eClips Web
Technical Specification V4.2

Introduction

This document provides a comprehensive technical description of the eClips Web service.

It is intended for use by organisations wishing to receive the eClips Web service, including media monitoring organisations and content aggregators. It includes both high-level and detailed technical information.

Product description

eClips Web is part of the eClips product, an online service that provides media monitoring organisations (MMOs) and content aggregators with digital news content. eClips Web specifically delivers content from UK newspaper websites in a timely and accurate manner.

The content in eClips Web is collected directly from publishers' content management systems (CMS), cleansed, standardised, and archived in a consistent data structure. Collection and processing of content is carried out close to real-time and excludes non-article content such as adverts and navigational pages.

This ingestion process delivers improvements in completeness, accuracy, timeliness, and reliability when compared to other services which rely on page scraping to deliver similar content. The process is also subject to detailed monitoring and analysis to assure the continued quality of the archive.

This assurance allows MMOs and aggregators using eClips Web to deliver high quality monitoring solutions to their customers (end users). Elements of the service exposed to end users are specifically designed to support their information needs whilst minimising the technical complexity to which they are exposed.

Service overview

eClips Web uses a service-oriented architecture (SOA) with four functional components.

Component	Function
Payload Orchestrator	Returns article details in standardised XML
Redirector	Determines the appropriate method for viewing an article
Article Orchestrator	Renders the article in HTML or PDF
Authentication Engine	Authenticates the user for access to the specified content

Components must be accessed through SSL secure connection and therefore use the HTTPS protocol.

Articles & Versions

Within eClips Web, articles and article versions are considered differently.

An article is a page published by a CMS to a title's website.

An article version is a representation of the properties and contents of that article in any given update (including its creation).

Users interacting with eClips Web will receive article versions but may use the concept of an article to group article versions together.

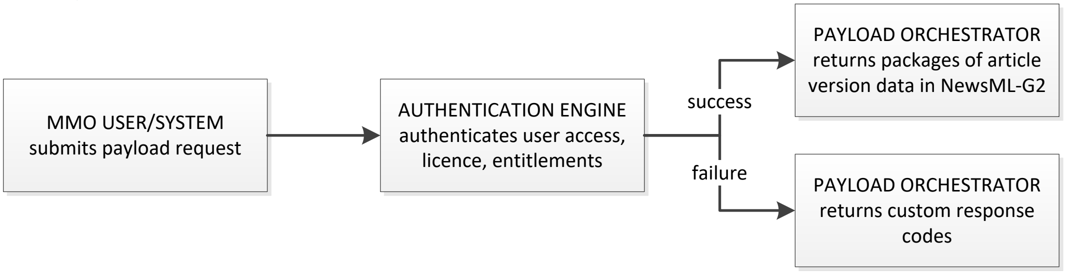
Payload Orchestrator

Payload Orchestrator is a RESTful API, which can be queried manually or programmatically using a set of defined parameters. It is accessible by MMOs and content aggregators and is designed to support machine interrogation and interpretation.

Payload Orchestrator returns article version details in a standardised XML format; specifically, the NewsML-G2 standard. It also returns custom HTTP response codes to indicate request status.

All Payload Orchestrator requests are routed through the Authentication Engine in order to confirm the user's right to access the specified content, based on either cookies or credentials supplied in the request. Licence restrictions will also apply.

Figure 1: Payload Orchestrator overview


MMO USER/SYSTEM
submits payload request
AUTHENTICATION ENGINE
authenticates user access, licence, entitlements
success
failure
PAYLOAD ORCHESTRATOR
returns packages of article version data in NewsML-G2
PAYLOAD ORCHESTRATOR
returns custom response codes

Payload Orchestrator Requests

All Payload requests are structured as a 'GET' method HTTP request with a number of component parts.

Base URL

All Payload Orchestrator requests start with the following base URL:

`www.nla-eclipsweb.com/service/api/payload.xml`

This indicates that the request is for **`eclipsweb`** content, that it is a **`payload`** request, and that the data should be returned in **`XML`** format.

Method Parameters

Payload Orchestrator requests must use one of two methods.

- Index continuation
- Date-time

Of these, NLA recommend Index continuation as the preferred method. This is because it is optimal for frequent requests, ensures that article versions are not skipped, and delivers reproducible results with high performance.

