

Hakutulosten relevanssi

Sisällysluettelo

- [Sisällysluettelo](#)
- [Johdanto](#)
- [VuFind ja Finna](#)
- [Hakutuloslistojen järjestykseen vaikuttavat DisMax-parametrit](#)
 - [Keskeisimmät parametrit](#)
 - [Täsmäntävät parametrit](#)
 - [Esimerkki 1: Korostetaan uusia tietueita](#)
 - [Esimerkki 2: Korostetaan suomenkielisiä tietueita Author-haussa](#)
 - [Esimerkki 3: Korostetaan uusia tietueita melkein aina ja suomenkielisiä tietueita Author-haussa](#)
 - [Esimerkki 4: Korostetaan tietueita, joissa on eniten niteitä \(Alma, Koha, Sierra\)](#)
 - [Linkkejä](#)

Johdanto

Finnassa hakutulokset voidaan järjestää relevanssin perusteella, eli hakukriteereihin nähden osuvimmat tulokset järjestetään tuloksessa ensimmäiseksi. Omassa Finna-näkyvässä organisaatio voi säätää relevanssia tarpeen mukaan.



Relevanssia säätämällä on helppo sotkea hakutulokset. Kannattaa olla varovainen ja testata muutokset huolellisesti.

VuFind ja Finna

Relevanssi määräytyy oletuksena `local/config/finna/searchspecs.yaml`-tiedoston perusteella. Tiedosto on [YAML](#)-muodossa, jossa sisennykset määrittävät tiedoston muodon. Merkistönä on UTF-8 ja sisennykseen käytetään välilyöntejä, ei sarkaimia (tab). Omat muutokset tulee tehdä `local/config/vufind/searchspecs.yaml`-tiedostoon. Siihen ei tarvitse kopioida kaikkea alkuperäisestä `searchspecs.yaml`:sta, muutetut hakukentät riittävät.



Relevanssista saa lisätietoa lisäämällä Finnan hakutulossivun osoitteen perään parametrin `&debugSolrQuery=true` ja tarkastelemalla sen jälkeen sivun lähdekoodia.

Useimmissa hakukentissä on kaksi erilaista prosessointitapaa, `DismaxFields` and `QueryFields`. Tee aina samat muutokset molempiin kohtiin. Finnassa käytetään `eDisMax`-hakusyntaksia aina kun mahdollista, mutta jotkin hakuehdot edellyttävät `Lucene`-hakusyntaksia. Näistä löytyy enemmän selitystä [VuFindin wikistä](#).

Hakukenttien perässä `^`-merkin jälkeen tai hakasuluissa mainittu numero kertoo, kuinka suuren suhteellisen painoarvon kentän sisältö saa hakutuloksessa (weighting). Nämä painoarvot ovat suhteessa vain muihin saman haun kenttiin, eivät mihinkään muuhun. Suurempi luku nostaa kentän tärkeämmäksi kuin pienemmän luvun omaavat kentät, mutta hakuehdon täsmävyvydellä on myös merkitys lopullista painoarvoa laskettaessa.

Joistakin hakukentistä on olemassa tavallinen ja unstemmed-versio (esim. `allfields` ja `allfields_unstemmed`). unstemmed-kentissä ei käytetä kielenkäsittelyä, esim. perusmuotoon palautusta. Niiden painoarvoa kasvattamalla nostetaan täsmälliset osumat tärkeämmiksi kuin epätäsmälliset. Lisäksi osassa hakumäärittämisistä, esim. `AllFields`:ssä, on `ExactSettings`-kohta, jota käytetään, jos hakuehto on ympäröity lainausmerkein.

Hakutulosten pisteytyksessä käytetään [BM25-algoritmia](#).

Korostukset kannattaa lisätä `searchspecs.yaml`-tiedostoon `GlobalExtraParams`-kohtaan. Hakutyypikohtaisesti tehdyissä määrittämisissä on ongelmana se, että ne kertautuvat tarkennetuissa haussa jokaiselle kentälle. `GlobalExtraParams` lisätään hakuun vain kerran riippumatta kenttien määrästä. Alla on esimerkkejä siitä, miten asetuksia lisätään. Tarkempi kuvaus ehdoista ym. löytyy [local/config/finna/searchspecs.yaml.sample](#)-tiedostosta.

Hakutuloslistojen järjestykseen vaikuttavat DisMax-parametrit

DisMaxin [dokumentaatio](#)ssa kerrotaan, että seuraavat DisMax-parametrit vaikuttavan relevanssipisteytykseen (engl. score) eli hakutuloslistan järjestykseen. Lisäksi käytettävissä on [eDisMaxin lisäparametrit](#).

