

# dRofus REST API

This document provides reference materials for using the dRofus REST API for data from a given database. For the database management REST API, see this: [Admin system REST API](#)

This API is providing an [Open API specification](#) describing the API. We also have an interactive GUI that lets you explore this API and try it out in the browser. Since we expose and document how to get database/project specific properties we require you to login to see the documentation. See [Authentication](#) below.

## Provide feedback

Do not hesitate to contact us and provide feedback on what use cases you would like to use the API for.

## URLS

The API is currently deployed to the following locations. More will follow. The API documentation will expose project specific properties so you need to log in to see it. Please see [Authentication](#) section below for how to connect with your dRofus username and password.

URL	Swagger Gui	Swagger JSON
<a href="https://api-no.drofus.com">https://api-no.drofus.com</a>	<a href="#">API documentation and test it in web GUI</a>	<a href="#">Open API Specification</a>
<a href="https://api-ca.drofus.com">https://api-ca.drofus.com</a>	<a href="#">API documentation and test it in web GUI</a>	<a href="#">Open API Specification</a>
<a href="https://api-us.drofus.com">https://api-us.drofus.com</a>	<a href="#">API documentation and test it in web GUI</a>	<a href="#">Open API Specification</a>
<a href="https://api-au.drofus.com/">https://api-au.drofus.com/</a>	<a href="#">API documentation and test it in web GUI</a>	<a href="#">Open API Specification</a>

## Authentication

Currently we use basic authentication (username and password). Username is on the form <db-name>/[project-id/]user-name[/language], where:

Name	Meaning
db-name	The database-name as you would enter in dRofus Client
project-id	Id of project. Usually '01'. This is optional only if database only contains one project
user-name	The name you log on with in dRofus client
language	The language you want to see data in. Optional, and default is the language configured for the database

We will provide more authentication methods in the future

## Queries

We model our query-syntax on the [OData](#) standard, but currently only support a small subset of it.

Argument	Description	Sample
\$select	A comma-separated list of columns to return. 'id' will always be returned	\$select=name, architect_no
\$orderby	A comma-separated list of columns to sort by. Sorting direction can be specified using <b>asc</b> or <b>desc</b> . Default is <b>asc</b> . If no orderby is specified, the result is sorted by id	\$orderby=name, architect_no desc,

\$filter	<p>Used to restrict which rows are returned. A column can have a criteria to filter by. The value for filter can be quoted to handle embedded spaces and strings. Currently only 'and' can be used to combine multiple filters.</p> <p>Operators:</p> <table border="1" data-bbox="272 260 662 674"> <thead> <tr> <th>Operator</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Eq</td> <td>Equals</td> </tr> <tr> <td>Ne</td> <td>Not equal</td> </tr> <tr> <td>Lt</td> <td>Less than</td> </tr> <tr> <td>Gt</td> <td>Greater than</td> </tr> <tr> <td>Le</td> <td>Less than or equal</td> </tr> <tr> <td>Ge</td> <td>Greater than or equal</td> </tr> <tr> <td>In</td> <td>Is a member of</td> </tr> <tr> <td>Contains</td> <td>String contains a sub-string</td> </tr> </tbody> </table>	Operator	Description	Eq	Equals	Ne	Not equal	Lt	Less than	Gt	Greater than	Le	Less than or equal	Ge	Greater than or equal	In	Is a member of	Contains	String contains a sub-string	<p>\$filter=created gt '2019-1-1'</p> <p>\$filter=name in ('kitchen', 'office')</p> <p>\$filter=id in (1001, 1003, 1005)</p> <p>\$filter=contains (name,'kitch')</p>
Operator	Description																			
Eq	Equals																			
Ne	Not equal																			
Lt	Less than																			
Gt	Greater than																			
Le	Less than or equal																			
Ge	Greater than or equal																			
In	Is a member of																			
Contains	String contains a sub-string																			
\$stop	<p>Max number of rows to return. Default is 1000. If more data is available than is returned, the result will contain a <a href="#">RFC 5988</a> header value</p>	\$stop=123																		
\$skip	<p>How many rows to skip on result-set. Default is 0. Can be combined with \$stop to implement paging</p>	\$skip=100																		

## RFC5988

Rfc5988 is an internet standard for navigating between pages in result-sets. Essentially it looks like this, and contains the full link for additional data:

```
<http://localhost/rooms?$skip=1&$top=1>; rel="next">
```

## Examples

Examples of how to use the API: