or `SUITABILITY-OFFSET-AND-DIGITAL`.

```
<book-project>
  <version>
    <version-number>H300.1</version-number>
    <stage>H300</stage>
  </version>
  <book-project-unique-ids>
    <pii>B978-0-323-01195-2.X0000-2</pii>
    <isbn>978-0-323-01195-2</isbn>
  </book-project-unique-ids>
  <book-project-properties>
    <working-title>Mosby's Clinical Nursing</working-title>
    <edition>5</edition>
    <prim-auth-surname>Thompson</prim-auth-surname>
  </book-project-properties>
  <files-info>
    <ml>
      <pathname>9780323011952/main.xml</pathname>
      <filesize>28344</filesize>
      <purpose>MAIN</purpose>
      <dtd-version>EHS-BOOKS 5.1.1 EHS-BOOK</dtd-version>
    </ml>
  </files-info>
</book-project>
```

Figure 6: Example tagging of a book project.

book-project/files-info/print-pdf/pdf-version

`pdf-version` is the version of the print-PDF file, the allowed values are `1.3 1.0`, and `1.6 2.0`.

book-project/files-info/print-pdf/pdf-property

The element `pdf-property` contains the value `SCANNED`, or `NOT SCANNED`.

book-project/files-info/print-pdf/pdf-pages-web

`pdf-pages-web` contains the total page count of print-pdf.

book-project/files-info/epub

In an E300 delivery the book project has a number of ePub files associated with it (usually one).

book-project/files-info/epub/pathname

`pathname` is the pathname, relative to dataset.xml, of the epub file.

book-project/files-info/epub/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the epub file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-project/files-info/ml/epub/filename

`filename` contains the filename of the epub file.

book-project/files-info/ml/epub/filesize

Mandatory `filesize` contains the filesize in bytes of the PDF file.

book-project/files-info/ml/epub/purpose

`purpose` indicates what the ePub file is for, it must be `MAIN`.

book-item

The element book-item contains all of the metadata elements needed to uniquely identify the book item as well as the location of the book item's asset files in the dataset. It contains the following elements: `version`, `book-item-unique-ids`, `book-item-properties`, and `files-info`.

It can have one attribute, `omitted`, which takes values `false` (default) and `true`. The latter value is to be used if an item is omitted from the dataset. It should only be used if the `dataset-action` has value `PARTIAL-RELOAD`. (See also p. 32 for information on partial deliveries.)

Sample tagging of `book-item` can be found in Fig. 7 (p. 74).

book-item/version

book-item/version/version-number

The element `version-number` contains the version number of the item, as described in Section 3.2. The version number is assigned by Elsevier.

book-item/version/stage

The value of `stage` is `S300`.

book-item/book-item-unique-ids

The element `book-item-unique-ids` contains a required `pii` element, with an optional `doi` element. Each should be properly formatted with dashes, slashes and parentheses and should be identical to the same identifiers in the XML and PDF files.

book-item/book-item-properties

The element `book-item-properties` contains three elements: `branch`, `pit` (optional), and `production-type`.

book-item/book-item-properties/branch

The `branch` element is used to identify in which portion of the book hierarchy the item occurs. There are three possible values: `FRONT`, `BODY` or `REAR`.

book-item/book-item-properties/pit

The pit element is a list of different types of items which can appear in books, similar to journal Publishing Item Types. The list of possible values is contained in the "pit-list" in the schema book-item-project.xsd. The value of pit must be identical to the value of the top-level attribute docsubtype in the XML file. (Except for the case, that is. PITs are always written uppercase in dataset.xml files and lowercase in DTD 5.x files.)

book-item/book-item-properties/production-type

The production-type element contains one of two possible values: CRC or NON-CRC.

book-item/files-info

The element files-info contains all the information needed to process the files belonging to the item.

book-item/files-info/ml

One or two XML files can be associated with the item. These are listed under ml.

book-item/files-info/ml/pathname

pathname is the pathname, relative to dataset.xml, of the XML file.

book-item/files-info/ml/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the XML file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/ml/filename

filename contains the filename of the XML file.

book-item/files-info/ml/filesize

Mandatory filesize contains the filesize in bytes of the XML file.

