



.adaptTo(Berlin)

CQ5 QueryBuilder

Alexander Klimetschek | Senior Developer, Day | @alexkli



Agenda

- (1) What it looks like & Use Cases
- (2) Philosophy & Consequences
- (3) Debugging
- (4) Queries & Samples
- (5) Running Queries
- (6) Facets
- (7) Extending

QueryBuilder - What it looks like

- Search for jar files, and order them newest first:

```
type=nt:file  
nodename=*.jar  
orderby=@jcr:content/jcr:lastModified  
orderby.sort=desc
```

- As URL:

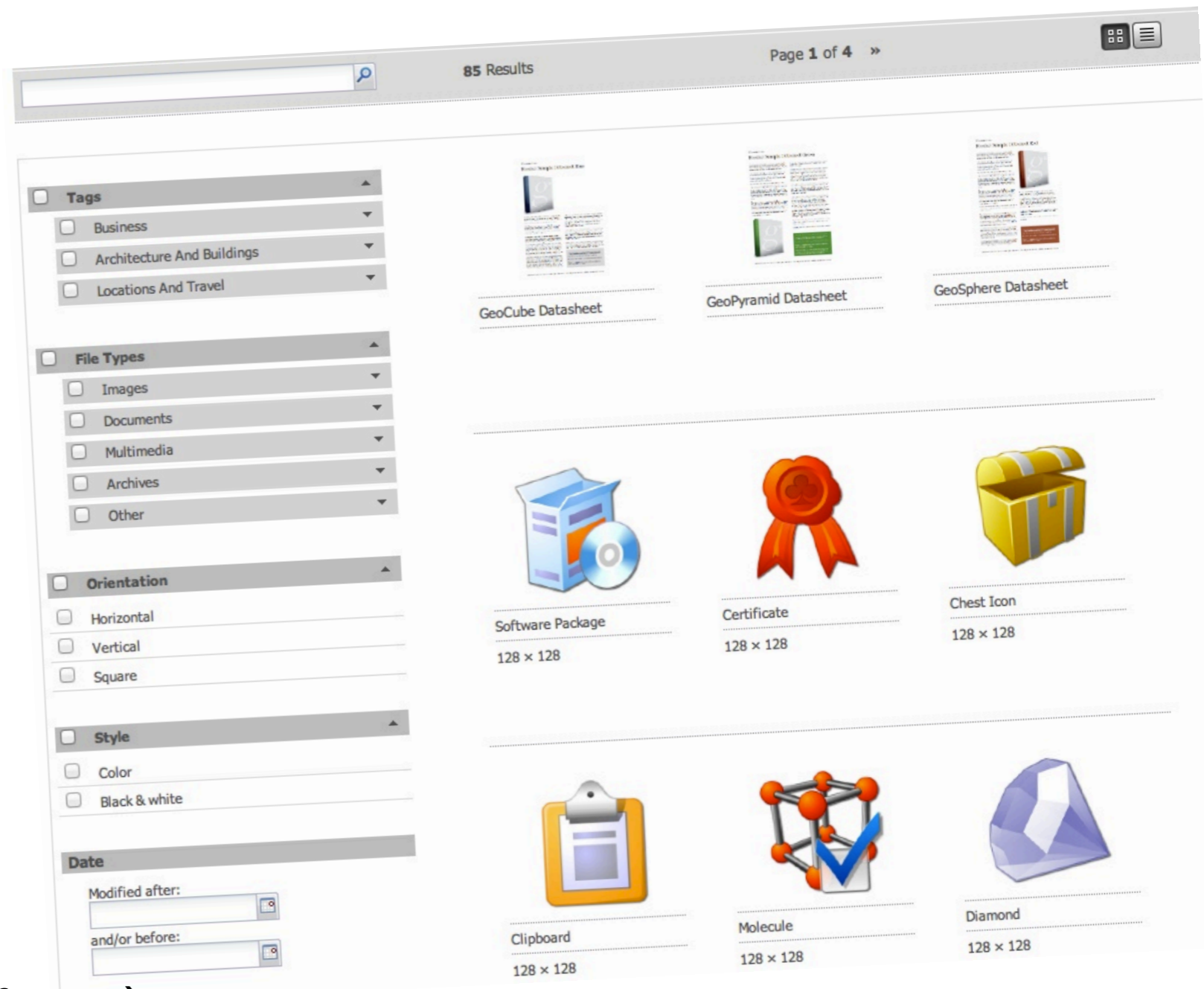
```
http://localhost:4502/bin/querybuilder.json?  
type=nt:file&nodename=*.jar&  
orderby=@jcr:content/jcr:lastModified&orderby.sort=desc
```

