



Australian Government
Department of Climate Change, Energy,
the Environment and Water

FullCAM PR External API

Databuilder API

FullCAM Team

v0.1



© Commonwealth of Australia 2024

Ownership of intellectual property rights

Unless otherwise noted, copyright (and any other intellectual property rights) in this publication is owned by the Commonwealth of Australia (referred to as the Commonwealth).

Creative Commons licence

All material in this publication is licensed under a [Creative Commons Attribution 4.0 International Licence](#) except content supplied by third parties, logos and the Commonwealth Coat of Arms.

Inquiries about the licence and any use of this document should be emailed to copyright@dcceew.gov.au.



Cataloguing data

This publication (and any material sourced from it) should be attributed as: DCCEEW 2022, *Our food future: trends and opportunities*, Department of Climate Change, Energy, the Environment and Water, Canberra.

CC BY 4.0.

This publication is available at [Publications - DCCEEW](#).

Department of Climate Change, Energy, the Environment and Water
GPO Box 3090 Canberra ACT 2601
Telephone 1800 920 528
Web dcceew.gov.au

Disclaimer

The Australian Government acting through the Department of Climate Change, Energy, the Environment and Water has exercised due care and skill in preparing and compiling the information and data in this publication. Notwithstanding, the Department of Climate Change, Energy, the Environment and Water, its employees and advisers disclaim all liability, including liability for negligence and for any loss, damage, injury, expense or cost incurred by any person as a result of accessing, using or relying on any of the information or data in this publication to the maximum extent permitted by law.

Acknowledgements

The authors thank interview and survey participants for their input.

Acknowledgement of Country

We acknowledge the Traditional Owners of Country throughout Australia and recognise their continuing connection to land, waters, and culture. We pay our respects to their Elders past and present.

Table of Contents

FulICAM PR External API.....	1
Databuilder API	1
Table of Contents	3
FulICAM 2020 Data Builder API.....	5
Overview.....	5
Key Features	5
API Architecture	5
Request for Access to API's:.....	5
Access Request Template:	5
API Authentication.....	6
Base URL	6
API Endpoints.....	7
1. siteinfo API.....	7
2. species API	9
3. regimes API.....	10
4. templates API.....	11
5. template API	12
6. update-spatialdata API	13
Error Handling.....	14
Development Best Practices	16
1. Model Objects.....	16
2. Asynchronous Calls	16
3. Error Handling Mechanisms.....	16
4. Caching Strategy	16
5. Version Compatibility	16
Performance Optimization	16
Support and Contact	17
Changelog (2020)	18
FulICAM 2024 Data Builder API.....	19
Overview.....	19
Key Features.....	19
API Architecture	19
API Access Request.....	19
Access Request Process	19

Response	20
API Authentication.....	20
Base URL	20
API Endpoints.....	20
1. siteinfo API.....	20
2. species API	22
3. regimes API.....	23
4. templates API.....	24
5. template API	25
6. convert-plotfile API.....	26
7. update-spatialdata.....	27
Error Handling	28
Error Response Format.....	28
Common Error Codes	29
Parameter Validation Errors	29
Development Best Practices	30
1. Model Objects.....	30
2. Asynchronous Calls	30
3. Error Handling Mechanisms.....	30
4. Caching Strategy	30
5. Rate Limiting Considerations	30
API Usage Workflow.....	30
Performance Optimization	31
Support and Contact	32
Changelog (2024)	32
Version Differences: 2020 vs 2024.....	32
Migration Considerations.....	33
Glossary.....	34

FullCAM 2020 Data Builder API

Overview

The **Full Carbon Accounting Model (FullCAM) 2020 Data Builder API** provides programmatic access to Australia's carbon accounting data for forestry and land use sectors. This RESTful API enables developers to retrieve site-specific information, species data, management regimes, and templates for carbon accounting calculations using the 2020 parameter set and calibrations.

Key Features

- **Geographic Site Information:** Soil, climate, and regional data for any Australian location
- **Species Database:** Comprehensive forest and agriculture species data with FullCAM 2020
- **Management Regimes:** Detailed forest and agriculture management protocols and practices
- **Template System:** Pre-configured carbon accounting templates for 2020 calculations
- **Cross-Platform Compatibility:** Removes dependency on Windows-only desktop application
- **Legacy Support:** Maintains compatibility with existing 2020 FullCAM workflows

API Architecture

FullCAM implements a REST (Representational State Transfer) architecture that is stateless, meaning servers do not save client data between requests. The API provides two main service categories:

1. **Data API:** Retrieves spatial data, species information, and templates using 2020 parameters
2. **Plot Simulation API:** Enables running simulations using existing 2020 plot files

Request for Access to APIs:

To gain access to the FullCAM API, you must request a subscription key from the FullCAM business team.

Access Request Template:

Send an email to **fullcam@dcceew.gov.au** with the following information:

1. **Organization Name**
2. **Email Address**
3. **Phone Number**
4. **Business Use Case:** Detailed description of intended usage

5. Usage Details:

- Number of requests per system intended
- Number of users
- Peak time of usage
- Any other relevant usage patterns

6. Number of Users

7. Access Duration: How long you need access

Response

Once approved, FullCAM will issue a **unique API subscription key** that must be included in all API requests.

API Authentication

All API requests require a subscription key for authentication.