book-item/files-info/ml/purpose

purpose indicates what the XML file is for. There must always be one XML file with purpose MAIN. An optional second XML file is a changes-with-respect-to file with purpose CHANGES. An optional third file is a file containing pagebreak information, the purpose is PAGEBREAK.

book-item/files-info/ml/dtd-version

The version of the DTD and the top-level element (doctype) used to capture the item is contained in dtd-version. This must, of course, be identical to the declaration in the XML file.

book-item/files-info/ml/weight

The weight of the XML file can be FULL-TEXT, HEAD-AND-TAIL or CONTENTS-ENTRY-ONLY. The weight of the item indicates which parts of the text are captured in XML.

book-item/files-info/ml/asset

All assets belonging to the item are listed under `asset`. This includes all external files declared in the XML file.

book-item/files-info/ml/asset/pathname

`pathname` is the pathname, relative to dataset.xml, of the asset file.

book-item/files-info/ml/asset/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/ml/asset/filename

Mandatory `filename` contains the filename of the asset file.

book-item/files-info/ml/asset/filesize

Mandatory `filesize` contains the filesize in bytes of the asset file.

book-item/files-info/ml/asset/type

The `type` of the asset can have the values `APPLICATION`, `IMAGE-CAP`, `IMAGE-NONCAP`, `IMAGE-DOWNSAMPLED`, `IMAGE-THUMBNAIL`, `IMAGE-HIGH-RES`, `IMAGE-STRIPIN`, `IMAGE-PREVIEW`, `AUDIO`, `VIDEO`, `XML`. The values `IMAGE-COVER`, `IMAGE-COVER-H150`, `IMAGE-COVER-H200`, `IMAGE-COVER-H400`, `IMAGE-COVER-H768` will not be used. These types are defined in Section 3.3.

book-item/files-info/ml/asset/height

Optional `height` contains the height in bytes of the asset file.

book-item/files-info/ml/asset/width

Optional `width` contains the width in bytes of the asset file.

book-item/files-info/web-pdf

An S300 item has at most one web PDF file associated with it, the main PDF file which contains the item. The element is optional to allow for deliveries without PDF files.

book-item/files-info/web-pdf/pathname

`pathname` is the pathname, relative to dataset.xml, of the web PDF file.

book-item/files-info/web-pdf/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the web PDF file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/web-pdf/filename

filename contains the filename of the PDF file.

book-item/files-info/web-pdf/filesize

Mandatory filesize contains the filesize in bytes of the PDF file.

book-item/files-info/web-pdf/purpose

purpose indicates what the web PDF file is for, it must be MAIN.

book-item/files-info/web-pdf/pdf-version

pdf-version is the version of PDF, the allowed values are 1.1, 1.2, 1.3, 1.4, 1.4 6.0, 1.7 6.1, 1.7 6.2, 1.7 6.3, 1.7 6.4 and 1.7 6.5,

book-item/files-info/web-pdf/pdf-property

The element pdf-property describes the nature of the web PDF file and has one of the following nine values: WRAPPED, WRAPPED OPTIMIZED, DISTILLED, DISTILLED BOOKMARKED. DISTILLED OPTIMIZED, DISTILLED OPTIMIZED BOOKMARKED, INTERPRETED, INTERPRETED OPTIMIZED, INTERPRETED OPTIMIZED BOOKMARKED.

The value WRAPPED is only allowed if the book item's production type is CRC.

If the purpose of the book item is not MAIN, then the value DISTILLED OPTIMIZED must be used.

book-item/files-info/web-pdf/pdf-pages-web

pdf-pages-web contains the total page count of web-pdf.

book-item/files-info/web-pdf/asset

The preview image of web-pdf first page is listed under asset.

book-item/files-info/web-pdf/asset/pathname

pathname is the pathname, relative to dataset.xml, of the asset file.

book-item/files-info/web-pdf/asset/ucs-locator

ucs-locator is the VTW's Unified Cloud Storage URL of the asset file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

book-item/files-info/web-pdf/asset/filename

Mandatory filename contains the filename of the preview image of web-pdf file.

book-item/files-info/web-pdf/asset/filesize

Mandatory filesize contains the filesize in bytes of the asset file.

book-item/files-info/web-pdf/asset/type

The type of the asset should have IMAGE-PREVIEW.

