

Searching through the text-based homeopathic repertory

Ing. Peter Bezemek

Affiliations of the author: AEON GROUP Ltd. co-owner and director
Correspondence and reprints: AEON GROUP Ltd., Tr. SNP 39, 974 01 Banska Bystrica,
Slovakia; email < support@mercurius.sk >; web: < <http://www.mercurius.sk> >

BANSKA BYSTRICA, SLOVAKIA, May 2005

Abstract

The document describes the effective searching techniques for the text-based homeopathic repertory¹, as implemented in Mercurius homeopathic software.

Introduction

As our homeopathic repertories grow bigger and bigger, the need for effective searching of this plethora of information is paramount.

Unfortunately, it seems that a majority of homeopaths relies solely on their (supposed) knowledge of the structure of traditional repertories, such as Kent's.

As a result, if the structure changes (as happened to Repertorium Universale III or Synthesis 9.1), they are often lost and therefore reluctant to use the most up-to-date material.

Even worse, such a behaviour is supported by some poorly designed homeopathic softwares, which function primarily as a computer representation of what is presented in the paper version of the repertories, with almost identical graphical representation and essentially no added benefit of using the computer, in terms of locating the right rubrics/symptoms.

This brings no progress to homeopathy.

The ability to find the representation of patient's symptoms in the repertory in a short time is essential and should be therefore made as simple and as effective as possible. The document describes the effective searching techniques as implemented in Mercurius homeopathic software.

¹ Currently, all available homeopathic repertories have fixed structure and are text-based. The repertory of the future should have web-like structure and be object based, to create a semantic repertory which will dispose the need for using both repertory and materia medica.

Basic search

Since we use text-based repertories, we need to search for a text.
Each rubric of the repertory can be divided to words which can be searched for.

In Mercurius, all you need to do to find a symptom is typing the **words or parts of the words** you think should be present in the rubric, in any order.

Full Partial matching

To search for a pain in the temples that moves from left to the right temple, we could type the search query

*Query 1: **temple left right pain***

The query executes as searching for rubrics containing the characters
"*temple*" AND "*left*" AND "*right*" AND "*pain*" (according to star convention¹)

Some of the results triggered by the query:

*Head, phenomena, **pain**, drawing, forehead, sides, **right**, **left** and to **left temple**, to
Head, phenomena, **pain**, **temples**, **left**, **right**, to
Head, phenomena, **pain**, bursting, **temples**, **left**, **right**, then
Head, phenomena, **pain**, neuralgic, **temple**, **left**, **right**, to*

As it is clear from the results, it is important to be able to search not only for whole words, but also parts of the words, otherwise the keyword "temple" would not trigger the results that contain the word "temples".

This is a very important feature, useful for finding the words about which we are unsure in what form they are used in the rubric/symptom. It is especially useful in some non-English languages such as German or Slavonic languages which use a lot prepositions and different word forms, making it very difficult to find anything using plain whole-word matching.

Full partial matching avoids this problem and allows the user to find the matching rubrics without the need to think about all the possible word forms.

¹ A mask for representing the names of multiple entries with asterisks. Asterisk (*) means any characters i.e. "*head*" matches the words "fore**head**" or "**head**ache".

One-sided partial matching or whole-word matching

Sometimes, full partial matching may trigger undesired results, such as when searching for "head" finds rubrics with "forehead".

This can be avoided by using one-sided partial matching or whole-word matching. Such behaviour is set by using the capital letters at the beginning or the end of the keyword.

Rules:

1. Left-sided partial matching - > If the keyword starts with a capital letter, the matching word starts with the keyword. ("Head" triggers "head" or "headache", but not "forehead")
2. Right-sided partial matching - > If the keyword ends with a capital letter, the matching word ends with the keyword. ("heaD" triggers "head", "forehead", but not "headache")
3. Whole-word matching - > If the keyword starts and ends with a capital letter, the matching word must match the keyword exactly. ("HeaD" or "HeAD" or "HEAD", all trigger "head" and nothing else)

For English language, the most suitable search seems to be left-sided partial matching.

Negative matching

To specify a negative match, we can use the minus “-” sign to filter out certain keywords we do not wish to be included in the results.

To look for rubrics related to hay fever occurring in spring or summer, but not in autumn or winter, we can specify the following query:

*Query 2: “**hay fever -autumn -winter**”*

Some of the results triggered by the query:

*Nose, phenomena, coryza, general, annual, **hay fever***
*Nose, phenomena, coryza, general, annual, **hay fever**, spring*
*Nose, phenomena, coryza, general, annual, **hay fever**, summer*
*Nose, phenomena, coryza, general, annual, **hay fever**, summer, diarrhea, in*
*Nose, phenomena, coryza, general, annual, **hay fever**, august, in*

OR matching

It is sometimes useful to be able to use OR function to specify.

In Mercurius, OR matching is specified by plus “+” sign.

To look for the previous example using the OR matching, we could write.

