

eCFR API Documentation

The base URL for all API endpoints is <https://www.ecfr.gov>.

Admin Service Access eCFR metadata through JSON

GET **/api/admin/v1/agencies.json** Agencies

All top-level agencies in name order with children also in name order

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code **Description**

200

```
{
  "agencies": [
    {
      "name": "Administrative Conference of the United States",
      "short_name": "ACUS",
      "display_name": "Administrative Conference of the United States",
      "sortable_name": "Administrative Conference of the United States",
      "slug": "administrative-conference-of-the-united-states",
      "children": [],
      "cfr_references": [
        {
          "title": 1,
          "chapter": "III"
        }
      ]
    },
    ...
    {
      "name": "Department of Agriculture",
      "short_name": "USDA",
      "display_name": "Department of Agriculture",
      "sortable_name": "Agriculture, Department of",
      "slug": "agriculture-department",
      "children": [
        {
          "name": "Agricultural Marketing Service",
          "short_name": "AMS",
          "display_name": "Agricultural Marketing Service, Department of Agriculture",
          "sortable_name": "Agricultural Marketing Service",
          "slug": "agricultural-marketing-service",
          "cfr_references": [
            {
              "title": 7,
              "chapter": "I"
            },
            ...
          ],
          "cfr_references": [
            {
              "title": 2,
              "chapter": "IV",
            },
            ...
          ]
        },
        ...
      ]
    },
    ...
    {
      "name": "World Agricultural Outlook Board",
      "short_name": "WAOB",
      "display_name": "World Agricultural Outlook Board",
      "sortable_name": "World Agricultural Outlook Board",
      "slug": "world-agricultural-outlook-board",
      "cfr_references": [
        {
          "title": 7,
          "chapter": "XXXVIII"
        }
      ]
    }
  ]
}
```

The Corrections service can be used to determine all corrections or can be filtered by title, effective date, or correction date.

Parameters

Try it out

Name	Description
date string (query)	Restricts results to eCFR corrections that occurred on or before the specified date and that were corrected on or after the specified date. Format: YYYY-MM-DD
title string (query)	Restricts results to the given title number: Format: '1', '2', '50', etc
error_corrected_date string (query)	Returns only corrections that were corrected on the given date. Format: YYYY-MM-DD

Responses

Response content type

application/json

Code	Description
200	<pre>"ecfr_corrections": [{ "id": 1, "cfr_references": [{ "cfr_reference": "7 CFR 2.80", "hierarchy": { "title": "7", "subtitle": "A", "part": "2", "subpart": "N", "section": "2.80" } }], "corrective_action": "Amended", "error_corrected": "2005-12-02", "error_occurred": "2005-09-23", "fr_citation": "70 FR 55706", "position": 1, "display_in_toc": false, "title": 7, "year": 2005, "last_modified": "2021-12-22" }, ... { "id": 3858, "cfr_references": [{ "cfr_reference": "50 CFR 660.71", "hierarchy": { "title": "50", "part": "660", "section": "660.71" } }], "corrective_action": " (e)(398) and (399) correctly designated as (e)(298) and (299)\r\n\r\n", "error_corrected": "2023-01-09", "error_occurred": "2023-01-01", "fr_citation": "87 FR 77015", "position": 3, "display_in_toc": false, "title": 50, "year": 2023, "last_modified": "2023-01-31" }]</pre>
400	<pre>null // Bad request</pre>
404	<pre>Title not found</pre>

GET

/api/admin/v1/corrections/title/{title}.json

Corrections title route returns all corrections for the supplied title.

The Corrections service can be used to determine all corrections for the given title.

Parameters

Try it out

Name	Description
title * required string (path)	Restricts results to the given title number: Format: '1', '2', '50', etc

Responses

Response content type

application/json

Code	Description
200	<pre>{ "ecfr_corrections": [{ "id": 1473, "cfr_references": [{ "cfr_reference": "6 CFR Part 5", "hierarchy": { "title": "6", "chapter": "I", "part": "5" } }], "corrective_action": "Paragraph 72 added", "error_corrected": "2014-11-12", "error_occurred": "2014-04-02", "fr_citation": "79 FR 18441", "position": 1, "display_in_toc": false, "title": 6, "year": 2014, "last_modified": "2021-12-22" }, { "id": 2252, "cfr_references": [{ "cfr_reference": "6 CFR 1001.6", "hierarchy": { "title": "6", "chapter": "X", "part": "1001", "section": "1001.6" } }], "corrective_action": "(b)(4) amended", "error_corrected": "2018-01-16", "error_occurred": "2017-08-28", "fr_citation": "82 FR 34836", "position": 1, "display_in_toc": false, "title": 6, "year": 2018, "last_modified": "2021-12-22" }]}</pre>
404	