book-item/files-info/web-pdf/asset/height

Optional `height` contains the height in bytes of the asset file.

book-item/files-info/web-pdf/asset/width

Optional `width` contains the width in bytes of the asset file.

book-item/files-info/raw-text

A book item can also possess a raw text manifestation, which is obtained by optical character recognition from the scanned PDF file. The element is optional.

book-item/files-info/raw-text/pathname

`pathname` is the pathname, relative to dataset.xml, of the raw text file.

book-item/files-info/raw-text/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the raw text file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

journal-item/files-info/raw-text/filename

`filename` contains the filename of the raw text file.

book-item/files-info/raw-text/filesize

Mandatory `filesize` contains the filesize in bytes of the raw text file.

```
<book-item>
  <version>
    <version-number>S300.1</version-number>
    <stage>S300</stage>
  </version>
  <book-item-unique-ids>
    <pii>B978-0-323-01195-2.00002-8</pii>
  </book-item-unique-ids>
  <book-item-properties>
    <branch>BODY</branch>
    <pit>CHP</pit>
    <production-type>NON-CRC</production-type>
  </book-item-properties>
  <files-info>
    <ml>
      <pathname>9780323011952/body/B9780323011952000028/main.xml</pathname>
      <filesize>54335</filesize>
      <purpose>MAIN</purpose>
      <dtd-version>EHS-BOOKS 5.1.1 CHAPTER</dtd-version>
      <weight>FULL-TEXT</weight>
      <asset>
        <pathname>9780323011952/body/B9780323011952000028/⌒
          main.assets/gr1.jpg</pathname>
        <filesize>14544</filesize>
        <type>IMAGE-DOWNSAMPLED</type>
      <asset>
      <asset>
        <pathname>9780323011952/body/B9780323011952000028/⌒
          main.assets/gr1.sml</pathname>
        <filesize>3390</filesize>
        <type>IMAGE-THUMBNAIL</type>
      <asset>
    </ml>
    <web-pdf>
      <pathname>9780323011952/body/B9780323011952000028/main.pdf</pathname>
      <filesize>87661</filesize>
      <purpose>MAIN</purpose>
      <pdf-version>1.4 6.0</pdf-version>
      <pdf-property>DISTILLED OPTIMIZED BOOKMARKED</pdf-property>
    </web-pdf>
  </files-info>
</book-item>
```

Figure 7: Example tagging of a book-item.

Chapter 7
Satellite items

This chapter describes the interaction between Elsevier's Electronic Warehouse and the online content repositories that receive satellites via Contrast-out. It describes all the elements of the transport schema for satellite items.

7.1. Satellites

In this chapter, we traverse the Electronic Warehouse output schema for satellites starting from the top element `dataset-content`.

The elements `dataset-unique-ids` and `dataset-properties` are explained in Section 4.2 (p. 30).

The element `dataset-content` is the container element for all of the metadata covering all of the items in the satellite dataset delivery. It consists of one or more `satellite` elements.

satellite

Information on the satellite is stored in four subelements: `version`, `satellite-unique-ids`, `satellite-properties` and `files-info`.

satellite/version

satellite/version/version-number

This element contains the version number of the satellite as described in Section 3.2 (p. 19). The version number is assigned by Elsevier's Electronic Warehouse.

satellite/version/stage

The stage of a satellite is always `A300`.

satellite/satellite-unique-ids

The satellite is uniquely identified by a URI. This is stored in subelement `uri`.

satellite/satellite-properties/satellite-group

Satellites are divided into groups, for example `ANNOTATION`. The group is stored in `satellite-group`.

satellite/satellite-properties/satellite-code

Satellites are further subdivided into types which are represented by a code. This code is stored in `satellite-code`.

satellite/satellite-properties/parent-collection

A satellite belongs to an article, an issue or a book chapter. The optional element `parent-collection` contains the PII, ISSN or ISBN of the collection (i.e. the journal or book) that article, issue or book chapter is part of.

