
Invenio-Query-Parser Documentation

Release 0.3.0

Invenio collaboration

July 29, 2015

| | | |
|----------|----------------------------|----------|
| 1 | Contents | 3 |
| 1.1 | Installation | 3 |
| 1.2 | Usage | 3 |
| 1.3 | API | 3 |
| 1.4 | Changes | 7 |
| 1.5 | Contributing | 7 |
| 1.6 | Authors | 7 |
| | Python Module Index | 9 |

Search query parser supporting Invenio and SPIRES search syntax.

Contents

- *Installation*
- *Usage*
- *API*
- *Changes*
- *Contributing*
- *Authors*

1.1 Installation

Invenio-Query-Parser is on PyPI so all you need is:

```
$ pip install invenio-query-parser
```

1.2 Usage

The easiest way is to use *pyppeg2* directly with *Main*.

```
import pyppeg2
from invenio_query_parser.parser import Main
pyppeg2.parse('author:"Ellis"', Main)
```

1.3 API

API for Invenio-Query-Parser. Define abstract classes.

```
class invenio_query_parser.ast.AndOp(left, right)
```

```
class invenio_query_parser.ast.BinaryKeywordBase(left, right)
```

```
    keyword
```

```
class invenio_query_parser.ast.BinaryOp(left, right)
```

```

    accept (visitor)
class invenio_query_parser.ast.DoubleQuotedValue (value)
class invenio_query_parser.ast.EmptyQuery (value)
class invenio_query_parser.ast.GreaterEqualOp (op)
class invenio_query_parser.ast.GreaterOp (op)
class invenio_query_parser.ast.Keyword (value)
class invenio_query_parser.ast.KeywordOp (left, right)
class invenio_query_parser.ast.Leaf (value)

```

```

    accept (visitor)
class invenio_query_parser.ast.ListOp (children)

```

```

    accept (visitor)
class invenio_query_parser.ast.LowerEqualOp (op)
class invenio_query_parser.ast.LowerOp (op)
class invenio_query_parser.ast.NotOp (op)

```

```

    keyword
class invenio_query_parser.ast.OrOp (left, right)
class invenio_query_parser.ast.RangeOp (left, right)
class invenio_query_parser.ast.RegexValue (value)
class invenio_query_parser.ast.SingleQuotedValue (value)
class invenio_query_parser.ast.UnaryOp (op)

```

```

    accept (visitor)
class invenio_query_parser.ast.Value (value)
class invenio_query_parser.ast.ValueQuery (op)

```

Define parsers.

```

class invenio_query_parser.parser.And

```

```

    grammar = (-6, [<_sre.SRE_Pattern object at 0x7f7c5f6a9718>, Literal('+')])
class invenio_query_parser.parser.AndQuery

```

```

    grammar = [((-6, <class 'invenio_query_parser.parser.And'>), [((-6, <class 'invenio_query_parser.parser.Whitespace'>), A
class invenio_query_parser.parser.BinaryRule
class invenio_query_parser.parser.DoubleQuotedString

```

```

    grammar = (Literal(''), Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f720e90>, subtype=None), L