In contrast, Date-time carries a risk of different article versions being returned when the same request is made at a later time due to the natural delay between the actual publication of an article version and it's processing in eCW. It is also a slower request and may deliver very high numbers of article versions in a single payload.

Note that a single payload request cannot use both methods, so any request using parameters for more than one method will be unsuccessful.

Index continuation

Each article version in eCW has a unique index value, which increments by one for each new article version processed by the NLA Not all article versions will be published to MMOs, but this value will always increase in sequence. This method uses this value to return article versions in order of their receipt and is independent of the accuracy of the article version metadata.

Index continuation Payload Orchestrator requests use the following method parameters:

Parameter	Data type	Function
`index=[######]`	Integer 7-9 digits	The index value for the last received article version (not the article ID or article version ID) Specifies that subsequent index values should be returned
`rows=[###]`	Integer ≤ 200	Specifies the number of article versions (rows) required in the output. Minimum value = 1, Maximum value = 200 Optional (20 rows returned if unspecified)

Date-time

Each article in eCW has a publication timestamp, which is provided by the publisher. This method returns article versions where the timestamp is within a range specified in the request, including where the article version was restricted during that time period.

Date-time Payload Orchestrator requests use the following method parameters:

Parameter	Data type	Function
`start=[DD/MM/YYYY HH:MM]`	Date OR Date-time	Specifies the earliest publication time from which article should be returned Must be within the last 28 days
`End=[DD/MM/YYYY HH:MM]`	Date OR Date-time	Specifies the latest publication time from which article should be returned Must be within 24 hours of `start`

Title Filter

Payload Orchestrator requests can specify from which title(s) results should be returned. The full list of titles and codes available at blog.nla.co.uk/ecwdocs/.

Payload Orchestrator title filtering details are supplied by the following parameters:

Parameter	Data type	Function
`title=[ABCD],[ABCE]`	Text Comma separated	Specifies the acronym(s) of the title(s) from which results should be returned Optional (all licensed titles returned if unspecified)

User Credentials

Payload Orchestrator requires user credentials. On the first request, these must be supplied in the query string. Subsequently, these can be provided by a cookie for up to 365 days.

Payload Orchestrator user credentials are supplied by the following parameters:

Parameter	Data type	Function
`user=[abcde@me.com]`	Text	Specifies the username for authentication Optional (cookie required if not specified)
`pwd=[xxxxxxxxx]`	Text	Specifies the password for the indicated user Optional (cookie required if not specified)

Complete Request

The above elements and parameters are combined to form a payload request, as shown in the examples below.

Index continuation (basic)

```
www.nla-eclipsweb.com/service/api/payload.xml?index=12345679
```

Index continuation (with rows, and user credentials)

```
www.nla-
eclipsweb.com/service/api/payload.xml?index=12345678&rows=200
&user=user@me.com&pwd=password
```

Date-time (basic)

```
www.nla-eclipsweb.com/service/api/payload.xml?start=11/02/2015
&end=11/02/2015
```

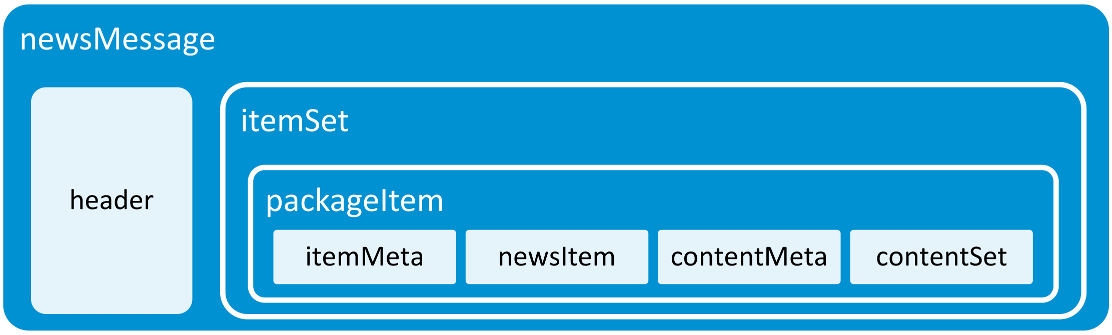
Date-time (with times, title filter, and user credentials)

```
www.nla-eclipsweb.com/service/api/payload.xml?start=11/02/2015
08:00&end=11/02/2015 09:00&title=WEBDM&user=username&pwd=password
```

Payload Orchestrator Output

All successful Payload Orchestrator requests return articles in XML format using the NewsML-G2 schema. The structure of a payload containing one article is shown below.