Required Header

Ocp-Apim-Subscription-Key: {{YOUR_SUBSCRIPTION_KEY}}

Note: FullCAM 2020 and 2024 APIs require **separate subscription keys**:

- **FullCAM 2020:** Use subscription key for 2020 version access
- **FullCAM 2024:** Use subscription key for 2024 version access

Error Response (401)

```
{  
    "statusCode": 401,  
  
    "message": "Access denied due to missing subscription key. Make sure to include subscription  
key when making requests to an API."  
}
```

Base URL

<https://api.dcceew.gov.au/climate/carbon-accounting/2020/data/v1>

API Endpoints

1. siteinfo API

Retrieve comprehensive site information for a specific geographic location including soil, climate, and regional data using 2020 parameters.

API Endpoint: GET /2020/data-builder/siteinfo

Parameters

Parameter	Type	Required	Description	Valid Values
latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
area	string	✗	Spatial averaging area	Cell, Hectare, OneKm, TwoKm, ThreeKm, FiveKm
plotT	string	✗	Plot type for analysis	SoilF, CompF, SoilA, CompA, CompM
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
incGrowth	boolean	✗	Include growth parameters	true, false
version	integer	✗	FulCAM PR version	2020

Area Parameter Values

- **Cell:** No spatial averaging
- **Hectare:** 1 hectare averaging
- **OneKm:** 100 hectare averaging
- **TwoKm:** 400 hectare averaging
- **ThreeKm:** 900 hectare averaging
- **FiveKm:** 2500 hectare averaging

Plot Type Values

- **SoilF:** Forest soil analysis
- **CompF:** Forest system analysis
- **SoilA:** Agricultural soil analysis
- **CompA:** Agricultural system analysis
- **CompM:** Mixed (forest and agricultural) system analysis

Forest Category Values

- **null**: All categories
- **MVG**: Native Vegetation Groups
- **Plantation**: Plantation Species
- **EnvMallee**: Environmental and Mallees
- **ERF**: ERF Methods
- **ERFH**: ERF Methods-EMP specific calibrations

Example Request

```
GET /2020/data-builder/siteinfo?latitude=-  
27.35757265&longitude=140.3277883&area=ThreeKm&plotT=CompF&frCat=ERF&incGrowth=t  
rue&version=2020
```

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```
<DocFragment Version="5007">  
  
<SiteInfo latitude="-27.35757265" longitude="140.3277883" areaBL="ThreeKm" state="SA"  
npi="Blank" growthRegion="0" SA2="41141" sa2Name="Outback" mvgTreeId="10" />  
  
<LocnSoil initFracBiof="0.0" initFracBios="0.0" initFracDpma="0.0"  
initFracRpma="0.0769481897354" initFracHums="0.7504163146019"  
initFracInrt="0.1726354777813" initTotalC="9.6754886627197" soilCoverA="0.6">  
  
    <!-- Additional soil and climate data using 2020 parameters -->  
  
    <TimeSeries tInTS="initTSMD" tExtrapTS="AvgYr" tOriginTS="Calendar" yr0TS="1970"  
nYrsTS="49">  
  
        <!-- Time series data for soil moisture deficit -->  
  
    </TimeSeries>  
  
</LocnSoil>  
  
<!-- Species list for location using 2020 data -->  
  
<ItemList count="6" id="FrSpecies">  
  
    <ItemInfo id="106" value="Mallee eucalypt species" content="TYFParameters" />  
  
    <ItemInfo id="104" value="Native Species Regeneration >=500mm rainfall"  
content="TYFParameters" />  
  
    <ItemInfo id="105" value="Native Species Regeneration <500mm rainfall"  
content="TYFParameters" />  
  
    <ItemInfo id="35" value="Mixed species environmental planting" content="TYFParameters" />  
  
    <ItemInfo id="16" value="Eucalyptus globulus" content="TYFParameters" />
```

```

<ItemInfo id="44" value="Pinus radiata" content="TYFParameters" />
</ItemList>
</DocFragment>

```

2. species API

Retrieve detailed species information including carbon coefficients and growth parameters using 2020 calibrations.

API Endpoint: GET /2020/data-builder/species

Parameters

Parameter	Type	Required	Description	Valid Values
latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
area	string	✗	Spatial averaging area	Cell, Hectare, OneKm, TwoKm, ThreeKm, FiveKm
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
specId	string	✓	Species ID	Extract from Site Info API ItemInfo id
version	integer	✗	FulCAM PR version	2020

Getting Species ID

Species IDs should be extracted from the Site Info API response:

```

<ItemInfo id="23" value="Eucalyptus Species" content="Meta" />
<ItemInfo id="53" value="Pine Species" content="Meta" />

```

Use the id attribute value (e.g., "23", "53") as the specId parameter.

Example Request

```

GET /2020/data-builder/species?latitude=-
27.35757265&longitude=140.3277883&specId=1&frCat=ERF&version=2020

```

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```

<DocFragment Version="5007">
<SpeciesForest idSP="35" nmSP="Mixed species environmental planting" grthModeSP="Yield"
tAgeIxSP="AvgAge" curEdit="false" idRegimeSP="0" mvgTreId="82" pltnType="Other">

```

```

tSpecFrCat="EnvMallee" CFracStemF="0.5" CFracBranF="0.468" CFracBarkF="0.487"
CFracLeafF="0.529" CFracCortF="0.492" CFracFirtF="0.461">

<!-- Detailed carbon fraction and growth data using 2020 parameters -->

<YieldClass>
    <!-- Yield class specific parameters for 2020 -->
</YieldClass>
</SpeciesForest>
</DocFragment>

```

3. regimes API

Retrieve regime information for specific species and locations using 2020 protocols.