```



```
class invenio_query_parser.parser.EmptyQueryRule
```

```
    grammar = Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f6a98b0>, subtype=None)
```

```
class invenio_query_parser.parser.ImplicitAndQuery
```

```
    grammar = [Attribute(name='op', thing=<class 'invenio_query_parser.parser.ParenthesizedQuery'>, subtype=None), At
```

```
class invenio_query_parser.parser.KeywordQuery
```

```
    grammar = [(Attribute(name='left', thing=<class 'invenio_query_parser.parser.KeywordRule'>, subtype=None), (-6, ((0,
```

```
class invenio_query_parser.parser.KeywordRule
```

```
    grammar = Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f752a30>, subtype=None)
```

```
class invenio_query_parser.parser.LeafRule
```

```
class invenio_query_parser.parser.ListRule
```

```
class invenio_query_parser.parser.Main
```

```
    grammar = [((-6, (0, <class 'invenio_query_parser.parser.Whitespace'>)), Attribute(name='op', thing=<class 'invenio_qu
```

```
class invenio_query_parser.parser.NestableKeyword
```

```
    grammar = Attribute(name='value', thing=[<_sre.SRE_Pattern object at 0x7f7c5f6aaea0>, <_sre.SRE_Pattern object at 0
```

```
class invenio_query_parser.parser.Not
```

```
    grammar = (-6, [(-6, <_sre.SRE_Pattern object at 0x7f7c5f70df08>), <_sre.SRE_Pattern object at 0x7f7c5f751c68>, Litera
```

```
class invenio_query_parser.parser.NotQuery
```

```
    grammar = [((-6, <class 'invenio_query_parser.parser.Not'>), [((-6, <class 'invenio_query_parser.parser.Whitespace'>), A
```

```
class invenio_query_parser.parser.Number
```

```
    grammar = Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f6b8880>, subtype=None)
```

```
class invenio_query_parser.parser.Or
```

```
    grammar = (-6, [<_sre.SRE_Pattern object at 0x7f7c5f5e34b0>, Literal('')])
```

```
class invenio_query_parser.parser.OrQuery
```

```
    grammar = [((-6, <class 'invenio_query_parser.parser.Or'>), [((-6, <class 'invenio_query_parser.parser.Whitespace'>), At
```

```
class invenio_query_parser.parser.ParenthesizedQuery
```

```
    grammar = ((-6, (Literal('('), (0, <class 'invenio_query_parser.parser.Whitespace'>))), Attribute(name='op', thing=<class
```

```
class invenio_query_parser.parser.Query
```

```

    grammar = Attribute(name='children', thing=[<class 'invenio_query_parser.parser.ParenthesizedQuery'>, <class 'invenio_query_parser.parser.RangeOp'>])
class invenio_query_parser.parser.RangeOp

```

```

    grammar = (Attribute(name='left', thing=<class 'invenio_query_parser.parser.RangeValue'>, subtype=None), Literal('-', <class 'invenio_query_parser.parser.RangeValue'>))
class invenio_query_parser.parser.RangeValue

```

```

    grammar = Attribute(name='op', thing=[<class 'invenio_query_parser.parser.DoubleQuotedString'>, <class 'invenio_query_parser.parser.SimpleQuery'>])
class invenio_query_parser.parser.SimpleQuery

```

```

    grammar = Attribute(name='op', thing=[<class 'invenio_query_parser.parser.KeywordQuery'>, <class 'invenio_query_parser.parser.SimpleRangeValue'>])
class invenio_query_parser.parser.SimpleRangeValue

```

```

    grammar = Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f753030>, subtype=None)
class invenio_query_parser.parser.SimpleValue (values)

```

```

    grammar = (-2, <class 'invenio_query_parser.parser.SimpleValueUnit'>)
class invenio_query_parser.parser.SimpleValueUnit (args)

```

```

    grammar = [<_sre.SRE_Pattern object at 0x7f7c5f5fe030>, (<_sre.SRE_Pattern object at 0x7f7c5f6a97a0>, <class 'invenio_query_parser.parser.SingleQuotedString'>)]
class invenio_query_parser.parser.SingleQuotedString

```

```

    grammar = (Literal('"'), Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f720dd8>, subtype=None), <class 'invenio_query_parser.parser.SlashQuotedString'>))
class invenio_query_parser.parser.SlashQuotedString

```

```

    grammar = (Literal('/'), Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f720f48>, subtype=None), <class 'invenio_query_parser.parser.UnaryRule'>))
class invenio_query_parser.parser.UnaryRule
class invenio_query_parser.parser.Value

```

```

    grammar = Attribute(name='op', thing=[<class 'invenio_query_parser.parser.RangeOp'>, <class 'invenio_query_parser.parser.ValueQuery'>])
class invenio_query_parser.parser.ValueQuery

```

```

    grammar = Attribute(name='op', thing=<class 'invenio_query_parser.parser.Value'>, subtype=None)
class invenio_query_parser.parser.Whitespace

```

```

    grammar = Attribute(name='value', thing=<_sre.SRE_Pattern object at 0x7f7c5f6b87e8>, subtype=None)
Store the actual visitor methods.