Figure 2: Payload Orchestrator output structure



More than one **packageItem** can appear within one **itemSet**, but only one **itemSet** can appear within one **newsMessage**.

Within the **contentSet** element, the article's text fields adhere to the NITF specification.

The details of the contents of a **packageitem** are available in Appendix A.

Redirector

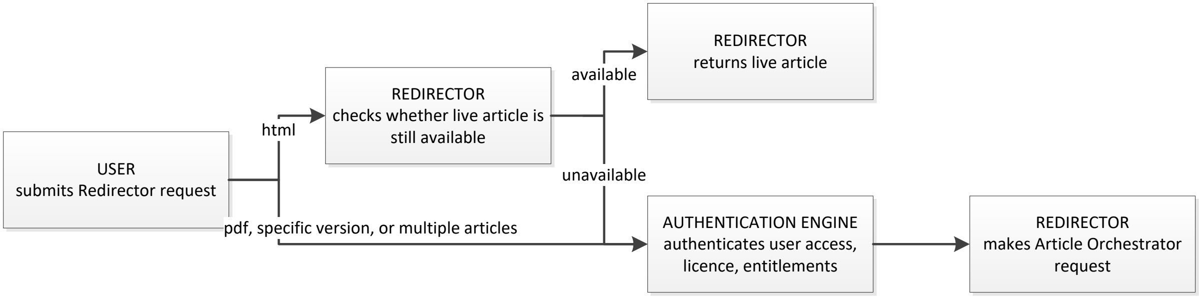
Redirector is a RESTful API, which can be queried manually or programmatically. It is accessible by all users.

Redirector checks the availability of a web news article, and reroutes if available, thereby allowing a user to access a live article in preference to an archived version.

Authentication is not required to access the live version of a web news article, although articles behind publisher paywalls may not be fully accessible without the appropriate subscriptions. However, if the live version is not available, the Authentication Engine requests user details before the archived version of the article can be returned.

Note that any request for a PDF of an article, or a specific version of an article, will always cause Redirector to make an Article Orchestrator request even if the article is still live.

Figure 3: Redirector overview


REDIRECTOR
returns live article
available
REDIRECTOR
checks whether live article is still available
html
USER
submits Redirector request
unavailable
pdf, specific version, or multiple articles
AUTHENTICATION ENGINE
authenticates user access, licence, entitlements
REDIRECTOR
makes Article Orchestrator request

Redirector Requests

All Redirector requests are structured as a 'GET' method HTTP request with a number of component parts.

Base URL

All Redirector requests start with the following base URL:

www.nla-eclipsweb.com/service/redirector/article/

This indicates that the request is for **eclipsweb** content, that it is a **redirector** request for **article** data.

Method Parameters

Redirector requests target one or more specific articles. The method for requesting multiple articles is different from that for requesting a single article.

Single article

Single article Redirector requests use the following method parameters:

Parameter	Data type	Function
`[########]`	Integer 8 digits	Specifies the Article ID of the article required in the output
`.[xxx]`	HTML OR PDF	Specifies the format in which the article must be returned If specified as PDF, leads to Article Orchestrator request
`version=[#]`	Integer 1 digit	Optional (latest version returned if unspecified) If specified, leads to Article Orchestrator request
`meta=[xxxx];[yyyy]`	Text Up to 5 items of free text	Specifies additional custom metadata that should be displayed If specified, defaults to Article Orchestrator request

Multiple articles

Multiple article Redirector requests use the following method parameters, where articleID:version groups are comma separated. The result will always be an Article Orchestrator call.

Parameter	Data type	Function
`.[xxx]`	PDF	Specifies the format in which the article must be returned
`[########]`	Integer 8 digits	Specifies the Article ID of the articles required in the output
`[#]`	Integer 1-2 digits	Specifies the version of the articles required in the output

Supplier details

Parameter	Data type	Function
`orgid=[###]`	Integer <6 digits	Specifies the MMO organisation whose branding should be applied to the orchestrated article, if available Optional (eCW branding applied if unspecified or no branding available for specified org) If specified, leads to Article Orchestrator request

User Credentials

Redirector does not require user credentials. However, if valid credentials are provided in the query string, and the request requires a subsequent background Article Orchestrator request, no further authentication will be required.

