

---

# **invenio-marc21 Documentation**

***Release 1.0.0a8***

**CERN**

**Mar 25, 2018**



---

## Contents

---

<b>1</b>	<b>User's Guide</b>	<b>3</b>
1.1	Installation . . . . .	3
1.2	Usage . . . . .	3
<b>2</b>	<b>API Reference</b>	<b>5</b>
2.1	API Docs . . . . .	5
<b>3</b>	<b>Additional Notes</b>	<b>9</b>
3.1	Contributing . . . . .	9
3.2	Changes . . . . .	11
3.3	License . . . . .	11
3.4	Contributors . . . . .	11
	<b>Python Module Index</b>	<b>13</b>



Invenio module with nice defaults for MARC21 overlay.

*This is an experimental developer preview release.*

- Free software: MIT license
- Documentation: <https://invenio-marc21.readthedocs.io/>



This part of the documentation will show you how to get started in using Invenio-MARC21.

## 1.1 Installation

## 1.2 Usage

Invenio module with nice defaults for MARC21 overlay.





If you are looking for information on a specific function, class or method, this part of the documentation is for you.

## 2.1 API Docs

Invenio module with nice defaults for MARC21 overlay.

**class** `invenio_marc21.ext.InvenioMARC21` (*app=None*)

Invenio-MARC21 extension.

Extension initialization.

**Parameters** `app` – An instance of `flask.Flask`.

**init\_app** (*app*)

Flask application initialization.

**Parameters** `app` – An instance of `flask.Flask`.

**init\_config** (*app*)

Initialize configuration.

**Parameters** `app` – An instance of `flask.Flask`.

### 2.1.1 Serializers

Base class for dojson based serializers.

**class** `invenio_marc21.serializers.dojson.DoJSONSerializer` (*dojson\_model*, *re-place\_refs=False*)

Base class for marshmallow serializers.

Initialize serializer.

**Parameters**

- **dojson\_model** – The DoJSON model able to convert JSON through the `do()` function.
- **replace\_refs** – Boolean value to configure if the `$ref` keys are replaced. (Default: `False`)

**dump** (*obj*)

Serialize object with schema.

**Parameters** *obj* – The object to serialize.

**Returns** The object serialized.

**transform\_record** (*pid, record, links\_factory=None*)

Transform record into an intermediate representation.

**Parameters**

- **pid** – The `invenio_pidstore.models.PersistentIdentifier` instance.
- **record** – The `invenio_records.api.Record` instance.
- **links\_factory** – The link factory. (Default: `None`)

**Returns** The intermediate representation for the record.

**transform\_search\_hit** (*pid, record\_hit, links\_factory=None*)

Transform search result hit into an intermediate representation.

**Parameters**

- **pid** – The `invenio_pidstore.models.PersistentIdentifier` instance.
- **record\_hit** – A dictionary containing a `'_source'` key with the record data.
- **links\_factory** – The link factory. (Default: `None`)

**Returns** The intermediate representation for the record.

MARC21 based serializer.

```
class invenio_marc21.serializers.marcxml.MARCXMLSerializer (dojson_model,  
                                                         xslt_filename=None,  
                                                         schema_class=None,  
                                                         replace_refs=False)
```

DoJSON based MARCXML serializer for records.

Note: This serializer is not suitable for serializing large number of records due to high memory usage.

Initialize serializer.

**Parameters**

- **dojson\_model** – The DoJSON model able to convert JSON through the `do()` function.
- **xslt\_filename** – XSLT filename. (Default: `None`)
- **schema\_class** – The schema class. (Default: `None`)
- **replace\_refs** – Boolean value to configure if replace the `$ref` keys within the JSON. (Default: `False`)

**dump** (*obj*)

Serialize object with schema.

**Parameters** *obj* – The object to serialize.

**Returns** The object serialized.

**serialize** (*pid*, *record*, *links\_factory=None*)

Serialize a single record and persistent identifier.

**Parameters**

- **pid** – The `invenio_pidstore.models.PersistentIdentifier` instance.
- **record** – The `invenio_records.api.Record` instance.
- **links\_factory** – Factory function for the link generation, which are added to the response.