```

```

class invenio_query_parser.visitor.make_visitor (methods=None)
    Make a visitor decorator.

```

1.4 Changes

Version 0.3.0 (released 2015-07-29):

- Allows search keywords to contain dots to point to subfield content, i.e. *author.name: Ellis*.

Version 0.2.0 (released 2014-12-10):

- Initial public release.
- Adds Python2/Python3 compatibility layer. (#2)
- Adds new Sphinx documentation page. (#3)
- Adds simple inheritance for visitor pattern and separates SPIRES syntax parser to contrib module. (#7)

Version 0.1.0 (release 2014-10-13):

- Initial private release. Includes code developed by Alessio Deiana and Federico Poli.

1.5 Contributing

Bug reports, feature requests, and other contributions are welcome. If you find a demonstrable problem that is caused by the code of this library, please:

1. Search for [already reported problems](#).
2. Check if the issue has been fixed or is still reproducible on the latest *master* branch.
3. Create an issue with **a test case**.

If you create a feature branch, you can run the tests to ensure everything is operating correctly:

```
$ python setup.py test
...
===== 146 passed in 1.90 seconds =====
```

1.6 Authors

Invenio-Query-Parser is developed for the [Invenio](#) digital library software.

Contact us at info@invenio-software.org.

- Alessio Deiana <<https://github.com/osso>>
- Federico Poli <<https://github.com/fpoli>>
- Jiri Kuncar <<https://github.com/jirikuncar>>
- Tibor Simko <<https://github.com/tiborsimko>>

i

`invenio_query_parser`, 3
`invenio_query_parser.ast`, 3
`invenio_query_parser.parser`, 4
`invenio_query_parser.visitor`, 6

A

accept() (invenio_query_parser.ast.BinaryOp method), 3
 accept() (invenio_query_parser.ast.Leaf method), 4
 accept() (invenio_query_parser.ast.ListOp method), 4
 accept() (invenio_query_parser.ast.UnaryOp method), 4
 And (class in invenio_query_parser.parser), 4
 AndOp (class in invenio_query_parser.ast), 3
 AndQuery (class in invenio_query_parser.parser), 4

B

BinaryKeywordBase (class in invenio_query_parser.ast),
 3
 BinaryOp (class in invenio_query_parser.ast), 3
 BinaryRule (class in invenio_query_parser.parser), 4

D

DoubleQuotedString (class in invenio_query_parser.parser), 4
 DoubleQuotedValue (class in invenio_query_parser.ast),
 4

E

EmptyQuery (class in invenio_query_parser.ast), 4
 EmptyQueryRule (class in invenio_query_parser.parser),
 4

G

grammar (invenio_query_parser.parser.And attribute), 4
 grammar (invenio_query_parser.parser.AndQuery attribute), 4
 grammar (invenio_query_parser.parser.DoubleQuotedString attribute), 4
 grammar (invenio_query_parser.parser.EmptyQueryRule attribute), 5
 grammar (invenio_query_parser.parser.ImplicitAndQuery attribute), 5
 grammar (invenio_query_parser.parser.KeywordQuery attribute), 5
 grammar (invenio_query_parser.parser.KeywordRule attribute), 5

