Navigating Search Results

Chad Campbell
INDEPENDENT SOFTWARE ENGINEER

@chadcampbell   www.ecofic.com

Review

Setup a search service

Run queries

Tailor result rankings

Overview

Sort search results

Paging

Faceted navigation

Query suggestions

Sorting Search Results

Sorting Examples

People Names

first name   last name

Beers

brewery name   beer name

Sorting in Azure Search

Any sortable **field**

`Collection(Edm.String)` **fields are not** sortable

orderby

A query string parameter used to sort results.

Setting up a Sort

By default, results are ordered by descending @search.score value

The higher the @search.score, the higher the ranking

Sorting Direction

Sorts automatically occur in ascending order

Ascending Sort

Rank		
1	Agogwe Ale	2.9
2	Jackalobe Lager	3.3
3	Sigbin Stout	8.1

orderby suffix

desc - Sorts in descending order

asc - Sorts in ascending order

Combining Sorts

Great for getting more specific

Sort by multiple fields at once

Scoring and Sorting

Every result has an `@search.score`

By default, results are sorted by

`@search.score`

Results for "Ale"

Rank	Score	Name	ABV
1	0.28196415	Ahool Ale	5.4
2	0.28196415	Agogwe Ale	3.9
3	0.28196415	Aswang Ale	4.2
4	0.22557132	Megalodon Pale Ale	5.7
5	0.22557132	Pale Popobawa Ale	4.4
6	0.22557132	Snallygaster Pale Ale	9.7

Sort by Score and Fields

Must update or add scoring profile

Cannot just update the `orderby` **parameter**

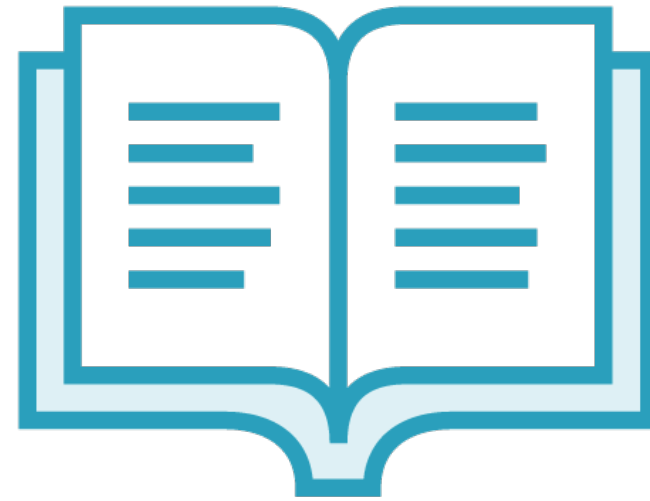`@search.score` **cannot be used in** `orderby`

Summary

Relationship between scores and sorts

Combining sorts

Direction of sorts

Paging Search Results

Navigating with Pages

Summary

Page sizes

Navigate between pages

Setting the Page Size

Tell Azure Search how many results to return

top
Parameter

50 is the default page size

1,000 is the max page size

skip
Parameter

Specifies how many documents to pass over

Summary

Navigating page-to-page

Faceted Navigation

Built by Azure Search

Give you a count

Facets

Adding Facets

Include the `facets` **parameter**

Must use `facetable` **fields**

Creating Ranged Facets

Define buckets with counted results

Creating Ranged Buckets

Creating Ranged Buckets

3.5% 7.0%

< 3.5% > 7.0%

```json
{
  "search":"*",
  "facets":[
    "abv,values:3.5|7"
  ]
}
```

◄ **Defines the list of buckets**
    < 3.5

Supported Types

Edm.Int32

Edm.Int64

Edm.Double

Edm.DateTimeOffset

```
{
  "search":"*",
  "facets":[
    "lastTappedOn,values:2016-01-01T00:00:00Z|2016-03-01T00:00:00Z"
  ]
}
```

Facet on lastTappedOn

Creates ranged buckets on Edm.DateTimeOffset

Creating Intervaled Buckets

Automatically created buckets based on regular intervals

```
{
  "search":"*",
  "facets":[
    "abv,interval:4"
  ]
}
```