As with Payload Orchestrator, credentials provided in the query string will place a cookie, if possible, which will then authenticate the user for Article Orchestrator for up to 365 days.

Article Orchestrator user credentials are supplied by the following parameters:

Parameter	Data type	Function
`user=[abcde@me.com]`	Text	Specifies the username for authentication Optional (cookie or manual authentication required if not specified)
`pwd=[xxxxxxxxx]`	Text	Specifies the password for the indicated user Optional (cookie or manual authentication required if not specified)

Complete Request

The above elements and parameters are combined to form a redirector request, as shown in the examples below.

Single article (basic HTML)

```
www.nla-eclipsweb.com/service/redirector/article/12345678.html
```

Single article (PDF with version, custom metadata, and branding)

```
www.nla-eclipsweb.com/service/redirector/article/12345678.pdf
?version=1&meta=Exclusive;Positive sentiment;Recommended for
followup&orgid=34
```

Multiple articles (with branding and credentials)

```
www.nla-
eclipsweb.com/service/redirector/article/articles.pdf?articles=12345
678:1,12345679:3,12345689:2&orgid=66&user=username&pwd=password
```

Redirector Output

All successful Redirector requests return either:

- The live article on the source webpage
- The article in Article Orchestrator format

The format of the live article on the source webpage is not controlled by NLA or eClips Web. The format of Article Orchestrator is detailed below.

Article Orchestrator

Article Orchestrator is a component which renders one or more eCW article into a human-readable format.

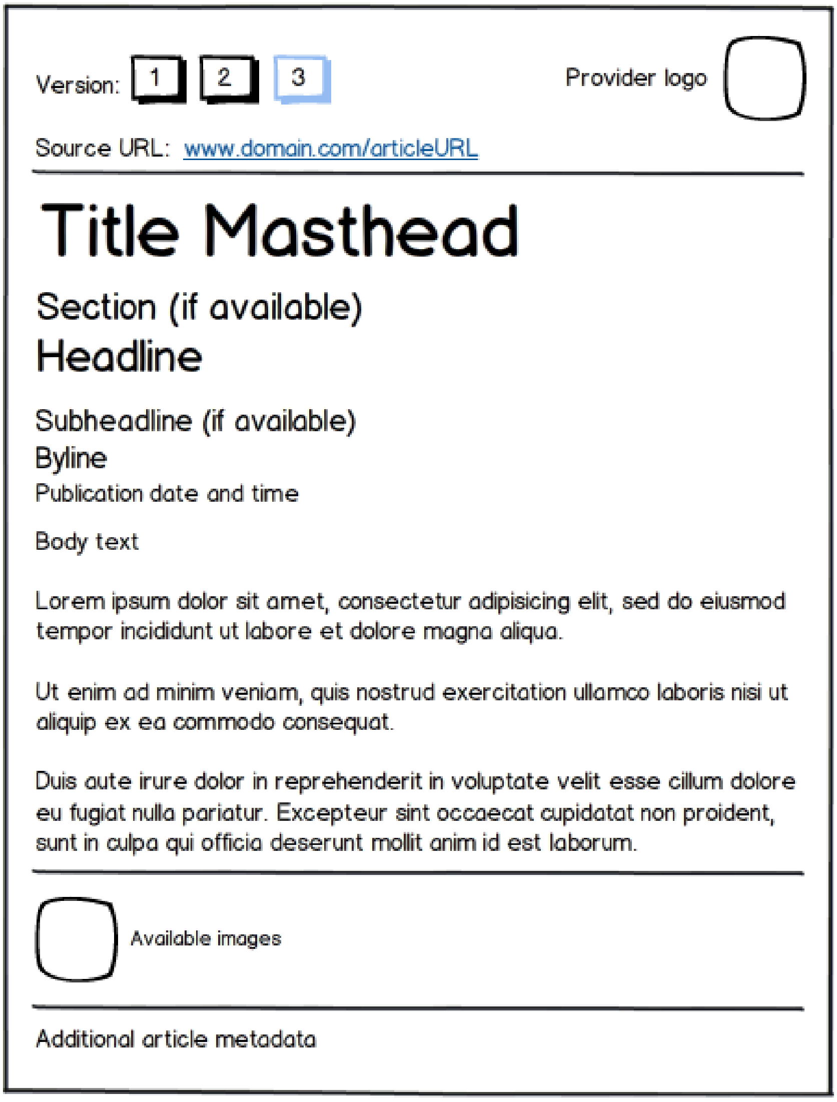
As described above, Redirector requests for which the live article is unavailable, or where certain parameters are present in the request, and where authentication is successful, will result in a background request to Article Orchestrator.