API Endpoint: GET /2020/data-builder/regimes

Parameters

Parameter	Type	Required	Description	Valid Values
latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
specId	string	✓	Species ID	Extract from Site Info API ItemInfo id
version	integer	✗	FulCAM PR version	2020

Example Request

```
GET /2020/data-builder/regimes?latitude=-27.35757265&longitude=140.3277883&specId=1&frCat=ERF&version=2020
```

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```

<DocFragment Version="5008">

<RegimeList idSP="1">

    <RegimeF idRG="58" tSuite="ERF" nmRG="Acacia mangium (ERF 2020)" idSpecies="1"
    tSpecFrCat="ERF" rotLenRGF="Long" stateRGF="Any" npiRGF="Blank" yrStRGF="1970"
    yrEnRGF="2100" prevLUseRGF="Pasture" sitePrepRGF="None" ppWeedCntRGF="None"
    cultRGF="Spot" thinRGF="Thin0" pruneRGF="Prune0">

```

```

<RegimeEvent seq="1" ruleSet="--212,CPN(CWSW)=-212GP" tevREV="PlnF" idSP="1"
eventNmREV="Plant trees" />

<!-- Additional regime events using 2020 protocols -->

</RegimeF>

</RegimeList>

</DocFragment>

```

4. templates API

Retrieve available carbon accounting templates for 2020 calculations.

API Endpoint: GET /2020/data-builder/templates

Parameters

Parameter	Type	Required	Description	Valid Values
version	integer	✗	FulCAM PR version	2020

Example Request

GET /2020/data-builder/templates?version=2020

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```

<DocFragment Version="5007">

<ItemList count="3" id="Templates">

  <ItemInfo id="1" value="ERF\Environmental Plantings Method.plo" content="Meta">

    <Meta nmME="Template - Environmental Plantings Method" savedByResearch="false"
    savedByVersion="" lockTime="" lockId="" lockOnME="false">

      <notesME></notesME>

    </Meta>

  </ItemInfo>

  <ItemInfo id="2" value="ERF\HIR NFMR and Avoided Clearing Methods greater than
  500mm.plo" content="Meta">

    <Meta nmME="Template - HIR NFMR and Avoided Clearing Methods"
    savedByResearch="false" savedByVersion="" lockTime="" lockId="" lockOnME="false">

      <notesME></notesME>

    </Meta>

  </ItemInfo>

```

```

<ItemInfo id="3" value="ERF\Additional Template.plo" content="Meta">
    <!-- Template metadata for 2020 -->
</ItemInfo>
</ItemList>
</DocFragment>

```

5. template API

Retrieve a specific template by name using 2020 parameters.

API Endpoint: GET /2020/data-builder/template

Parameters

Parameter	Type	Required	Description	Valid Values
templateName	string	<input checked="" type="checkbox"/>	Template name	Extract from Templates API ItemInfo value
version	integer	<input type="checkbox"/>	FulCAM PR version	2020 (int32 format)

Getting Template Name

Template names should be extracted from the Templates API response:

```

<ItemInfo id="1" value="ERF\Environmental Plantings Method 2020.plo" content="Meta" />
<ItemInfo id="2" value="ERF\HIR NFMR and Avoided Clearing Methods 2020.plo" content="Meta" />

```

Use the value attribute (e.g., "ERF\Environmental Plantings Method 2020.plo") as the templateName parameter.

Example Request

GET /2020/data-builder/template?templateName=ERF\Environmental Plantings Method 2020.plo&version=2020

Host: api.dccceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```

<DocFragment Version="5008">
    <DocumentPlot FileType="FullCAM Plot" Version="5008" pageIxDO="1" tDiagram="-1">
        <Meta nmME="Environmental Plantings Method 2020" savedByVersion="8.20.04.1101" lockTime="" lockId="" lockOnME="false">
            <notesME>2020 calibrated template for environmental plantings</notesME>
        </Meta>

```

```

<Config tPlot="CompF" userSoilMnrl="true" userMulchF="false" userMulchA="false">
    <!-- Detailed template configuration using 2020 parameters -->
</Config>
<!-- Complete template data structure with 2020 calibrations -->
</DocumentPlot>
</DocFragment>

```

6. update-spatialdata API

Update spatial data in FullCAM 2020 plot files while maintaining 2020 parameter compatibility.

API Endpoint: POST /2020/data-builder/update-spatialdata

Request Body

Parameter	Type	Required	Description
file	binary	<input checked="" type="checkbox"/>	Plot file requiring spatial data updates using 2020 parameters

Description

Upload a FullCAM 2020 plot file that requires spatial data updates. The API will refresh the spatial data using 2020 parameter sets and calibrations while maintaining compatibility with FullCAM 2020 PR workflows.

Key functionality:

- **Refresh location-specific data** using 2020 spatial datasets
- **Update soil information** with 2020 calibrated values
- **Maintain 2020 parameter compatibility** throughout the update process
- **Preserve existing plot structure** while updating spatial components

Content Type

- **Request:** multipart/form-data
- **File Format:** a

Example Request

POST /2020/data-builder/update-spatialdata

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Content-Type: multipart/form-data

--boundary

Content-Disposition: form-data; name="file"; filename="my-plot-file-2020.xml"

Content-Type: application/xml

[Plot file content requiring 2020 spatial data update]

--boundary--

Example Response