◀ Buckets for every four

minute

hour

day

week

month

quarter

year

Interval for
Edm.DateTimeOffset data

Customizing Facet Labels

Display option labels

Sorting facet options

Limiting facet choices

Facet Option Labels

Generated from available data

Formats for values are not available

You're responsible for formatting

You're responsible for converting facet values

Sorting by Facet Values

use the `sort` **option**

Sort Facet Options by Value

```
{
  "search":"*",
  "facets":[
      "breweryName,sort:value"
  ]
}
```

Facet Options in Descending Order

```
{
  "search":"*",
  "facets":[
      "breweryName,sort:-value"
  ]
}
```

Sorting by Facet Result Counts

Sort Facet Options by Count

```
{
  "search":"*",
  "facets":[
      "breweryName,sort:count"
  ]
}
```

Facet Options in Descending Order

```
{
  "search":"*",
  "facets":[
      "breweryName,sort:-count"
  ]
}
```

Use the count

Limiting Search Facets

```
{
  "search":"*",
  "facets":[
      "breweryName,sort:count,count:3"
  ]
}
```

Limit the Facet Options

(to the top 3)

Tradeoffs w/higher count

Search takes longer

More accurate count

What's Important to You?

Highest number of facet options

Faster query

Accurate result count

Using Faceted Navigation

A facet is an entry point into a view of results

Facets must be applied as filters

You must do the work of
building the filter

Facets

Let you build more complex filters

Can improve search performance

`facets` property only supports individual fields

Facets

Navigation pattern

Filter

Highlighting Result Data

highlight
Property

Comma-delimited

Field names considered for hit highlighting

Highlight Parameter

```
{
  "search":"North",
  "highlight":"name"
}
```

Highlight Two Fields

```
{
  "search":"North",
  "highlight":"name,breweryName"
}
```

Highlighting Fields

```json
{
  "search":"North",
  "highlight":"name",
  "highlightPreTag": "<span style='background-color:orange;'>",
  "highlightPostTag": "</span>"
}
```

Search Suggestions with Auto-Complete

Overview

Add suggesters

Get suggestions

Customize suggestion appearance

Adding Suggesters

Part of a search index

Defined by two fields:

name **and** sourceFields

name Property

An identifier

sourceFields
Property

Array of field names

Specifies which fields are examined for suggestions

Requesting Suggestions

Each search index has a docs **endpoint**

Append the suggest **path to get suggestions**

search
Parameter

Informs the search index what you're looking for

Can only be between 1 and 100 characters

```
{
  "search":"Ale*",
  "suggesterName":"beerSuggestions"
}
```

Referencing a Suggester

Using the suggesterName parameter

```json
{
    "search":"north",
    "suggesterName":
        "beerSuggestions",
    "searchFields":
        "name"
}
```