- Result as JSON:

```
{  
  success: true,  
  results: 10,  
  total: 155,  
  offset: 0,  
  hits: [{  
    path: "/apps/cloudmgr/install/cq-change-admin-pwd-1.0.1-SNAPSHOT.jar"  
    excerpt: "application/java-archive"  
    name: "cq-change-admin-pwd-1.0.1-SNAPSHOT.jar"  
    ....  
  }, {...}]  
}
```

Use Cases!?

- Advanced query forms
 - DAM Asset share
 - form-based elements
 - „querybuilder UI components“
 - author can add/remove them individually
 - normal form POST or via AJAX
- Content Finder would be cool...
- Search component in CQ (for facets)
- Quick AJAX json query from the client side



- QueryBuilder is...
 - an API to *build queries* for a query engine (JCR XPath underneath)
 - especially via URL query parameters (GET & POSTs)
 - compatible with HTML forms
 - allowing to add/remove conditions („predicates“) individually
 - allowing copy/paste of queries
 - easily extensible
 - providing some goodies (e.g. facets)

- QueryBuilder is **not**...
 - a query engine itself
 - does not have its own search index
 - or even cache

Consequences: URL parameters

- Set of key = value pairs
 - Java: hash maps, property files
- Keep it short for GET requests
 - as fallback, use POST to transport „long“ queries
 - but short queries are more readable
 - ⇒ Avoid duplication in parameter names
 - ⇒ Allow to write custom „shortcut“ predicates (extensible)
- Order must be encoded in parameter names
 - HTML form GETs/POSTs are required to be in order
 - but the Java servlet spec gives you a hash map....
 - ⇒ Conflicts with short names above
 - ⇒ Solution is not very intuitive the first time, but it works



```
http://localhost:4502/bin/querybuilder  
type=nt:file&nodename=*.jar&  
orderBy=@jcr:content/jcr:lastModified
```

More Consequences

- HTML checkbox behaviour
 - checked: **field=on**
 - not checked: *<not sent with the request>*
- Predicates are separate entities
 - though grouping exists
- Copy/paste
 - obvious one, but XPath for JCR does not respect it (limit & offset)
 - *nothing* that can *only* be done through an API call
 - global settings, given as **p.offset=10**
- Extensible
 - predicates evaluators as OSGi components
 - SCR registration to associate with predicate name

QueryBuilder Debugger

- <http://localhost:4502/libs/cq/search/content/querydebug.html>

QueryBuilder Debugger

Search >>>

Extract facets
 Query is given as URL

```
type=nt:file  
nodename=*.jar  
orderby=@jcr:content/jcr:lastModified  
orderby.sort=desc
```

Query tree + URLs

```
ROOT=group: [  
  {nodename=nodename: nodename=*.jar}  
  {orderby=orderby: sort=desc, orderby=@jcr:content/jcr:lastModified}  
  {type=type: type=nt:file}  
]
```

[JSON QueryBuilder Link](#)

[ATOM Feed QueryBuilder Link](#)

```
nodename=*.jar&orderby=%40jcr%3acontent%2fjcr%3alastModified&orderby
```