In case the satellite belongs to an article or an issue, the collection is identified by a PII, an ISSN or both (subelements `pii` and `issn`). In case the satellite belongs to a book chapter, the collection is identified by an ISBN-13 (subelement `isbn`).

satellite/satellite-properties/linked-issues

This optional element contains the issue(s) the satellite is linked to. They are identified in one or more subelements `pii`.

satellite/satellite-properties/linked-items

This optional element contains the item(s) the satellite is linked to. They are identified in one or more subelements `pii`.

satellite/files-info

The element `files-info` contains all the information needed to process the files belonging to the item.

satellite/files-info/ml

One XML file is associated with the item. This is listed under `ml`. This element describes the location and purpose of all the hub XML file and its assets. The element `ml` contains mandatory elements `pathname`, `purpose`, and either `dtd-version` or `schema-version`, followed by optional and repeatable `asset` elements.

satellite/files-info/ml/pathname

`pathname` is the pathname, relative to dataset.xml, of the satellite file.

satellite/files-info/ml/ucs-locator

`ucs-locator` is the VTW's Unified Cloud Storage URL of the satellite file. The max occurrence of element ucs-locator is set to 2 to enable capture both the location.

satellite/files-info/ml/filename

`filename` contains the filename of the XML file.

satellite/files-info/ml/filesize

Mandatory `filesize` contains the filesize in bytes of the XML file.

satellite/files-info/ml/purpose

purpose indicates what the XML file is for. For satellite files the purpose is always SATELLITE.

dataset-content/satellite/files-info/ml/dtd-version

This element is used if the satellite files adhere to a DTD. Currently there is only one possible value, EF 5.0.0 ENHANCEMENT-FRAGMENT.

dataset-content/satellite/files-info/ml/schema-version

This element is used if the satellite files adhere to a W3C schema. It contains the namespace of the schema.

```
<schema-version>http://www.elsevier.com/xml/schema/
    rdf/Lancet-JMi-1/jmi_v010.xsd</schema-version>
```

satellite/files-info/ml/asset

All assets belonging to the item are listed under asset. This includes all external files declared in the XML file.

The pathname of the asset, relative to dataset.xml, is stored in subelement pathname. The type of the asset is stored in subelement type.

Chapter 8
Dataset delivery protocol

This chapter deals with the protocol for delivering CONTRAST datasets from the Electronic Warehouse.

8.1. Network delivery from the Electronic Warehouse

The method used for network delivery is FTP transfer according to the pull model, in which the EW makes datasets available via FTP on a dedicated area on the Electronic Warehouse drop zone, and subsequently — after the transfer of each dataset is fully complete — a so-called *ready message* which is stored next to the dataset on the EW drop zone. Data recipients scan their areas ("buckets") for ready messages, and act upon the information found in the ready messages. Details on the ready messages, which contain some basic information about the dataset in an XML format, are given below.

The complete dataset including its dataset.xml is packed into one file according to either the zip or the tar format (where the latter may be compressed using gzip), as described in Chapter 4.1. Its filename is equal to the last eight characters of the dataset identifier affixed with one of the extensions `.zip`, `.tar` or `.tgz`, for example `x0001829.zip`. It is not possible to use the same dataset identifier twice. It has been agreed that the dataset identifier contains at most 12 characters.

If the size of the delivery is very large, it is possible to split the dataset over more than one ZIP, tar or gzipped tar file. These files must be valid ZIP, tar or gzipped tar files in their own right, i.e. methods such as ZIP spanning or physically cutting a file in pieces are not to be used.

8.2. CD, DVD, tape

The same structure is found on other delivery media: a "ready" file and a ZIP, tar or gzipped tar file.

8.3. Ready messages

The ready messages are in an XML format which validates against a very simple W3C schema, depicted in Fig. 8.

- `dataset-ready`, the top-level element;
- `dataset-unique-ids`, element similar to the one in the transport schema, containing two subelements identifying the dataset as well as the file name of the dataset, i.e. it contains the same information as the dataset.xml with exception of the timestamp;
- `dataset-package-file`, with two subelements identifying the ZIP or tar file;
- `filename`, the filename of the package file;
- `md5`, the MD5 checksum of the dataset package file.