◄ Refines the query

◄ Comma-delimited list of field names

Fuzzy Searches

Provide a more forgiving search experience

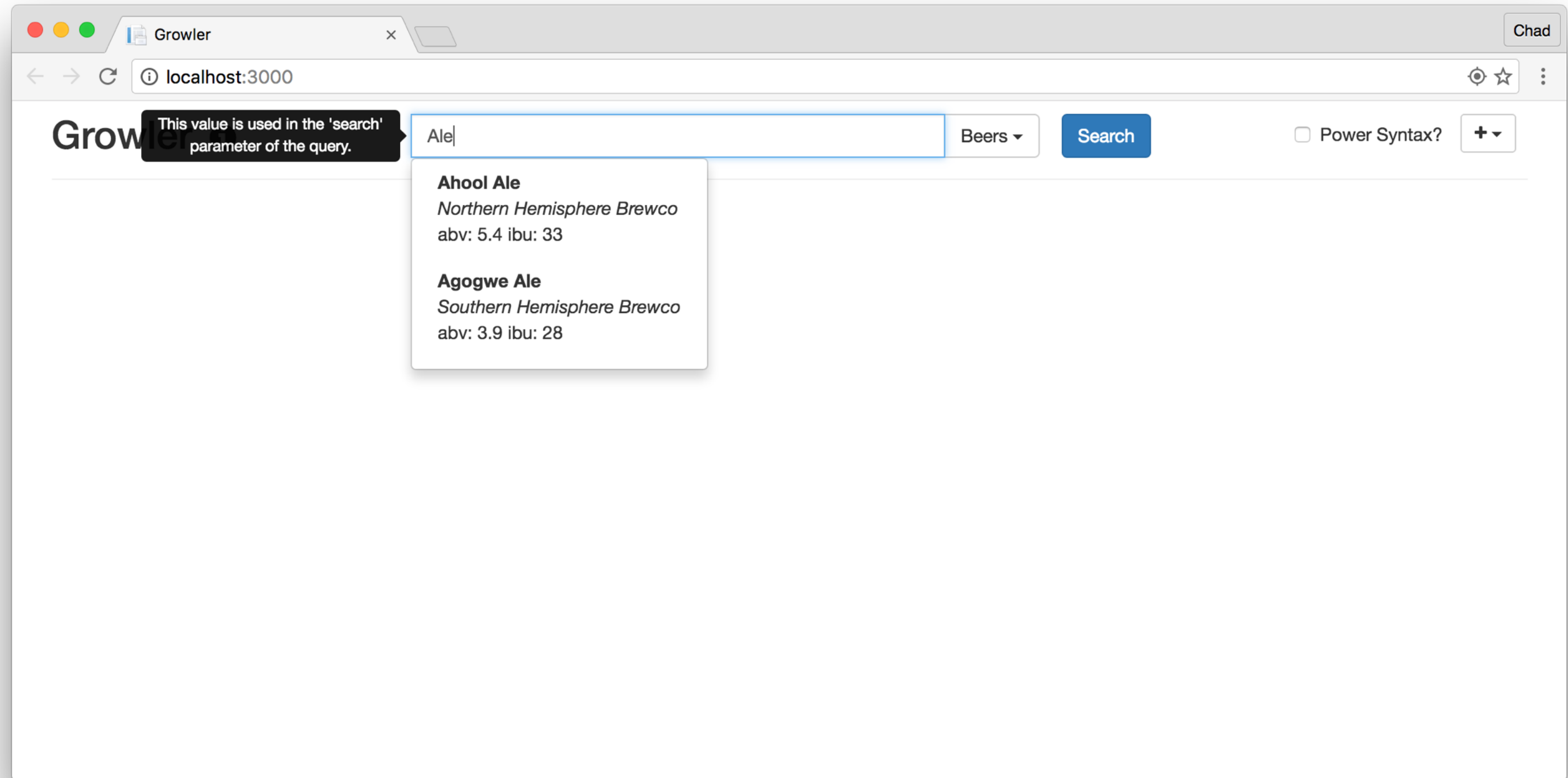Fuzzy searches are not enabled by default

Show additional details

Restrict suggestions

Sort

Highlight

Customizing the
Suggestion List

Selecting Suggestion Fields

```
{
  "search":"north",
  "suggesterName":"beerSuggestions",
  "select":"name, breweryName, abv, ibu"
}
```

Using the `select` Parameter
Returning additional fields

top Parameter

Specifies max number of suggestions to return

Five is the default value

```
{
  "search":"north",
  "suggesterName":"beerSuggestions",
  "select":"name, breweryName, abv, ibu",
  "top":3
}
```

Using the **top** Parameter
To get three suggestions

top Parameter

The first [x] matches are returned

Relevancy is not a factor

Scoring profiles are not used with suggestions

orderby
Parameter

Lets you order by field names

```
{
  "search":"north",
  "suggesterName":"beerSuggestions",
  "select":"name, breweryName, abv, ibu",
  "top":10,
  "orderby":"name"
}
```

Using the orderby Parameter
To order by beer names in alphabetical order

Summary

Providing suggestions to users

Faceted navigation

Pages

Sort search results

User authentication

Authorization

Search indexers

Topics for Consideration

Thank You!

BEST WISHES

Chad Campbell
INDEPENDENT SOFTWARE ENGINEER

@chadcampbell   www.ecofic.com