XPath query

```
//element(*, nt:file)  
order by jcr:content/@jcr:lastModified descending
```

Filtering predicates

```
{nodename=nodename: nodename=*.jar}
```

Results

Number of hits: 155

Time: 16,74 seconds

- [/apps/cloudmgr/install/cq-change-admin-pwd-1.0.1-SNAPSHOT.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.servlets.resolver-2.1.0.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.bgservlets-0.0.1-Rev1063222.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.auth.core-1.0.6.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.commons.scheduler-2.2.0.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.commons.threads-3.1.0.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.event-3.0.2.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.scripting.jsp.taglib-2.1.0.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.jcr.classloader-3.1.4.jar](#) (html, json)
- [/libs/sling/install/org.apache.sling.bundleresource.impl-2.0.6.jar](#) (html, json)

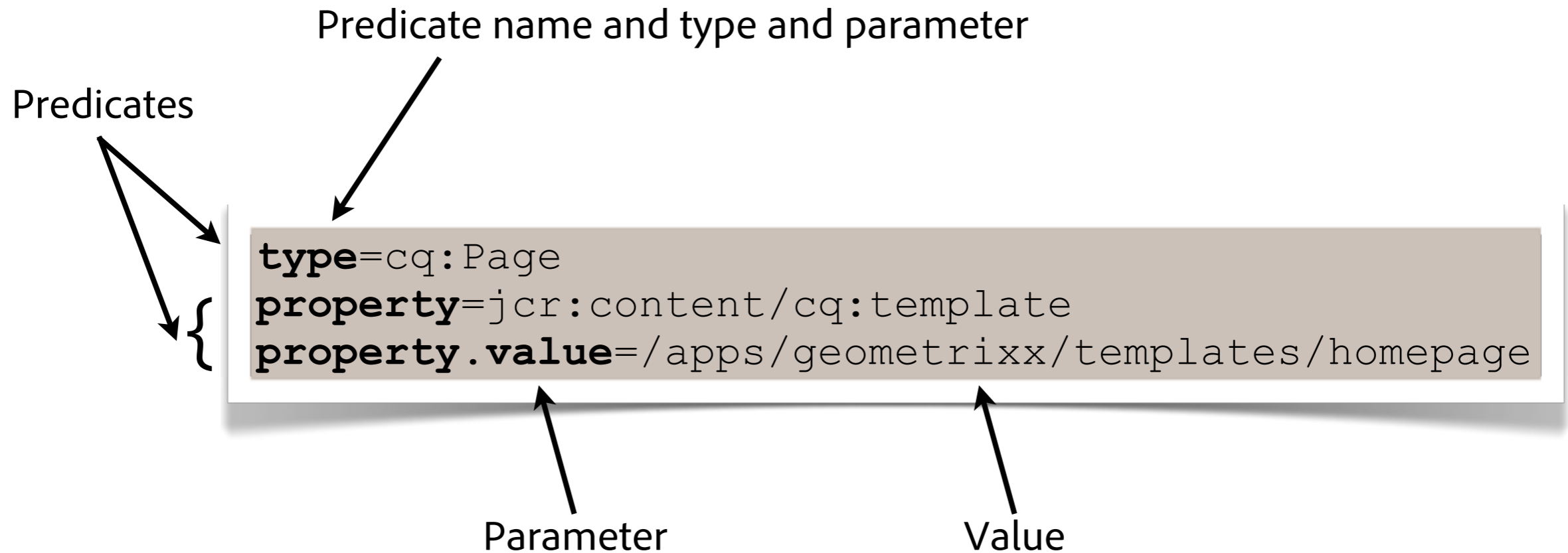
Debug logs

- For all query builder executed queries
- Set „com.day.cq.search“ to INFO or DEBUG or TRACE level
- Standard INFO level (shortened):

```
15.09.2011 19:17:48.566 *INFO* ...QueryImpl executing query (URL):
nodename=*.jar&type=nt%3afile
15.09.2011 19:17:48.566 *INFO* ...QueryImpl executing query (predicate tree):
ROOT=group: [
  {nodename=nodename: nodename=*.jar}
  {type=type: type=nt:file}
]
15.09.2011 19:17:48.567 *INFO* ...QueryImpl xpath query: //element(*, nt:file)
15.09.2011 19:17:48.569 *INFO* ...QueryImpl xpath query took 3 ms
15.09.2011 19:17:48.569 *INFO* ...QueryImpl filtering predicates: {nodename=nodename: nodename=*.jar}
15.09.2011 19:17:48.752 *INFO* ...QueryImpl >> xpath query returned 5098 results (counted)
15.09.2011 19:17:48.753 *INFO* ...QueryImpl filtering took 184 ms
15.09.2011 19:17:48.753 *INFO* ...QueryImpl >> after filtering there are 155 results
15.09.2011 19:17:48.753 *INFO* ...QueryImpl entire query execution took 187 ms
```

- For production, set it to WARN or ERROR ;-)

Anatomy of a query



Predicate's type is mirrored as parameter internally:

```
type.type=cq:Page
property.property=jcr:content/cq:template
property.value=/apps/geometrixx/templates/homepage
```

Predicate resolution & execution

- Internally, a *predicate evaluator* is resolved