Search Service

Historical search of the eCFR

GET

/api/search/v1/results

Search results

Parameters

Try it out

Name	Description
query string (query)	Search term; searches the headings and the full text

Name	Description
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)
last_modified_on_or_before string (query)	limit to content last modified on or before this date (YYYY-MM-DD)
per_page integer (query)	number of results per page; max of 1000
page integer (query)	page of results; can't paginate beyond 10000 total results
order string (query)	order of results
paginate_by string (query)	how results should be paginated - 'date' will group results so that all results from a date appear on the same page of pagination. If 'date' is chosen then one of the last_modified_* options is required.

Responses

Response content type

application/json

Code	Description
200	<p><i>Success</i></p> <p>Example Value Model</p> <pre>{ "results": [null], "meta": { "description": "string", "current_page": 0, "total_count": 0, "total_pages": 0, "max_score": 0, "min_date": "string", "max_date": "string" } }</pre>
405	<p><i>Invalid input</i></p>

GET /api/search/v1/count Search result count

Parameters

Try it out

Name	Description
query string (query)	Search term; searches the headings and the full text
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)
last_modified_on_or_before string (query)	limit to content last modified on or before this date (YYYY-MM-DD)

Responses

Response content type

application/json

Code	Description
200	<i>Success</i>
405	<i>Invalid input</i>

GET /api/search/v1/summary Search summary details

Parameters

Try it out

Name	Description
query string (query)	Search term; searches the headings and the full text
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)

Name	Description
last_modified_on_or_before	limit to content last modified on or before this date (YYYY-MM-DD)
string	
(query)	
Responses	Response content type application/json
Code	Description
200	Success
405	Invalid input

GET /api/search/v1/counts/daily Search result counts by date

Parameters	Try it out
Name	Description
query	Search term; searches the headings and the full text
string	
(query)	
agency_slugs[]	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
array [string]	
(query)	
date	limit to content present on this date (YYYY-MM-DD)
string	
(query)	
last_modified_after	limit to content last modified after this date (YYYY-MM-DD)
string	
(query)	
last_modified_on_or_after	limit to content last modified on or after this date (YYYY-MM-DD)
string	
(query)	
last_modified_before	limit to content last modified before this date (YYYY-MM-DD)
string	
(query)	
last_modified_on_or_before	limit to content last modified on or before this date (YYYY-MM-DD)
string	
(query)	
Responses	Response content type application/json
Code	Description
200	Success
405	Invalid input

GET /api/search/v1/counts/titles Search result counts by title

Parameters	Try it out
------------	------------

Name	Description
query string (query)	Search term; searches the headings and the full text
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)
last_modified_on_or_before string (query)	limit to content last modified on or before this date (YYYY-MM-DD)

Responses

Response content type

application/json

Code	Description
200	<i>Success</i>
405	<i>Invalid input</i>

GET

/api/search/v1/counts/hierarchy

Search result counts by hierarchy

Parameters

Try it out

Name	Description
query string (query)	Search term; searches the headings and the full text
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)

Name	Description
last_modified_on_or_before	limit to content last modified on or before this date (YYYY-MM-DD)
string	
(query)	
Responses	Response content typeapplication/json
Code	Description
200	Success
405	Invalid input

GET

/api/search/v1/suggestions

Search suggestions

Parameters

Try it out

Name	Description
query string (query)	Search term; searches the headings and the full text
agency_slugs[] array [string] (query)	limit to content currently associated with these agencies (see AdminService agencies endpoint to retrieve a list of agency slugs)
date string (query)	limit to content present on this date (YYYY-MM-DD)
last_modified_after string (query)	limit to content last modified after this date (YYYY-MM-DD)
last_modified_on_or_after string (query)	limit to content last modified on or after this date (YYYY-MM-DD)
last_modified_before string (query)	limit to content last modified before this date (YYYY-MM-DD)
last_modified_on_or_before string (query)	limit to content last modified on or before this date (YYYY-MM-DD)

Responses

Response content type

application/json

Code	Description
200	Success
405	Invalid input

Versioner Service

Access CFR content and structure files through JSON and XML

GET	/api/versioner/v1/ancestry/{date}/title-{title}.json	Ancestors route returns all ancestors (including self) from a given level through the top title node.
The Ancestry service can be used to determine the complete ancestry to a leaf node at a specific point in time.		