Keskeisimmät parametrit

pf

Parametrilla `pf` voidaan nostaa dokumenttien pisteitä tapauksissa, joissa kaikki hakusanat ovat dokumentissa lähellä toisiaan. Normaalisti DisMax laskee jokaiselle hakusalle omat relevanssipisteensä. Jos parametri `pf` on annettu, DisMax antaa lisäksi sen mukaisen painotuksen hakusanojen muodostamalle fraasille.

Muotoilu on lista kenttä- ja nostokerroinpareja. Näitä käytetään, kun luodaan kyselyä parametrin `q` (hakulauseke) perusteella.

Esim. Korostetaan tietueita, joissa on otsikossa sanat juuri halutussa järjestyksessä:

```
- param: 'pf'
  value: 'title_txtP^1000 author^500'
```

tie

Oletusarvoisesti relevanssinlaskennassa käytetään suurimmat pisteet saaneen kentän pisteitä.

Kun tie-parametri on määritelty, muidenkin osumakenttien pisteitä lisätään suurimmat pisteet saaneen kentän pisteisiin.

bq

bq on SolrQuerySyntax-muotoinen merkkijono, joka sisällytetään käyttäjän kyselyyn ja joka vaikuttaa relevanssipisteiden laskentaan. bq:ssa annetut ehdot eivät rajaa hakutulosta, vaan vaikuttavat ainoastaan sen järjestykseen.

Esim. Korostetaan suomenkielisiä tietueita:

```
- param: "bq"
  value: "(language:fin)^0.11"
```

bf

bf-parametrin avulla voi sisällyttää käyttäjän kyselyyn [funktioita](#). Funktio voi olla mikä tahansa Solrin natiivifunktio, ja kullekin on mahdollista antaa nostokerroin. bf-parametrin funktiot sisällytetään käyttäjän kyselyyn ja vaikuttavat relevanssipisteisiin.

Esim. Korostetaan uusia tietueita:

```
- param: "bf"
  value: "recip(ms(NOW/DAY,main_date),1,1000,1000)^20000"
```

Täsmentävät parametrit

ps

Parametri ps täsmentää parametria pf. Parametri ps kertoo, montako "positiota", eli käytännössä sanaa, merkkijonon osaa saisi enintään siirtää, jotta hakutulokseen sisältyvä merkkijono olisi täsmälleen sama kuin käyttäjän syöttämät hakusanat (vrt. "fraasihaku"). Lisätietoja löytyy [Solr-wikistä](#). Kirjain s on tässä lyhenne englanninkielisestä sanasta slop, jolla tarkoitetaan tämän siirron suuruutta.

Esim. Korostetaan hakutuloksia, joissa sanojen etäisyys toisistaan on enintään 2 positiota:

```
- param: "ps"
  value: "2"
```

Esimerkki 1: Korostetaan uusia tietueita

Tämä esimerkki korostaa uusia tietueita AllFields- ja Title-hauissa, kunhan DismaxParams-asetuksella ei ole tehty muita korostuksia ko. hakutyypin kohdalla.

```
GlobalExtraParams:
- param: bf
  value: "recip(ms(NOW/DAY,main_date),1,1000,1000)^20000"
conditions:
- SearchTypeIn:
  - AllFields
  - Titles
- NoDismaxParams:
  - bf
  - bq
```

Esimerkki 2: Korostetaan suomenkielisiä tietueita Author-haussa

Tämä esimerkki korostaa suomenkielisiä tietueita Author-haussa.

```
GlobalExtraParams:
- param: bq
  value: "(language:fin)^0.11"
conditions:
- SearchTypeIn:
  - Author
```

Esimerkki 3: Korostetaan uusia tietueita melkein aina ja suomenkielisiä tietueita Author-haussa

Tämä esimerkki korostaa uusia tietueita kaikissa muissa paitsi year-haussa ja suomenkielisiä tietueita Author-haussa.

```
GlobalExtraParams:
- param: bf
  value: "recip(ms(NOW/DAY,main_date),1,1000,1000)^20000"
  conditions:
    - SearchTypeNotIn:
      - year
- param: bq
  value: "(language:fin)^0.11"
  conditions:
    - SearchTypeIn:
      - Author
```

Esimerkki 4: Korostetaan tietueita, joissa on eniten niteitä (Alma, Koha, Sierra)

Alma-, Koha- ja Sierra-kirjastoissa indeksoidaan niteiden lukumäärä kenttään `items_int` ja tilausten lukumäärä (vain Koha ja Sierra) kenttään `orders_int`, jolloin näitä voidaan käyttää lukumäärän perusteella suosituimmiksi katsottujen tietueiden korostamiseen.

```
GlobalExtraParams:
- param: bf
  value: "items_int^20"
```

Linkkejä

[DisMax-ohje Solr-wikissä](#)

[eDisMax-ohje Solr-wikissä](#)

[DisMax-opas Solr:n Confluencessa](#)

[Solr Relevancy FAQ](#)

[BM25 The Next Generation of Lucene Relevance](#)