grammar (invenio_query_parser.parser.Main attribute), 5
 grammar (invenio_query_parser.parser.NestableKeyword attribute), 5
 grammar (invenio_query_parser.parser.Not attribute), 5
 grammar (invenio_query_parser.parser.NotQuery attribute), 5
 grammar (invenio_query_parser.parser.Number attribute), 5
 grammar (invenio_query_parser.parser.Or attribute), 5
 grammar (invenio_query_parser.parser.OrQuery attribute), 5
 grammar (invenio_query_parser.parser.ParenthesizedQuery attribute), 5
 grammar (invenio_query_parser.parser.Query attribute), 5
 grammar (invenio_query_parser.parser.RangeOp attribute), 6
 grammar (invenio_query_parser.parser.RangeValue attribute), 6
 grammar (invenio_query_parser.parser.SimpleQuery attribute), 6
 grammar (invenio_query_parser.parser.SimpleRangeValue attribute), 6
 grammar (invenio_query_parser.parser.SimpleValue attribute), 6
 grammar (invenio_query_parser.parser.SimpleValueUnit attribute), 6
 grammar (invenio_query_parser.parser.SingleQuotedString attribute), 6
 grammar (invenio_query_parser.parser.SlashQuotedString attribute), 6
 grammar (invenio_query_parser.parser.Value attribute), 6
 grammar (invenio_query_parser.parser.ValueQuery attribute), 6
 grammar (invenio_query_parser.parser.Whitespace attribute), 6
 GreaterEqualOp (class in invenio_query_parser.ast), 4
 GreaterOp (class in invenio_query_parser.ast), 4

I

ImplicitAndQuery (class in invenio_query_parser.parser),
 5

invenio_query_parser (module), 3
invenio_query_parser.ast (module), 3
invenio_query_parser.parser (module), 4
invenio_query_parser.visitor (module), 6

K

Keyword (class in invenio_query_parser.ast), 4
keyword (invenio_query_parser.ast.BinaryKeywordBase attribute), 3
keyword (invenio_query_parser.ast.NotOp attribute), 4
KeywordOp (class in invenio_query_parser.ast), 4
KeywordQuery (class in invenio_query_parser.parser), 5
KeywordRule (class in invenio_query_parser.parser), 5

L

Leaf (class in invenio_query_parser.ast), 4
LeafRule (class in invenio_query_parser.parser), 5
ListOp (class in invenio_query_parser.ast), 4
ListRule (class in invenio_query_parser.parser), 5
LowerEqualOp (class in invenio_query_parser.ast), 4
LowerOp (class in invenio_query_parser.ast), 4

M

Main (class in invenio_query_parser.parser), 5
make_visitor (class in invenio_query_parser.visitor), 6

N

NestableKeyword (class in invenio_query_parser.parser), 5
Not (class in invenio_query_parser.parser), 5
NotOp (class in invenio_query_parser.ast), 4
NotQuery (class in invenio_query_parser.parser), 5
Number (class in invenio_query_parser.parser), 5

O

Or (class in invenio_query_parser.parser), 5
OrOp (class in invenio_query_parser.ast), 4
OrQuery (class in invenio_query_parser.parser), 5

P

ParenthesizedQuery (class in invenio_query_parser.parser), 5

Q

Query (class in invenio_query_parser.parser), 5

R

RangeOp (class in invenio_query_parser.ast), 4
RangeOp (class in invenio_query_parser.parser), 6
RangeValue (class in invenio_query_parser.parser), 6
RegexValue (class in invenio_query_parser.ast), 4

S

SimpleQuery (class in invenio_query_parser.parser), 6
SimpleRangeValue (class in invenio_query_parser.parser), 6
SimpleValue (class in invenio_query_parser.parser), 6
SimpleValueUnit (class in invenio_query_parser.parser), 6
SingleQuotedString (class in invenio_query_parser.parser), 6
SingleQuotedValue (class in invenio_query_parser.ast), 4
SlashQuotedString (class in invenio_query_parser.parser), 6

U

UnaryOp (class in invenio_query_parser.ast), 4
UnaryRule (class in invenio_query_parser.parser), 6

V

Value (class in invenio_query_parser.ast), 4
Value (class in invenio_query_parser.parser), 6
ValueQuery (class in invenio_query_parser.ast), 4
ValueQuery (class in invenio_query_parser.parser), 6

W

Whitespace (class in invenio_query_parser.parser), 6