**Returns** The object serialized.

**serialize\_oaipmh** (*pid*, *record*)

Serialize a single record for OAI-PMH.

**Parameters**

- **pid** – The `invenio_pidstore.models.PersistentIdentifier` instance.
- **record** – The `invenio_records.api.Record` instance.

**Returns** The object serialized.

**serialize\_search** (*pid\_fetcher*, *search\_result*, *item\_links\_factory=None*, *\*\*kwargs*)

Serialize a search result.

**Parameters**

- **pid\_fetcher** – Persistent identifier fetcher.
- **search\_result** – Elasticsearch search result.
- **item\_links\_factory** – Factory function for the items in result. (Default: None)

**Returns** The objects serialized.



Notes on how to contribute, legal information and changes are here for the interested.

## 3.1 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

### 3.1.1 Types of Contributions

#### Report Bugs

Report bugs at <https://github.com/inveniosoftware/invenio-marc21/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

#### Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

#### Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

## Write Documentation

Invenio-MARC21 could always use more documentation, whether as part of the official Invenio-MARC21 docs, in docstrings, or even on the web in blog posts, articles, and such.

## Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/inveniosoftware/invenio-marc21/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

### 3.1.2 Get Started!

Ready to contribute? Here's how to set up *invenio* for local development.

1. Fork the *invenio* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/invenio-marc21.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv invenio-marc21
$ cd invenio-marc21/
$ pip install -e .[all]
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass tests:

```
$ ./run-tests.sh
```

The tests will provide you with test coverage and also check PEP8 (code style), PEP257 (documentation), flake8 as well as build the Sphinx documentation and run doctests.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -s -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

### 3.1.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests and must not decrease test coverage.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring.
3. The pull request should work for Python 2.7, 3.3, 3.4 and 3.5. Check [https://travis-ci.com/inveniosoftware/invenio-marc21/pull\\_requests](https://travis-ci.com/inveniosoftware/invenio-marc21/pull_requests) and make sure that the tests pass for all supported Python versions.

## 3.2 Changes

Version 1.0.0 (released 2018-03-23)

- Initial public release.

## 3.3 License

MIT License

Copyright (C) 2016-2018 CERN.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

**Note:** In applying this license, CERN does not waive the privileges and immunities granted to it by virtue of its status as an Intergovernmental Organization or submit itself to any jurisdiction.

---

## 3.4 Contributors

- Alexander Ioannidis
- Alizee Pace
- Dinos Kousidis
- Javier Delgado
- Javier Martin Montull

- Jiri Kuncar
- Krzysztof Nowak
- Lars Holm Nielsen
- Leonardo Rossi
- Sebastian Witowski
- Tibor Simko



### i

- `invenio_marc21`, [3](#)
- `invenio_marc21.ext`, [5](#)
- `invenio_marc21.serializers.dojson`, [5](#)
- `invenio_marc21.serializers.marcxml`, [6](#)



## D

DoJSONSerializer (class in invenio\_marc21.serializers.dojson), 5  
dump() (invenio\_marc21.serializers.dojson.DoJSONSerializer method), 6  
dump() (invenio\_marc21.serializers.marcxml.MARCXMLSerializer method), 6

## I

init\_app() (invenio\_marc21.ext.InvenioMARC21 method), 5  
init\_config() (invenio\_marc21.ext.InvenioMARC21 method), 5  
invenio\_marc21 (module), 3  
invenio\_marc21.ext (module), 5  
invenio\_marc21.serializers.dojson (module), 5  
invenio\_marc21.serializers.marcxml (module), 6  
InvenioMARC21 (class in invenio\_marc21.ext), 5

## M

MARCXMLSerializer (class in invenio\_marc21.serializers.marcxml), 6

## S

serialize() (invenio\_marc21.serializers.marcxml.MARCXMLSerializer method), 6  
serialize\_oaipmh() (invenio\_marc21.serializers.marcxml.MARCXMLSerializer method), 7  
serialize\_search() (invenio\_marc21.serializers.marcxml.MARCXMLSerializer method), 7

## T

transform\_record() (invenio\_marc21.serializers.dojson.DoJSONSerializer method), 6  
transform\_search\_hit() (invenio\_marc21.serializers.dojson.DoJSONSerializer method), 6