```
<DocFragment Version="5008">

<DocumentPlot FileType="FullCAM Plot" Version="5008" pageIxDO="1" tDiagram="-1">
    <!-- Updated plot file with refreshed 2020 spatial data -->
    <Meta nmME="Updated 2020 Plot File" savedByVersion="8.20.01.0000" />
    <SiteInfo latitude="-27.35757265" longitude="140.3277883" areaBL="ThreeKm" state="QLD" />
    <LocnSoil initFracBiof="0.0" initFracBios="0.0" initTotalC="45.0667" />
    <!-- Updated spatial data using 2020 parameters -->
</DocumentPlot>
</DocFragment>
```

Error Handling

HTTP Status Codes

The FullCAM API uses standard HTTP status codes. Note that successful requests return a 200 status response code, even if no resources match the query (in which case an empty result set will be returned).

Error Response Format

```
{
    "type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
    "title": "Error description",
    "status": 400,
    "detail": "Detailed error message",
    "traceId": "00-trace-id-here-00",
    "errors": {
        "parameter": ["Validation error message"]
    }
}
```

Common Error Codes

Status Code	Description	Common Causes
200	Success	Request completed successfully (may return empty results)
400	Bad Request	Invalid parameter values, malformed request
401	Unauthorized	Missing or invalid subscription key
404	Not Found	Invalid API Endpoint, missing required parameters
500	Internal Server Error	Invalid parameter combinations, server issues

Parameter Validation Errors

Invalid Coordinates

```
{  
  "status": 500,  
  "detail": "BadRequest - Invalid latitude & longitude details. longitude:200,latitude:-27.35757265"  
}
```

Invalid Parameter Format

```
{  
  "status": 400,  
  "title": "One or more validation errors occurred.",  
  "errors": {  
    "latitude": ["The value 'invalid' is not valid."]  
  }  
}
```

Invalid Species ID

```
<DocFragment Version="5008">  
  <ItemList count="1" id="Messages">  
    <ItemInfo id="1" value="Invalid SpeciesId" />  
  </ItemList>  
</DocFragment>
```

Development Best Practices

1. Model Objects

Define model objects according to the API documentation to handle appropriate responses from API calls. Structure your application to parse the XML response format properly. Note that 2020 responses use Version="5007" in XML fragments.

2. Asynchronous Calls

To avoid poor user experience with large delays, call the API's asynchronously rather than synchronously. This is especially important given the size of some responses (templates can be 700KB+).

3. Error Handling Mechanisms

Implement robust error handling to detect and respond to different HTTP status codes:

- **200:** Success (may include empty results)
- **400:** Client-side parameter errors
- **401:** Authentication failures
- **404:** API Endpoint not found
- **500:** Server-side processing errors

4. Caching Strategy

To improve performance and reduce API calls:

- **Cache spatial data locally** for quick searches rather than requesting from server each time
- **Refresh cache daily** to ensure you have the latest data
- **Cache templates and species data** as they change infrequently for 2020 version
- Consider implementing cache invalidation strategies

5. Version Compatibility

When working with FullCAM 2020 API:

- **Always specify version=2020** in parameters
- **Maintain 2020 parameter sets** when processing data
- **Be aware of differences** in carbon coefficients compared to 2024 version
- **Test compatibility** if migrating from 2020 to 2024 workflows

Performance Optimization

Response Data Sizes (2020 Version)

Be aware that API responses can be quite large:

- **Site Info:** ~38-40KB per location
- **Species Data:** ~260-270KB per species
- **Templates:** ~1.1KB per template list
- **Regimes:** ~1-2KB per species
- **Plot File Operations:** Variable size depending on input file complexity

File Upload Considerations (2020)

- **Update Spatial Data:** Refreshes spatial data using 2020 parameter sets
- **File Size Limits:** Test with your specific file sizes to determine limits
- **Timeout Settings:** Use 60+ second timeouts for file upload operations
- **Version Maintenance:** Ensures plot files maintain 2020 parameter compatibility

Caching Recommendations (2020)

- **Cache site information** for frequently used locations with 2020 parameters
- **Cache templates daily** as 2020 templates are stable
- **Cache species data** for commonly used species with 2020 calibrations
- **Implement version-specific caching** to avoid conflicts with 2024 data
- **Do not cache file upload results** as they are typically one-time operations

Request Optimization (2020)

- **Always specify version=2020** in requests to ensure correct parameter sets
- **Batch requests** for multiple species when possible
- **Use appropriate timeout values** (30+ seconds for data, 60+ seconds for file uploads)
- **Implement retry logic** with exponential backoff
- **Process file uploads sequentially** rather than in parallel
- **Monitor parameter compatibility** when migrating between 2020 and 2024 versions

Support and Contact

- **Email:** fullcam@dcceew.gov.au
 - **Terms of Service:** [FullCAM Terms](#)
 - **License Agreement:** [FullCAM License](#)
-

Changelog (2020)

Version 2020

- Legacy API maintaining compatibility with FullCAM 2020 PR desktop application
 - RESTful architecture with subscription-based authentication
 - Support for Australian-wide geographic coverage using 2020 spatial datasets
 - Comprehensive species and regime databases with 2020 calibrations
 - Template system for standardized calculations using 2020 parameters
 - **File Processing Capabilities:**
 - Plot file validation for FullCAM 2020 PR compatibility
 - Spatial data updates using 2020 parameter sets
 - Maintains compatibility with existing 2020 workflows
 - Cross-platform compatibility for web and non-web applications
 - **Legacy Support:** Ensures continuity for existing 2020-based carbon accounting workflows
-

FullCAM 2024 Data Builder API

Overview

The **Full Carbon Accounting Model (FullCAM) 2024 Data Builder API** provides programmatic access to Australia's comprehensive carbon accounting data for forestry and land use sectors. This RESTful API enables developers to retrieve site-specific information, species data, management regimes, and templates for carbon accounting calculations across various web and non-web based platforms.

Key Features

- **Geographic Site Information:** Soil, climate, and regional data for any Australian location
- **Species Database:** Comprehensive forest and agriculture species data with carbon coefficients
- **Management Regimes:** Detailed forest and agriculture management protocols and practices
- **Template System:** Pre-configured carbon accounting templates
- **Cross-Platform Compatibility:** Removes dependency on Windows-only desktop application
- **Real-time Data:** Current carbon accounting parameters and coefficients

API Architecture

FullCAM implements a REST (Representational State Transfer) architecture that is stateless, meaning servers do not save client data between requests. The API provides two main service categories:

1. **Data API:** Retrieves spatial data, species information, and templates
2. **Plot Simulation API:** Enables running simulations using existing plot files

API Access Request

To gain access to the FullCAM API, you must request a subscription key from the FullCAM business team.

Access Request Process

Send an email to **fullcam@dcceew.gov.au** with the following information:

1. **Organization Name**
2. **Email Address**
3. **Phone Number**
4. **Business Use Case:** Detailed description of intended usage

5. Usage Details:

- Number of requests per system intended
- Number of users
- Peak time of usage
- Any other relevant usage patterns

6. Number of Users

7. Access Duration: How long you need access

Response

Once approved, FullCAM will issue a **unique API subscription key** that must be included in all API requests.

API Authentication

All API requests require a subscription key for authentication.

Required Header

Ocp-Apim-Subscription-Key: {{YOUR_SUBSCRIPTION_KEY}}

Error Response (401)

```
{  
  "statusCode": 401,  
  "message": "Access denied due to missing subscription key. Make sure to include subscription  
key when making requests to an API."  
}
```

Base URL

<https://api.dcceew.gov.au/climate/carbon-accounting/2024/data/v1>

API Endpoints

1. siteinfo API

Retrieve comprehensive site information for a specific geographic location including soil, climate, and regional data.

API Endpoint: GET /2024/data-builder/siteinfo

Parameters

Parameter	Type	Required	Description	Valid Values
-----------	------	----------	-------------	--------------

latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
area	string	✗	Spatial averaging area	Cell, Hectare, OneKm, TwoKm, ThreeKm, FiveKm
plotT	string	✗	Plot type for analysis	SoilF, CompF, SoilA, CompA, CompM
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
incGrowth	boolean	✗	Include growth parameters	true, false
version	integer	✗	FulCAM PR version	2024

Area Parameter Values

- **Cell:** No spatial averaging
- **Hectare:** 1 hectare averaging
- **OneKm:** 100 hectare averaging
- **TwoKm:** 400 hectare averaging
- **ThreeKm:** 900 hectare averaging
- **FiveKm:** 2500 hectare averaging

Plot Type Values

- **SoilF:** Forest soil analysis
- **CompF:** Forest system analysis
- **SoilA:** Agricultural soil analysis
- **CompA:** Agricultural system analysis
- **CompM:** Mixed (forest and agricultural) system analysis

Forest Category Values

- **null:** All categories
- **MVG:** Native Vegetation Groups
- **Plantation:** Plantation Species
- **EnvMallee:** Environmental and Mallees
- **ERF:** ERF Methods
- **ERFH:** ERF Methods-EMP specific calibrations

Example Request

GET /2024/data-builder/siteinfo?latitude=-33.8688&longitude=151.2093&area=Cell&plotT=CompF&frCat>All&incGrowth=false

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```
<DocFragment Version="5009">

<SiteInfo latitude="-33.8688" longitude="151.2093" areaBL="Cell" state="NSW" npi="Blank"
growthRegion="2" SA2="11337" sa2Name="Sydney - Haymarket - The Rocks" mvgTreeId="3" />

<LocnSoil initFracBiof="0.0" initFracBios="0.0" initFracDpma="0.0"
initFracRpma="0.1751229614019" initFracHums="0.5600624084473"
initFracInrt="0.2648146152496" initTotalC="49.0667381286621" soilCoverA="0.6">

    <!-- Additional soil and climate data -->

</LocnSoil>

<!-- Species list for location -->

<ItemList count="145" id="Species">

    <ItemInfo id="1" value="Acacia mangium" content="Meta" />
    <ItemInfo id="23" value="Eucalyptus Species" content="Meta" />
    <ItemInfo id="53" value="Pine Species" content="Meta" />
    <!-- More species entries -->
</ItemList>

</DocFragment>
```

2. species API

Retrieve detailed species information including carbon coefficients and growth parameters.

API Endpoint: GET /2024/data-builder/species

Parameters

Parameter	Type	Required	Description	Valid Values
latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
area	string	✗	Spatial averaging area	Cell, Hectare, OneKm, TwoKm, ThreeKm, FiveKm
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
specId	string	✓	Species ID	Extract from Site Info API ItemInfo id
version	integer	✗	FulCAM PR version	2024

Getting Species ID

Species IDs should be extracted from the Site Info API response:

```
<ItemInfo id="23" value="Eucalyptus Species" content="Meta" />
<ItemInfo id="53" value="Pine Species" content="Meta" />
```

Use the id attribute value (e.g., "23", "53") as the specId parameter.

Example Request

```
GET /2024/data-builder/species?latitude=-  
33.8688&longitude=151.2093&specId=1&frCat=Plantation
```

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```
<DocFragment Version="5009">  
  
<SpeciesForest idSP="1" nmSP="Acacia mangium" grthModeSP="Yield" tAgeIxSP="AvgAge"  
curEdit="false" idRegimeSP="1" mvgTreId="97" pltnType="Hardwood" tSpecFrCat="Plantation"  
CFracStemF="0.5" CFracBranF="0.468" CFracBarkF="0.487" CFracLeafF="0.529"  
CFracCortF="0.492" CFracFirtF="0.461">  
  
<!-- Detailed carbon fraction and growth data -->  
  
<YieldClass>  
  
<!-- Yield class specific parameters -->  
  
</YieldClass>  
  
</SpeciesForest>  
  
</DocFragment>
```

3. regimes API

Retrieve management regime information for specific species and locations.

API Endpoint: GET /2024/data-builder/regimes

Parameters

Parameter	Type	Required	Description	Valid Values
latitude	number	✓	Latitude coordinate	-90 to 90
longitude	number	✓	Longitude coordinate	-180 to 180
frCat	string	✗	Forest category filter	null, MVG, Plantation, EnvMallee, ERF, ERFH
specId	string	✓	Species ID	Extract from Site Info API ItemInfo id
version	integer	✗	FulCAM PR version	2024

Example Request

```
GET /2024/data-builder/regimes?latitude=-33.8688&longitude=151.2093&specId=1&frCat=ERF
```

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```
<DocFragment Version="5009">

<RegimeList idSP="1">

  <RegimeF idRG="59" tSuite="NPI" nmRG="Acacia mangium (ERF)" idSpecies="1"
tSpecFrCat="All" rotLenRGF="Long" stateRGF="Any" npiRGF="Blank" yrStRGF="1970"
yrEnRGF="2100" prevLUseRGF="Pasture" sitePrepRGF="None" ppWeedCntRGF="None"
cultRGF="Spot" thinRGF="Thin0" pruneRGF="Prune0">

    <RegimeEvent seq="1" ruleSet="=212,CPN(CWSW)=-212GP" tevREV="PInF" idSP="1"
eventNmREV="Plant trees" />

    <!-- Additional regime events -->

  </RegimeF>

</RegimeList>

</DocFragment>
```

4. templates API

Retrieve available carbon accounting templates.

API Endpoint: GET /2024/data-builder/templates

Parameters

Parameter	Type	Required	Description	Valid Values
version	integer	X	FulCAM PR version	2024

Example Request

```
GET /2024/data-builder/templates

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY
```

Example Response

```
<DocFragment Version="5009">

<ItemList count="3" id="Templates">

  <ItemInfo id="1" value="ERF\Environmental Plantings Method.plo" content="Meta">

    <Meta nmME="Environmental Plantings Method" savedByVersion="8.23.04.1101"
lockTime="" lockId="" lockOnME="false">

      <notesME></notesME>

    </Meta>

  </ItemInfo>

  <ItemInfo id="2" value="ERF\HIR NFMR and Avoided Clearing Methods greater than
500mm.plo" content="Meta">
```

```

<Meta nmME="HIR NFMR and Avoided Clearing Methods" savedByVersion="8.23.04.1101"
lockTime="" lockId="" lockOnME="false">

  <notesME></notesME>

</Meta>

</ItemInfo>

<ItemInfo id="3" value="ERF\Additional Template.plo" content="Meta">

  <!-- Template metadata -->

</ItemInfo>

</ItemList>

</DocFragment>

```

5. template API

Retrieve a specific template by name.

API Endpoint: GET /2024/data-builder/template

Parameters

Parameter	Type	Required	Description	Valid Values
templateName	string	✓	Template name	Extract from Templates API ItemInfo value
version	integer	✗	FulCAM PR version	2024 (int32 format)

Getting Template Name

Template names should be extracted from the Templates API response:

```

<ItemInfo id="1" value="ERF\Environmental Plantings Method.plo" content="Meta" />

<ItemInfo id="2" value="ERF\HIR NFMR and Avoided Clearing Methods greater than
500mm.plo" content="Meta" />

```

Use the value attribute (e.g., "ERF\HIR NFMR and Avoided Clearing Methods greater than 500mm.plo") as the templateName parameter.