Article Orchestrator Output

An article rendered by Article Orchestrator adheres to a standard structure. This structure is the same whether the article is delivered in HTML or PDF, although the exact format may vary depending on the branding applied and the user's settings.

The below figure outlines the structure of an Article Orchestrator document. The structure shown is shared by HTML and PDF documents.

Figure 4: Article Orchestrator output structure



Title Masthead

Section (if available)

Headline

Subheadline (if available)

Byline

Publication date and time

Body text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Additional article metadata

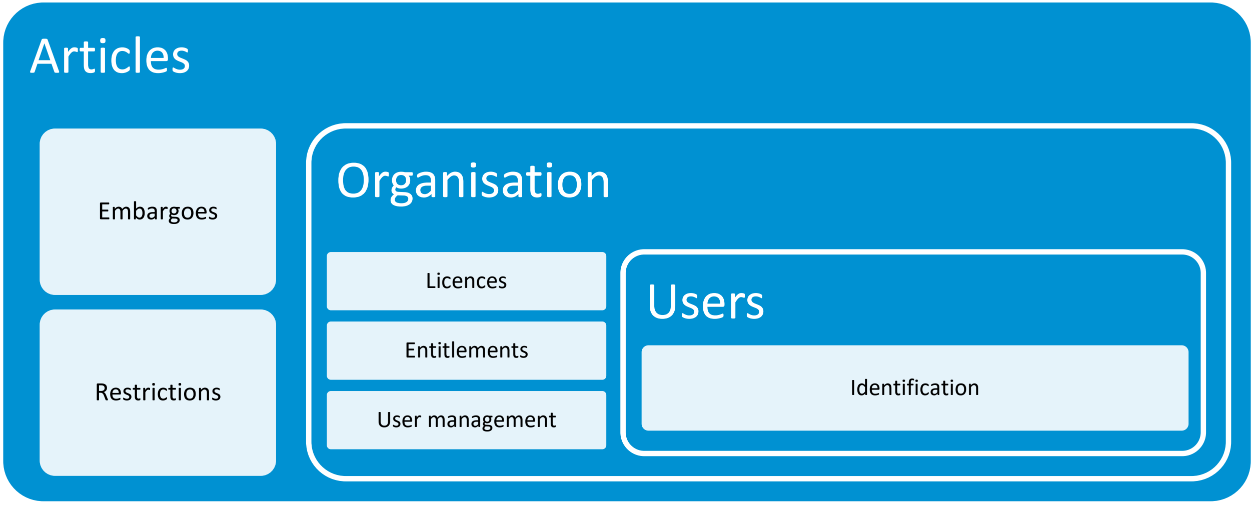
Authentication Engine

Authentication Engine is the mechanism by which users attempting to access any part of eClips Web are assessed and then allowed or denied access to content and features.

This mechanism is made up of a connected set of permissions functions:

- User management
- User identification
- Licences
- Entitlements
- Embargoes
- Restrictions

Figure 5: Authentication Engine overview



User management

Users of eClips Web are managed through the eClips User Management Interface (UMI). This is available at `https://www.nla-eclips.com/manage/`.

Once an organisation has been set up in eClips Web by the NLA, the organisation will have the appropriate licences assigned to it, as well as at least one user.

If a user has Admin permissions, they will be able to create and manage other users for their organisation through the UMI.

An organisation can be designated an MMO by NLA, also allowing them to be indicated as providing MMO services to another organisation. When this link is in place, an admin user for an MMO organisation can also create and manage users for the linked organisations through the UMI.

User identification

On each request for eClips Web content, the requesting user must be identified by providing a username and password. The mechanisms for providing these credentials are as follows.