Example

The complete hierarchy for **2 CFR 1532.137** is

Title 2

Subtitle B

Chapter XV

Part 1532

Subpart A

Section 1532.137

To retrieve this complete hierarchy you can use the ancestry endpoint and provide a Title, Part and Section (you can provide additional layers of the hierarchy) to retrieve a full ancestry.
See sample json responses below.

Parameters

Try it out

Name	Description
date * required string (path)	YYYY-MM-DD
title * required string (path)	Title Number: '1', '2', '50', etc
subtitle string (query)	Uppercase letter. 'A', 'B', 'C'
chapter string (query)	Roman Numerals and digits 0-9. 'I', 'X', '1'
subchapter string (query)	A SUBCHAPTER REQUIRES A CHAPTER. Uppercase letters with optional underscore or dash. 'A', 'B', 'I'
part string (query)	Uppercase letters with optional underscore or dash. 'A', 'B', 'I'
subpart string (query)	A SUBPART REQUIRES A PART. Generally an uppercase letter. 'A', 'B', 'C'
section string (query)	A SECTION REQUIRES A PART. Generally a number followed by a dot and another number. '121.1', '13.4', '1.9'
appendix string (query)	AN APPENDIX REQUIRES A SUBTITLE, CHAPTER or PART. Multiple formats. 'A', 'III', 'App. A'

Responses

Response content typeapplication/json

CodeDescription

200

```
[
  {
    "type": "title",
    "label": "Title 2 - Grants and Agreements",
    "label_level": "Title 2",
    "label_description": "Grants and Agreements",
    "identifier": "2"
  },
  {
    "type": "subtitle",
    "label": "Subtitle B - Federal Agency Regulations for Grants and Agreements",
    "label_level": "Subtitle B",
    "label_description": "Federal Agency Regulations for Grants and Agreements",
    "identifier": "B",
    "reserved": false,
    "section_range": "$$ 300.1 - 5900.101"
  },
  {
    "type": "chapter",
    "label": "Chapter XV - Environmental Protection Agency",
    "label_level": "Chapter XV",
    "label_description": "Environmental Protection Agency",
    "identifier": "XV",
    "reserved": false,
    "section_range": "$$ 1500.1 - 1536.505"
  },
  {
    "type": "part",
    "label": "Part 1532 - Nonprocurement Debarment and Suspension",
    "label_level": "Part 1532",
    "label_description": "Nonprocurement Debarment and Suspension",
    "identifier": "1532",
    "reserved": false,
    "section_range": "$$ 1532.10 - 1532.1600"
  },
  {
    "type": "subpart",
    "label": "Subpart A - General",
    "label_level": "Subpart A",
    "label_description": "General",
    "identifier": "A",
    "reserved": false,
    "section_range": "$ 1532.137"
  },
  {
    "type": "section",
    "label": "$ 1532.137 Who in the EPA may grant an exception to Let an excluded person participate in a covered transaction?",
    "label_level": "$ 1532.137",
    "label_description": "Who in the EPA may grant an exception to Let an excluded person participate in a covered transaction?",
    "identifier": "1532.137",
    "reserved": false
  }
]
```

404

null // Invalid input or missing some hierarchy

GET /api/versioner/v1/full/{date}/title-{title}.xml

Source XML for a title or subset of a title. Requests can be for entire titles or part level and below. Downloadable XML document is returned for title requests. Processed XML is returned if part, subpart, section, or appendix is requested.

The title source route can be used to retrieve the source xml for a complete title or subset. The subset of xml is determined by the lowest leaf node given. For example, if you request Title 1, Chapter I, Part 1, you'll receive the XML only for Part 1 and its children. If you request a section you'll receive the section XML inside its parent Part as well as relevant non-section sibling nodes (Auth, Source, etc). The largest title source xml files can be dozens of megabytes. GPO eCFR XML User guide

Parameters

Try it out

NameDescription

date * required YYYY-MM-DD
string
(path)

title * required Title Number: '1', '2', '50', etc
string
(path)

Name	Description
subtitle string (query)	Uppercase letter. 'A', 'B', 'C'
chapter string (query)	Roman Numerals and digits 0-9. 'I', 'X', '1'
subchapter string (query)	A SUBCHAPTER REQUIRES A CHAPTER. Uppercase letters with optional underscore or dash. 'A', 'B', 'I'
part string (query)	Uppercase letters with optional underscore or dash. 'A', 'B', 'I'
subpart string (query)	A SUBPART REQUIRES A PART. Generally an uppercase letter. 'A', 'B', 'C'
section string (query)	A SECTION REQUIRES A PART. Generally a number followed by a dot and another number. '121.1', '13.4', '1.9'
appendix string (query)	AN APPENDIX REQUIRES A SUBTITLE, CHAPTER or PART. Multiple formats. 'A', 'III', 'App. A'