Example Request

GET /2024/data-builder/template?templateName=ERF\HIR NFMR and Avoided Clearing Methods greater than 500mm.plo&version=2024

Host: api.dccceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Example Response

```

<DocFragment Version="5009">

  <DocumentPlot FileType="FullCAM Plot" Version="5009" pageIxDO="1" tDiagram="-1">

```

```

<Meta nmME="HIR NFMR and Avoided Clearing Methods" savedByVersion="8.23.04.1101"
lockTime="" lockId="" lockOnME="false">

<notesME></notesME>

</Meta>

<Config tPlot="CompF" userSoilMnrl="true" userMulchF="false" userMulchA="false">
    <!-- Detailed template configuration -->
</Config>

    <!-- Complete template data structure -->
</DocumentPlot>

</DocFragment>

```

6. convert-plotfile API

Convert FullCAM plot files from 2020 PR version to 2024 PR version.

API Endpoint: POST /2024/data-builder/convert-plotfile

Request Body

Parameter	Type	Required	Description
file	binary	<input checked="" type="checkbox"/>	plot file for version conversion

Description

Upload a FullCAM plot file that needs to be updated from FullCAM 2020 PR to FullCAM 2024 PR. The API will process the uploaded XML file and return the converted version with updated parameters and structure compatible with FullCAM 2024.

Content Type

- **Request:** multipart/form-data
- **File Format:** Plot file (.plo)

Example Request

POST /2024/data-builder/convert-plotfile

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Content-Type: multipart/form-data

--boundary

Content-Disposition: form-data; name="file"; filename="my-plot-2020.xml"

Content-Type: application/xml

[Plot file content from FullCAM 2020 PR]

--boundary--

Example Response

```
<DocFragment Version="5009">  
  <DocumentPlot FileType="FullCAM Plot" Version="5009" pageIxDO="1" tDiagram="-1">  
    <!-- Converted FullCAM 2024 PR plot file content -->  
    <Meta nmME="Converted Plot File" savedByVersion="8.24.01.0000" />  
    <!-- Updated parameters and structure for 2024 PR -->  
  </DocumentPlot>  
</DocFragment>
```

7. update-spatialdata

Update spatial data in FullCAM plot files and optionally upgrade from 2020 PR to 2024 PR version.

API Endpoint: POST /2024/data-builder/update-spatialdata

Request Body

Parameter	Type	Required	Description
file	binary	<input checked="" type="checkbox"/>	plot file requiring spatial data updates

Description

Upload a FullCAM plot file that requires spatial data updates. The API behavior depends on the input file version:

- **FullCAM 2020 PR plot files:** The API will generate a plot file upgraded to FullCAM 2024 PR version, including updated spatial data with current parameters and coefficients.
- **FullCAM 2024 PR plot files:** The API will generate a plot file with refreshed spatial data, updating coordinates, soil information, climate data, and other location-specific parameters to current values.

Content Type

- **Request:** multipart/form-data
- **File Format:** Plot file (.plo)

Example Request

POST /2024/data-builder/update-spatialdata

Host: api.dcceew.gov.au

Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY

Content-Type: multipart/form-data

--boundary

Content-Disposition: form-data; name="file"; filename="my-plot-file.xml"

Content-Type: application/xml

[Plot file content requiring spatial data update]

--boundary--

Example Response

```
<DocFragment Version="5009">

<DocumentPlot FileType="FullCAM Plot" Version="5009" pageIxDO="1" tDiagram="-1">
    <!-- Updated plot file with refreshed spatial data -->
    <Meta nmME="Updated Plot File" savedByVersion="8.24.01.0000" />
    <SiteInfo latitude="-33.8688" longitude="151.2093" areaBL="Cell" state="NSW" />
    <LocnSoil initFracBiof="0.0" initFracBios="0.0" initTotalC="49.0667" />
    <!-- Updated spatial data and parameters -->
</DocumentPlot>
</DocFragment>
```

Error Handling

HTTP Status Codes

The FullCAM API uses standard HTTP status codes. Note that successful requests return a 200 status response code, even if no resources match the query (in which case an empty result set will be returned).

Error Response Format

```
{
    "type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
    "title": "Error description",
    "status": 400,
    "detail": "Detailed error message",
    "traceId": "00-trace-id-here-00",
    "errors": {
```

```
        "parameter": ["Validation error message"]  
    }  
}
```

Common Error Codes

Status Code	Description	Common Causes
200	Success	Request completed successfully (may return empty results)
400	Bad Request	Invalid parameter values, malformed request
401	Unauthorized	Missing or invalid subscription key
404	Not Found	Invalid API Endpoint, missing required parameters
500	Internal Server Error	Invalid parameter combinations, server issues

Parameter Validation Errors

Invalid Coordinates

```
{  
  "status": 500,  
  "detail": "BadRequest - Invalid latitude & longitude details. longitude:200,latitude:-33.8688"  
}
```

Invalid Parameter Format

```
{  
  "status": 400,  
  "title": "One or more validation errors occurred.",  
  "errors": {  
    "latitude": ["The value 'invalid' is not valid."]  
  }  
}
```

Invalid Species ID

```
<DocFragment Version="5009">  
  <ItemList count="1" id="Messages">  
    <ItemInfo id="1" value="Invalid SpeciesId" />  
  </ItemList>  
</DocFragment>
```

Development Best Practices

1. Model Objects

Define model objects according to the API documentation to handle appropriate responses from API calls. Structure your application to parse the XML response format properly.

2. Asynchronous Calls

To avoid poor user experience with large delays, call the web services asynchronously rather than synchronously. This is especially important given the size of some responses (templates can be 700KB+).