Mechanism	Interaction	Components	Availability
MMO user authentication	URL query string	All	MMO organisations only Intended for machine authentication
Client user authentication	Dialogue box in browser	Redirector only	All users

For each of these mechanisms, the first successful authentication will generate a user- and device-specific cookie. This avoids the need for further identification for 365 days, or until the cookie is removed. For this to work, cookies must be allowed on the device.

Licences

Organisations are set up with licences which define to which components and titles within eClips Web they have access. For example, MMO organisations have access to the Payload Orchestrator component but client organisations do not.

Licenses are set up by the NLA based on the agreements made with individual organisations.

Entitlements

An organisation's licence for a given title is accompanied by an entitlement. This is the period of time after the publication of an article during which users in that organisation will have access to that article and is component specific.

Most licences are set up with 7-day entitlements for Payload Orchestrator and 100-day entitlements for Article Orchestrator.

Embargoes

For some articles, the publisher of that article will apply an embargo to that article's availability in eClips Web.

In this case, the article will not be available in eClips Web feeds until that embargo has passed.

Restrictions

For some articles, the publisher of that article will apply a restriction to that article's availability in eClips Web. A restriction indicates the level of permission a user must have to continue to have access to the article.

In this case, the article will no longer be available in eClips Web feeds once the restriction has been applied if the user has a permission level lower than that required to access the restricted article.

More details are available in Appendix B.

Appendix A – Important Properties

For convenience we have listed the xPath routes to the most important properties of an eClips Web article below.

NLA article properties

Property	Description	xPath
NLA article ID	Unique identifier for the article in eClips Web, used to group versions of an article	`/newsMessage/itemSet/packageItem /itemMeta/nla:articleIdentifier/@id`
NLA index value	Unique identifier for the article version in eClips Web, used for index continuation	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:index`
NLA title acronym	Unique identifier for the title in which the article was published, used for title filtering	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:titleAcronym`
NLA redirector URI	URI which should be followed to find the article version through eClips Web	`/newsMessage/itemSet/packageItem /itemMeta/link[@rel=”irel:associatedWith”] /@href`
Original publication date	Date and time at which the first version of the article was published	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/firstCreated`

Title properties

Property	Description	xPath
Title domain	Domain of the title in which the article was published	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:provider/@literal`
Publisher name	Name of the publisher of the title	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:publisher/@literal`

ABCe data

Property	Description	xPath
Start date	Date on which ABCe’s measurements started for a given set of data	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:ABCe/From`
End date	Date on which ABCe’s measurements ended for a given set of data	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:ABCe/To`
Unique browsers	The number of unique browsers on the title domain in the given month as assessed by ABCe	`/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:ABCe/Primary`

Page impressions	The number of page impressions on the title domain in the given month as assessed by ABCe	/newsMessage/itemSet/packageItem /newsItem/itemMeta/nla:ABCe/Secondary

Article version properties

Property	Description	xPath
Article version URI	Full URI at which the article version was published	/newsMessage/itemSet/packageItem /itemMeta/link [@rel=”irel:processedFrom”]/@href
Article version	Version number of the article version	/newsMessage/itemSet/packageItem /itemMeta/nla:articleIdentifier /@version
Status	Indicator of whether the article version is usable or withdrawn	/newsMessage/itemSet/packageItem /newsItem/itemMeta/pubStatus
Publication date/time	Date and time at which the current article version was published	/newsMessage/itemSet/packageItem /newsItem/itemMeta/versionCreated
Loaded date/time	Date and time at which the current article version was loaded into the database	/newsMessage/itemSet/packageItem /newsItem/itemMeta/versionLoaded
Section	Section of the title website in which the article version was published	/newsMessage/itemSet/packageItem /newsItem/contentMeta/nla:section
Word count	Total number of words in the article version’s Headline, Body, and Caption fields	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /@wordcount
Character count	Total number of characters in the Headline, Body, and Caption fields	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /@nla:charactercount

Article version content

Property	Description	xPath
Headline	Headline text of the article version	/newsMessage/itemSet/packageItem /newsItem/contentMeta/headline
Slugline	Slugline, or subheadline, text of the article version	/newsMessage/itemSet/packageItem /newsItem/contentMeta/slugline
Byline	Byline, or authorship, details of the article version	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /nitf/body/body.head/byline/byttl
Body	Body text of the article version	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /nitf/body/body.content

Image URI	URI reference to image(s) in the article version	/newsMessage/itemSet/packageItem /newsItem/contentSet/remoteContent/@href
Image caption	Caption text of image(s) in the article version	/newsMessage/itemSet/packageItem /newsItem/contentMeta/description [@role=”drol:caption”]
Image credit	Attribution text of image(s) in the article version	/newsMessage/itemSet/packageItem /newsItem/contentMeta/creditline

Article version additional properties

These properties are only available for selected articles published by theguardian.com, and to those organisations and users who are enabled for these additional properties.

