QueryParser

A Query Parser is a component responsible for parsing the textual query and convert it into corresponding Lucene Query objects.

There are multiple ways to select which query parser to use for a certain request

- defType - The default type parameter selects which query parser to use by default for the main query. Example: &q=foo bar&defType=lucene
- LocalParams - Inside the main `q` or `fq` parameter you can select query parser using the localParam syntax. Example:

```
&q={!dismax}foo bar
```

See SolrQuerySyntax for in-depth info on the query parser framework, syntax and query syntax.

List of built-in query parsers

Some query parsers are built-in and will work out of the box. They are:

- lucene - The default "lucene" parser
- dismax - DisMax parser allows querying across multiple fields with different weights
- edismax - ExtendedDisMax parser builds on dismax but with more features
- maxscore - Same as "lucene" but returns max() score instead of sum ⚠️ Solr4.4
- func - create a function query
- boost - boost a query by a function query
- frange - functions as range filters, also see this intro blog
- field - simple field query
- prefix - simple prefix query
- raw - create a term query from the input value without any text analysis or transformation whatsoever. This is useful in debugging, or when raw terms are returned from the terms component (this is not the default). also useful for debugging, or for avoiding query parser escaping madness when drilling into facets on text or string fields via fq parameters. Note: use "term" for this in Solr 4.0 or above. Example:

```
&fq={!raw f=field_name}crazy+\"field+value
```

- term - term query, useful for avoiding query parser escaping madness when drilling into facets via fq parameters. Example:

```
&fq={!term f=weight}1.5
```

  ⚠️ Solr4.0 Solr3.2
- surround - SurroundQueryParser provides rich SpanQuery or NEAR support ⚠️ Solr4.0
- simple - Simple query parser is used to parse human readable query syntax. ⚠️ Solr4.7
- complexphrase - ComplexPhraseQueryParser accepts operators inside phrases ⚠️ Solr4.8
- query - nested query parsing M

List of known 3rd party/contrib/unfinished query parsers

These parsers are not built-in, and must be wired into your solrconfig.xml before use.

- synonym_edismax - Third party parser extending ExtendedDismax and providing support for multi word synonyms query time
- xml - Xml query parser (SOLR-839), specify query tree in XML syntax
- json - Json query parser (SOLR-4351), specify query tree as JSON objects
- antlrqueryparser - (LUCENE-5014) Query Parser(s) built on top of the Lucene modern query parser. Grammars are written with ANTLR. Each parser consists of a pipeline of query builders that you can combine at will, besides standard operators, they contain proximity operators (NEAR /x), multi-token synonym expansion, custom modifiers, functional syntax etc.