POLARIS
INTEGRATED LIBRARY SYSTEM

Relevancy Ranking in Polaris® PowerPAC™ FAQ

Polaris Library Systems
Box 4903
Syracuse, New York 13221-4903
www.polarislibrary.com

Send any comments or questions about this document to your Site Manager or to the Technical Communications Group: **TechComm@polarislibrary.com**.

Trademarks Polaris® is a registered trademark of GIS Information Systems, Inc., dba Polaris Library Systems.

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

Disclaimer The information contained in this document is subject to change without notice. Polaris Library Systems shall not be liable for technical or editorial omissions or mistakes in this document nor shall it be liable for incidental or consequential damages resulting from your use of the information contained in this document.

Printed in the
United States of America
January 08, 2010

This document is written for Polaris 3.6 or later.
Master Number Rev 1

Relevancy Ranking in Polaris PowerPAC

Relevancy ranking is the ordering of search results so that those that are most likely to meet the user's needs, as based on the user's search terms, appear first in the list. Most popular search engines rank search results by relevance, but the methods used to calculate relevance vary among search engines. This document summarizes relevancy ranking in Polaris PowerPAC version 3.6 or later.

How does it work?

Polaris PAC keyword and phrase searches use two levels of relevancy ranking logic.

The first level of relevancy ranking examines up to 100,000 records. The process uses the standard "term frequency - inverse document frequency" logic, but also includes structured weighting based on the field/subfield in which the search term was found. Tag weighting puts higher weights on the primary MARC fields (for example, 1xx, 245, 6xx) and lower weights on the secondary fields (for example, 767, 780). Most tags bear different weights for different particular subfields within that tag. Primary title and author fields are weighted higher than all other types of fields. There are four levels of weights for authorities and six levels of weights for bibliographic records.

Through a series of calculations, each instance of a match in a record is weighted and a record's relevancy rank is based on the cumulative total "record weight."A term that appears only once in a primary field gets a high weight, tending to move the record to the top of the list of results. A term that appears often in secondary fields gets a medium-to-high record weight, even if the term does not appear in a primary field. An additional magnitude calculation, which measures the size of the field containing the keyword and assigns a weight to the result, ensures that search terms such as **It** or **24 Hours** return the titles *It* and *24 Hours* as top-ranked results.

Relevance is the default sort option for PAC keyword and phrase searches. If the user selects a different sort option, this first level of relevancy ranking is automatically part of processing the search results before the selected sort option reorders the results. When the sort option is Relevance, however, the search goes to a second level of processing before results are displayed. This secondary level includes a calculation based on the proximity of search terms in the record. The library can change the default sort order setting in Polaris Administration, and the user can change that setting for any particular search.

Why do DVDs come before books? Shouldn't books come first?

Relevancy ranking is based strictly on keyword statistics. For example, type of material is not considered. A DVD may be ranked higher than a book because it is cataloged more thoroughly, with more subject headings and alternative titles containing the keywords than a book—but not because it is a DVD. As another example, for the same statistical reasons, you may see that books about Harry Potter come before the books in the actual Harry Potter series.

What's the difference between relevance and popularity?

Polaris offers both relevancy and popularity ranking, but they are very different sorting methods. Polaris relevancy algorithms capture and order titles that are likely to satisfy a searcher's query as submitted, without explicitly trying to second-guess the searcher's intent (such as "show me what other people are looking for").

Popularity ranking is based on circulation. Polaris offers a Most Popular sort option, which orders titles in the search results so that in effect, the titles most frequently checked out or requested over the past 120 days, for example, appear first in the list. The Most Popular option can be set as the default sort option for Polaris PowerPAC, as can any other available sort option. (First-pass relevancy ranking as described earlier in this document is applied regardless of the sort option selected.)

In Most Popular sorting, the program counts check-out and hold transactions by bibliographic record in the specified date range and sorts them in this order:

1) Total checkouts and hold requests in the specified date range, descending
2) Checkouts in date range, descending
3) Lifetime circulation transactions, descending
4) Lifetime in-house circulation transactions, descending
5) Bibliographic record creation date, descending

Polaris also offers a "Most Circulated" Web part, the elements of which are links that launch searches for popular titles, authors, or subjects in the Polaris database, based on circulation statistics for the past 31 days. A scheduled SQL job updates the list according to the job schedule.

What is the relevancy ranking algorithm?

Polaris PowerPAC relevancy ranking is an expanded span-detection algorithm based on the following assumptions:

- The relevance of a record should increase in proportion to the frequency of query terms matched in a given field.
- The relevance of a record should increase in proportion to the proximity of unique query terms matched in a given field.
- The relevance of a record should increase in proportion to the weight of a field.
- The relevance of a record should increase in inverse proportion to the size of the field compared to the query term.

For more details about this type of algorithm, see "Viewing Term Proximity from a Different Perspective," Ruihua Song, Ji-Rong Wen, and Wei-Ying Ma, Microsoft Research Asia.
http://research.microsoft.com/apps/pubs/default.aspx?id=70178