- Based on the **type**
- OSGi component (using factories)

- Handles:
 - mapping to xpath (required)
 - filtering of results
 - custom ordering mechanism
 - facet extraction

Multiple predicates of the same type

- Fixed numbering scheme
- Name = **<nr>_<type>**
- Allows to define an order
 - work around hash maps

```
type=cq:Page  
1_property=jcr:content/cq:template  
1_property.value=/apps/geometrixx/templates/homepage  
2_property=jcr:content/jcr:title  
2_property.value=English
```

- No custom names possible!

Standard predicates

- path
 - supports multiple paths
 - but beware: can be slow
- property
 - JCR property
 - different operations
- type
 - node type
- fulltext
 - full text search
- range
- daterange
- similar
 - rep:similar
- tagid & tag
- language
 - page languages
- event
 - calendar
 - example for shortening

Ordering

- Use (special) **orderby** predicate
 - sort ascending by default, use `orderby.desc=true` for descending
- (1) Order by JCR properties
 - `orderby=@cq:tags`
 - `orderby=@jcr:content/cq:tags`
- (2) Reference predicate by name
 - `orderby=1_property`
 - predicate evaluator must provide ordering
 - simply a list of properties (=> used in xpath query)
 - or a custom Comparator (=> run after filtering)
- Multiple orderings
 - `1_orderby=@cq:tags`
 - `2_orderby=@cq:lastModified`
 - `3_orderby=nodename`

Grouping of predicates

- Special group predicate

```
fulltext=Management
group.p.or=true
group.1_path=/content/geometrixx/en
group.2_path=/content/dam/geometrixx
```

- Like brackets:

- (fulltext AND (path=... OR path=...))

- Nested:

```
fulltext=Management
group.p.or=true
group.1_group.path=/content/geometrixx/en
group.1_group.type=cq:Page
group.2_group.path=/content/dam/geometrixx
group.2_group.type=dam:Asset
```

Beware!
Can be slow!

- (ft AND ((path= AND type=) OR (path= AND type=)))

Running queries

- Default servlets
 - JSON: <http://localhost:4502/bin/querybuilder.json>
 - Atom feed: <http://localhost:4502/bin/querybuilder.feed>
 - iCalendar: <http://localhost:4502/bin/querybuilder.ics>
- Java API

```
PredicateGroup root = PredicateGroup.create(request.getParameterMap());
Query query = queryBuilder.createQuery(root, session);

SearchResult result = query.getResult();
for (Hit hit : result.getHits()) {
    String path = hit.getPath();
    // .....
}
```