Property	Description	xPath
Page number	Page in the printed paper on which the equivalent articles was printed	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /nitf/head/pubdata/@position.sequence
Production office	Office in which the article was produced	/newsMessage/itemSet/packageItem /newsItem/contentSet/inlineXML /nitf/head/dateline/location

Appendix B – Article Version Status

As described above, the Article Version Status indicates whether an article version is usable or withdrawn (restricted).

By default, article versions in eClips Web are usable (`pubStatus=usable`), which means that it can be processed, viewed, and stored according to usage agreements.

However, on occasion, the publisher of an article will choose to restrict, or withdraw, access to one or more versions of a published article.

When one or more versions of an article is restricted (`pubStatus=withdrawn`), this usually happens after a given article version has already been received and processed. At this point, the metadata of article version with the new `pubStatus` will be updated with a new `index` value, ensuring that it will be returned in the next Payload Orchestrator (index continuation) request with the withdrawn status indicated.

Note that the date-time at which a new status was applied will also be considered in any Payload Orchestrator (date-time) requests covering that date-time. At the same time, NLA will issue a restriction notice by email to all MMOs who could have received the affected article versions.

When an article version’s status is withdrawn, all organisations and users receiving this are obligated to remove all instances of this article version from all stored and shared locations. Where multiple versions of the same article are withdrawn, the obligation applies to all instances of all affected article versions.

These article versions will now no longer be available through Redirector and Article Orchestrator. If later versions of the restricted article are unrestricted, these will still be available for use.

Appendix C – Errors

The following HTTPS response codes may be returned from a Payload Orchestrator request:

Code	Meaning	Context
200	Successful	Payload Orchestrator request
204	The licensed entitlement period has been exceeded or there are no articles available in the period you have specified. Please ensure that the date of your request falls within your licensed entitlement period	Date-time Payload Orchestrator request
400	Please ensure the date span does not exceed 24 hours OR Please include both start and end date/times on the querystring OR Start and/or end date cannot be in the future	Date-time Payload Orchestrator request
401	Unauthorised OR Please make sure you enter your username (user) and password (pwd) on the QueryString.	Payload Orchestrator request
404	Not found	Payload Orchestrator request

Appendix D – Glossary

Term	Meaning
ABCe	Audit Bureau of Circulations (ABC) is the industry body for media measurement. They supply domain-level access statistics. ABCe indicates the branch of the ABC that deals with electronic publications (although this terminology is no longer used by the ABC, it is helpful in distinguishing the information source within eClips). More information can be found at www.abc.org.uk/.
NewsML-G2	An XML standard for new content metadata. More information can be found at iptc.org/standards/newsml-g2/.
NITF	News Industry Text Format: an XML standard for news content structure. More information can be found at iptc.org/standards/nitf/.
RESTful	An architectural style for an API which uses Representational State Transfer. More information can be found at ibm.com/developerworks/library/ws-restful/.

Document Control

Version	Date	Updated by	Updates
1.4	12-02-2015	Tessa Radwan	Document adapted from existing spec v1.4, to update contents and branding.
1.5	12-03-2015	Tessa Radwan	Updated following feedback from MA
1.6	02-04-2015	Tessa Radwan	Updated with Redirector, Article Orchestrator, and Authentication sections
1.7	22-05-2015	Tessa Radwan	Updated with feedback from ISE
2.0	03-06-2015	Tessa Radwan	Prepared for publication

2.1	23-10-2015	Tessa Radwan	Updated with additional article version properties
3.0	23-10-2015	Tessa Radwan	Prepared for publication
3.1	12/01/2016	Tessa Radwan	Updated with details of new Payload date fields
4.0	12/01/2016	Tessa Radwan	Prepared for publication
4.1	21/06/2018	Stephen Handley	Updated with text fixes
4.2	08/07/2018	Mario-Robert Daigle	Updated with HTTPS details