Responses

Response content type

application/xml

Code	Description			
200	<p><i>A downloadable Title XML document for entire Title requests or processed XML for part level and below requests.</i></p> <p>Headers:</p> <table><tr><th>Name</th><th>Description</th><th>Type</th></tr></table>	Name	Description	Type
Name	Description	Type		
404	<p><i>Invalid input.</i></p>			

GET **/api/versioner/v1/structure/{date}/title-{title}.json** Structure JSON for a title

The structure JSON endpoint returns the complete structure of a title back as json. This format does not include the content of the title but does include all structure and content nodes as well as their meta data including their type, label, description, identifier and children.

Parameters

Try it out

Name	Description
date * required string (path)	YYYY-MM-DD
title * required string (path)	Title Number: '1', '2', '50', etc

Responses

Response content type

application/json

Code

Description

200

```
{
  "type": "title",
  "label": "Title 3 - The President",
  "label_level": "Title 3",
  "label_description": "The President",
  "identifier": "3",
  "children": [
    {
      "type": "chapter",
      "label": "Chapter I - Executive Office of the President",
      "label_level": "Chapter I",
      "label_description": "Executive Office of the President",
      "identifier": "I",
      "reserved": false,
      "children": [
        {
          "type": "part",
          "label": "Part 100 - Standards of Conduct",
          "label_level": "Part 100",
          "label_description": "Standards of Conduct",
          "identifier": "100",
          "reserved": false,
          "children": [
            {
              "type": "section",
              "label": "$ 100.1 Ethical conduct standards and financial disclosure regulations.",
              "label_level": "$ 100.1",
              "label_description": "Ethical conduct standards and financial disclosure regulations.",
              "identifier": "100.1",
              "reserved": false
            }
          ]
        },
        {
          "type": "section_range",
          "label": "$ 100.1"
        }
      ]
    }
  ]
}
```

404

Invalid input.

GET /api/versioner/v1/titles.json Summary information about each title

The Title service can be used to determine the status of each individual title and of the overall status of title imports and reprocessings. It returns an array of all titles containing a hash for each with the name of the title, the latest amended date, latest issue date, up-to-date date, reserved status, and if applicable, processing in progress status. The meta data returned indicates the latest issue date and whether titles are currently being reprocessed.

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code

Description

200

```
{
  "titles": [
    {
      "number": 1,
      "name": "General Provisions",
      "latest_amended_on": "2022-05-04",
      "latest_issue_date": "2022-05-04",
      "up_to_date_as_of": "2022-07-28",
      "reserved": false
    },
    {
      "number": 2,
      "name": "Grants and Agreements",
      "latest_amended_on": "2022-05-19",
      "latest_issue_date": "2022-06-03",
      "up_to_date_as_of": "2022-07-28",
      "reserved": false,
      "processing_in_progress": true
    },
    {
      "number": 3,
      "name": "The President",
      "latest_amended_on": "2015-03-17",
      "latest_issue_date": "2019-06-27",
      "up_to_date_as_of": "2022-07-28",
      "reserved": false
    },
    ...
  ],
  "meta": {
    "date": "2022-07-29",
    "import_in_progress": true
  }
}
```

GET

/api/versioner/v1/versions.json

Deprecated, use /v1/titles.json. Returns the array of all titles as hashes with their title number and date last updated.

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code

Description

200

```
{
  "title_versions": [
    {
      "title": 1,
      "last_updated": "2018-05-02"
    },
    {
      "title": 2,
      "last_updated": "2018-07-03"
    },
    {
      "title": 3,
      "last_updated": "2015-03-17"
    },
    ...
  ],
  "meta": {
    "last_update": "2018-10-29"
  }
}
```

GET

/api/versioner/v1/versions/title-{title}.json

Returns an array of all sections and appendices inside a title.

Each content object includes its identifier, parent hierarchy and when it was last updated.

Parameters

Try it out

Name	Description
title * required string (path)	Title Number: '1', '2', '50', etc

Responses	Response content type
	<div>application/json</div>

Code	Description
200	<div><pre>{ "content_versions": [{ "date": "2016-11-28", "identifier": "1.100", "name": "\$ 1.100 Content of this title.", "part": "1", "subpart": "A", "title": "2", "type": "section" }, { "date": "2018-07-03", "identifier": "700.16", "name": "\$ 700.16 Marking.", "part": "700", "subpart": "D", "title": "2", "type": "section" }, ...], "meta": { "title": "2", "result_count": "1273" } }</pre></div>