- JSON: <http://localhost:4502/bin/querybuilder.json>
- **p.hits** selects how the hits are written
 - **simple**
 - path, lastmodified, etc. only
 - **full**
 - full sling json rendering of node
 - by default entire subtree
 - p.nodedepth=1 as in sling's json (0 = infinity)
 - **selective**
 - p.properties is an array of properties to write
 - just the node itself

- From HTTP request:

```
Session session = request.getResourceResolver().adaptTo(Session.class);  
PredicateGroup root = PredicateGroup.create(request.getParameterMap());  
Query query = queryBuilder.createQuery(root, session);
```

- From hash map:

```
Map map = new HashMap();  
map.put("path", "/content");  
map.put("type", "nt:file");  
Query query = builder.createQuery(PredicateGroup.create(map), session);
```

- From predicates:

```
PredicateGroup group = new PredicateGroup();  
group.add(new Predicate("mypath", "path").set("path", "/content"));  
group.add(new Predicate("mytype", "type").set("type", "nt:file"));  
Query query = builder.createQuery(group, session);
```

Persisted Queries

- Store query
 - in Java properties file format
 - in the repository
 - as file node
 - or as string property

```
Query query = .... // create query normally

// store query as file
queryBuilder.storeQuery(query, „/content/myquery“, true, session);

// load it again
Query query = queryBuilder.loadQuery(„/content/myquery“, session);
```

- List component allows this

Facets

- Extract set of possible values found in current result
- Options for a more specific query
- Facet = set of buckets
 - Facet = tag
 - Buckets = product, business, marketing
- Buckets can also be custom ranges
 - Facet = daterange
 - Buckets = yesterday, last week, last year...

```
Map<String, Facet> facets = result.getFacets();
for (String key : facets.keySet()) {
    Facet facet = facets.get(key);
    if (facet.getContainsHit()) {
        writer.key(key).array();
        for (Bucket bucket : facet.getBuckets()) {
        }
    }
}
```


Run new query based on facet/bucket


- Simple as that:

```
String bucketURL =  
query.refine(bucket).getPredicates().toURL();
```

- Facets are extracted for all predicates in the current query
 - keep them „empty“ if they should not search

```
type=cq:Page  
1_property=jcr:content/cq:template  
1_property.value=/apps/geometrixx/templates/homepage  
2_property=jcr:content/jcr:title
```

No value for 2_property



Extending: Writing custom predicate evaluators

```
/** @scr.component metatype="no"
 *      factory="com.day.cq.search.eval.PredicateEvaluator/event" */
public class EventPredicateEvaluator extends AbstractPredicateEvaluator {
    public String getXPathExpression(Predicate p, EvaluationContext context) {
        final String from = p.get(„from“);
        final String to = p.get(„to“);

        // build xpath snippet
        return „@start = ,...‘ and @end = ,...““;
    }

    public String[] getOrderProperties(Predicate predicate, EvaluationContext ctx) {
        return new String[] { „start“ };
    }

    public boolean canFilter(Predicate predicate, EvaluationContext context) {
        return false;
    }

    public boolean canXpath(Predicate predicate, EvaluationContext context) {
        return true;
    }
}
```

Filtering & Facet extraction

- In addition or alternatively to xpath, a predicate can filter
- goes over nodes in result and says include or drop

```
public boolean includes(Predicate p, Row row, EvaluationContext context)
```

- Facet extraction is „lazy“
- Evaluator returns a *FacetExtractor*
- Base implementations available
 - PropertyFacetExtractor
 - DistinctValuesFacetExtractor
 - PredefinedBucketsFacetExtractor

- Basic docs plus some examples:
 - http://wem.help.adobe.com/enterprise/en_US/10-0/wem/dam/customizing_and_extendingcq5dam/query_builder.html
- Javadocs
 - http://wem.help.adobe.com/enterprise/en_US/10-0/wem/javadoc/com/day/cq/search/package-summary.html



Adobe