*Query 3: “**hay fever spring + hay fever summer**”*

Some of the results triggered by the query:

*Nose, phenomena, coryza, general, annual, **hay fever, spring***
*Nose, phenomena, coryza, general, annual, **hay fever, summer***
*Nose, phenomena, coryza, general, annual, **hay fever, summer**, diarrhea, in*

Synonymic matching

The support of synonyms is vital for achieving the best results. English expressions have lots of synonyms and many of them are used in the repertory.

By using synonymic matching the number of results can rise even 20 times!

Mercurius implements the best-in-the-industry support of synonyms.

All synonyms are contained in the file synonyms.cfg that can be edited by the user in any text editor.

Without synonymic support, the search for "**Rude**" triggers 10 records in Repertorium Universale III:

*Generalities, modalities, **rudeness** of others agg.*

*Head, phenomena, pain, talking, agg., **rude***

*Mind, phenomena, **rudeness***

*Mind, phenomena, **rudeness**, children, of naughty*

*Mind, phenomena, **rudeness**, diarrhea, in*

*Mind, phenomena, **rudeness**, employees to the chiefs, of*

*Mind, phenomena, **rudeness**, fever, during*

*Mind, phenomena, **rudeness**, women, in*

*Mind, phenomena, sadness, despondency, depression, melancholy, **rudeness** of others, from*

*Mind, phenomena, sensitive, oversensitive, general, **rudeness**, to*

On the other hand, the same search with synonymic matching triggers 238 symptoms!

Some of the results triggered by the query:

*Mind, modalities, **insults**, offenses, ailments from*

*Mind, phenomena, **abrupt***

*Mind, phenomena, **abusive, insulting***

*Mind, phenomena, **quarrelsomeness**, scolding*

*Mind, phenomena, **cursing, swearing**, desires*

*Mind, phenomena, **reproaches, others***

*Mind, phenomena, lewdness, **obscene***

*Mind, phenomena, **mocking**, general, friends, at his*

Multiple-words synonymic matching

Mercurius is the only known homeopathic software in the world that implements multiple words synonymic matching.

This feature is extremely useful for triggering synonyms based on a combination of keywords.

To look for a headache with the perspiration of hands you would type:

Query 4: "**head pain perspir hand**"

With synonymic matching, this would reveal the symptom:

Extremities, phenomena, perspiration, general, upper limbs, hands, migraine, in

As it is clear from the result, "head pain" combination propagated to "migraine" and discovered a symptom you would normally hardly find.

When the query is executed, the algorithm performs the following search:

"head pain perspir hand + headache perspir hand + migrai perspir hand"

The algorithm is able to map the one word or a multiple combination of words to one word or a multiple combination of words.

Thus "headache" triggers also "head pain".

"strong will" triggers both "Mind Stubborn" and "headstrong".

By using these advantages, the following feats are also possible:

To look for a headache that gets worse outside, you just type:

Query 5: "**headache outside worse**"

which triggers the right results, such as:

Head, phenomena, **pain**, **air**, **open**, **agg**.

Head, phenomena, **pain**, tearing, temples, **air**, **open**, **agg**.

Head, phenomena, **pain**, vertex, **air**, **open**, **agg**.

Internally, the algorithm executes a complicated query which covers the combination of all the defined synonyms.

Format of the synonyms.cfg

Synonymic engine loads the expressions from the file synonyms.cfg, which can be edited by the user.

The synonymic combinations are listed line by line and are divided by semicolon.

The first expression (called the trigger expression) is a word or a combination of words that triggers the rest of the expressions (called bullet expressions).

quick*;Fast -FASTING;Speed;Agil;Acceler;Rapid;Swift;Brisk;Hast;Vehement

In the example above "quick*" is the trigger expression, while the rest are bullet expressions.

The trigger expression uses a star convention, so to trigger both "hyperventilation" and "hyperventilate", it is not enough to specify

Incorrect: **hypervent**;Respira Acceler;Quick Breath;Fast Breath

but it should be

Correct: **hypervent***;Respira Acceler;Quick Breath;Fast Breath

The bullet expressions behave in accordance with the standard search used in Mercurius. Note that the trigger expression triggers the bullet expressions, but **not the other way round**.

For "respiration accelerated" to trigger "hyperventilation", it must be specified, for example:

respir* acceler*;Hyperventila

This behaviour is very suitable for situations when you would like "eruption" to trigger all sorts of eruptions, but "herpes" to trigger only herpes.

eruption*;BOIL;BOILS;Carbuncle;Furuncle;Vesicle;Impetigo;Herpes;Blister;Pimple;Pustul;Papul;Nodul;Comedone;Acne;Condylom;SCALES;SCALY;Exanthem;Crust;Tubercle;Dandruf;Rash;Nettlerash;Urticar;Eczema;Blotch;Molluscum;Miliar;Wart;Scurf;Morphea;Enanthem;Dermatitis;Caruncle