3. Error Handling Mechanisms

Implement robust error handling to detect and respond to different HTTP status codes:

- **200:** Success (may include empty results)
- **400:** Client-side parameter errors
- **401:** Authentication failures
- **404:** API Endpoint not found
- **500:** Server-side processing errors

4. Caching Strategy

To improve performance and reduce API calls:

- **Cache spatial data locally** for quick searches rather than requesting from server each time
- **Refresh cache daily** to ensure you have the latest data
- **Cache templates and species data** as they change infrequently
- Consider implementing cache invalidation strategies

5. Rate Limiting Considerations

While not explicitly documented, implement reasonable usage patterns:

- Avoid excessive concurrent requests
- Implement retry logic with exponential backoff
- Monitor your usage against your requested quotas

API Usage Workflow

Typical Carbon Accounting Data Retrieval Workflow

1. **Get Site Information** → Extract available species IDs and location-specific data

2. **Get Species Data** → For each relevant species using extracted IDs
3. **Get Management Regimes** → For specific management scenarios
4. **Get Templates** → For standardized calculation frameworks

Plot File Processing Workflow

1. **Convert Legacy Files** → Use convert-plotfile to upgrade FullCAM 2020 PR files to 2024 PR
 2. **Update Spatial Data** → Use update-spatialdata to refresh location data in existing plot files
 3. **Template Integration** → Download templates and integrate with updated plot files
-

Performance Optimization

Response Data Sizes

Be aware that API responses can be quite large:

- **Site Info:** ~40-60KB per location
- **Species Data:** ~200-220KB per species
- **Templates:** ~750KB per template
- **Regimes:** ~1-2KB per species
- **Plot File Operations:** Variable size depending on input file complexity

File Upload Considerations

- **Convert Plot File:** Processing time depends on file complexity (typically 5-30 seconds)
- **Update Spatial Data:** May involve version upgrades, allow extra processing time
- **File Size Limits:** Test with your specific file sizes to determine limits
- **Timeout Settings:** Use 60+ second timeouts for file upload operations

Caching Recommendations

- **Cache site information** for frequently used locations
- **Cache templates daily** as they change infrequently
- **Cache species data** for commonly used species
- **Implement cache invalidation** on a daily schedule
- **Do not cache file upload results** as they are typically one-time operations

Request Optimization

- **Batch requests** for multiple species when possible
- **Use appropriate timeout values** (30+ seconds for data, 60+ seconds for file uploads)

- **Implement retry logic** with exponential backoff
 - **Limit concurrent requests** to avoid overwhelming the API
 - **Process file uploads sequentially** rather than in parallel
-

Support and Contact

- **Email:** fullcam@dcceew.gov.au
 - **Terms of Service:** [FullCAM Terms](#)
 - **License Agreement:** [FullCAM License](#)
-

Changelog (2024)

Version 2024

- Modernized API replacing Windows-only desktop application
 - RESTful architecture with subscription-based authentication
 - Support for Australian-wide geographic coverage
 - Comprehensive species and regime databases
 - Template system for standardized calculations
 - **File Processing Capabilities:**
 - Plot file conversion from FullCAM 2020 PR to 2024 PR
 - Spatial data updates for existing plot files
 - Version upgrade automation during spatial data refresh
 - Cross-platform compatibility for web and non-web applications
-

Version Differences: 2020 vs 2024

Key Differences to Consider

Aspect	FullCAM 2020	FullCAM 2024	Migration Impact
Base URL	/carbon-accounting/2020/	/carbon-accounting/2024/	URL structure change
XML Version	Version="5007"	Version="5009"	Response format difference
Subscription Key	2020-specific key required	2024-specific key required	Separate authentication

Aspect	FullCAM 2020	FullCAM 2024	Migration Impact
Carbon Coefficients	2020 calibrated values	Updated 2024 calibrations	Data value differences
Parameter Sets	2020 parameter standards	Enhanced 2024 parameters	Calculation differences
Template Names	Standard template names	Template names may differ	File compatibility
Species Data	2020 species calibrations	Updated species data	Carbon fraction differences
Response Size	~38KB site info	~43KB site info	Slightly more compact

Migration Considerations

When transitioning from FullCAM 2020 to 2024:

- Data Validation:** Expect differences in carbon coefficients and calculations
- File Compatibility:** Use conversion API Endpoints to migrate plot files
- Parameter Updates:** Review and update any hardcoded parameter values
- Testing:** Thoroughly test calculations with both versions for comparison
- Documentation:** Update any documentation that references specific parameter values

Glossary

Term	Definition
API	Application Programming Interface
REST API	Representational State Transfer Application Programming Interface
FullCAM PR	Full Carbon Accounting Model Public Release
FullCAM PR version	Full Carbon Accounting Model Public Release Version (2020, 2024)
API Authentication	An API authentication is the process of verifying the identity of a user or application before granting access to an API
Base URL	An API Base URL is the primary or starting URL for an API, acting as a common prefix for all subsequent API Endpoints
API Endpoint	An API Endpoint is a specific location (typically a URL) on a server where an API (Application Programming Interface) receives requests for resources or functionality
API Request	An API request, also known as an API call, is a message sent from a client application to a server to access a specific resource or perform an action via an API (Application Programming Interface)
API Response	An API response is the data returned by a server in response to a client's API request

DRAFT