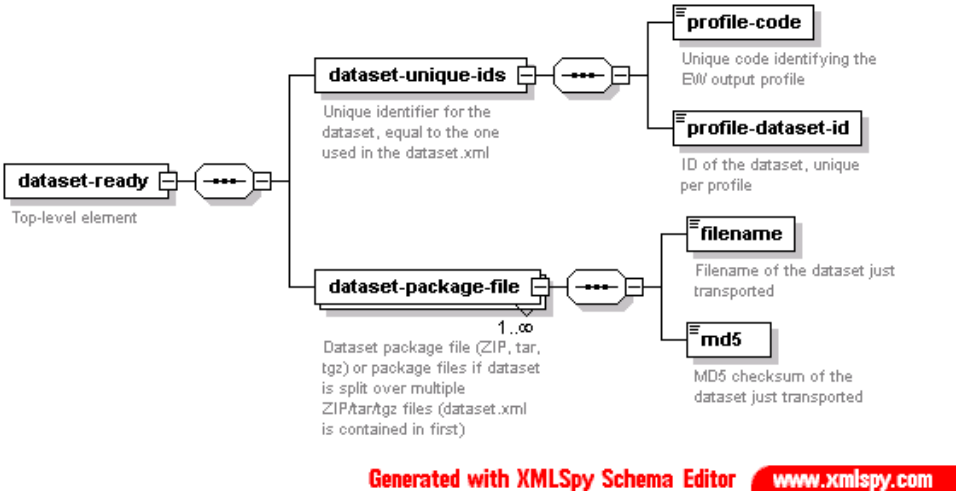

Figure 8: Ready XML schema.

The content of `profile-code` and `profile-dataset-id` is case-sensitive, which means that the dataset ID as used in the ready message should match the ID used in the dataset.xml exactly, and the name of the package file put on the EW drop zone should match the file name given in the ready message exactly.

The file name of the ready message is identical to the dataset identifier, followed by the extension "`.ready.xml`". The following is an example of a network delivery ready message:

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<dataset-ready
  xmlns="http://www.elsevier.com/xml/schema/transport/ew-xcr/ready-3.0/ready"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://www.elsevier.com/xml/schema/transport/ew-xcr/ready-3.0/ready
     http://www.elsevier.com/xml/schema/transport/ew-xcr/ready-3.0/ready.xsd"
  version="3.0">

  <dataset-unique-ids>
    <profile-code>SDX</profile-code>
    <profile-dataset-id>SDX001245437</profile-dataset-id>
  </dataset-unique-ids>

  <dataset-package-file>
    <filename>01245437.tar</filename>
    <md5>58f6fe00b23d175815fb1a18105bbf9e</md5>
  </dataset-package-file>

</dataset-ready>
```

The file name of this message would be `sdx001245437.ready.xml`.

If the dataset is delivered in more than one file, the first one that is mentioned in the ready message contains the dataset.xml file. Note that the filenames of these files should still have eight characters. The following format is suggested (using the above example):

`1245437A.tar`, `1245437B.tar`, etc. A ready message could for instance contain the following:

XML

```
<dataset-package-file>
  <filename>HSAR091A.tar</filename>
  <md5>58f6fe00b23d175815fb1a18105bbf9e</md5>
</dataset-package-file>
<dataset-package-file>
  <filename>HSAR091B.tar</filename>
  <md5>f8e07026e61dcc0643fc99830c0d25cb</md5>
</dataset-package-file>
<dataset-package-file>
  <filename>HSAR091C.tar</filename>
  <md5>ff7ac07d5d8c65c579c5d714bb6ba3c5</md5>
</dataset-package-file>
```

Bibliography

[1] CAP specs for artwork, see Supplier Artwork Extranet.
[2] Specification for Elsevier "Web" PDF files (pdfreq40.pdf)
[3] Publication item types (rs98022.pdf).
[4] Tag by Tag Documentation of the Common Element Pool and the JA DTD (2004).
[5] Tag by Tag Documentation of the Serial Issue DTD (2004).
[6] Tag by Tag Documentation of the Health Sciences DTD (2004).
[7] The CAP Guide for MFC Activities (2003).
[8] MultiMedia Components (MMCs) – instructions for full-service suppliers (2003).
[9] Handling supplementary files in DTD 5.0 – instructions for full-service suppliers (2003).
[10] TBA
[11] Article copyright lines – by status and PIT (2003).
[12] Online publication dates in print (2004).

Index