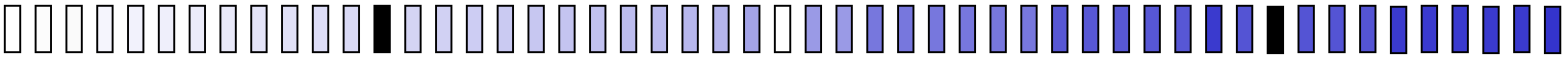
Lexical Tools

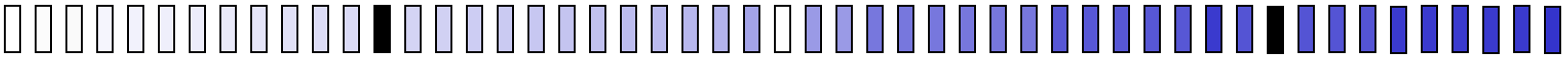
The Lexical Tools

- Introduction
- Norm
- WordInd
- Lvg
- Installation
- Embedding This into Your Application
- Using The Lexical Tools with The Metathesaurus
- Building an Index Using The Lexical Tools
- Changes for 2003
- Additional Tools Developed by NLM



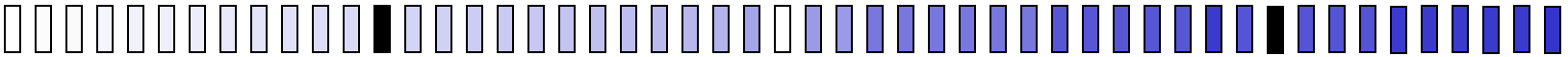
Lexical Tools: Introduction

- Command line tools
  - norm
  - lvg
  - wordInd
- Web GUI, Lexical Gui Tool (lgt)
- Embeddable Java API's



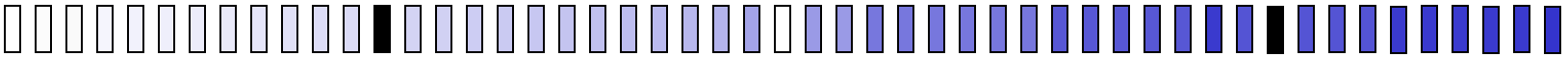
Lexical Tools: Introduction

- These tools are good for
  - aggressive text pattern matching
  - making word, term, phrase indexes
  - matching queries with indexed entries
  - increasing recall and/or precision

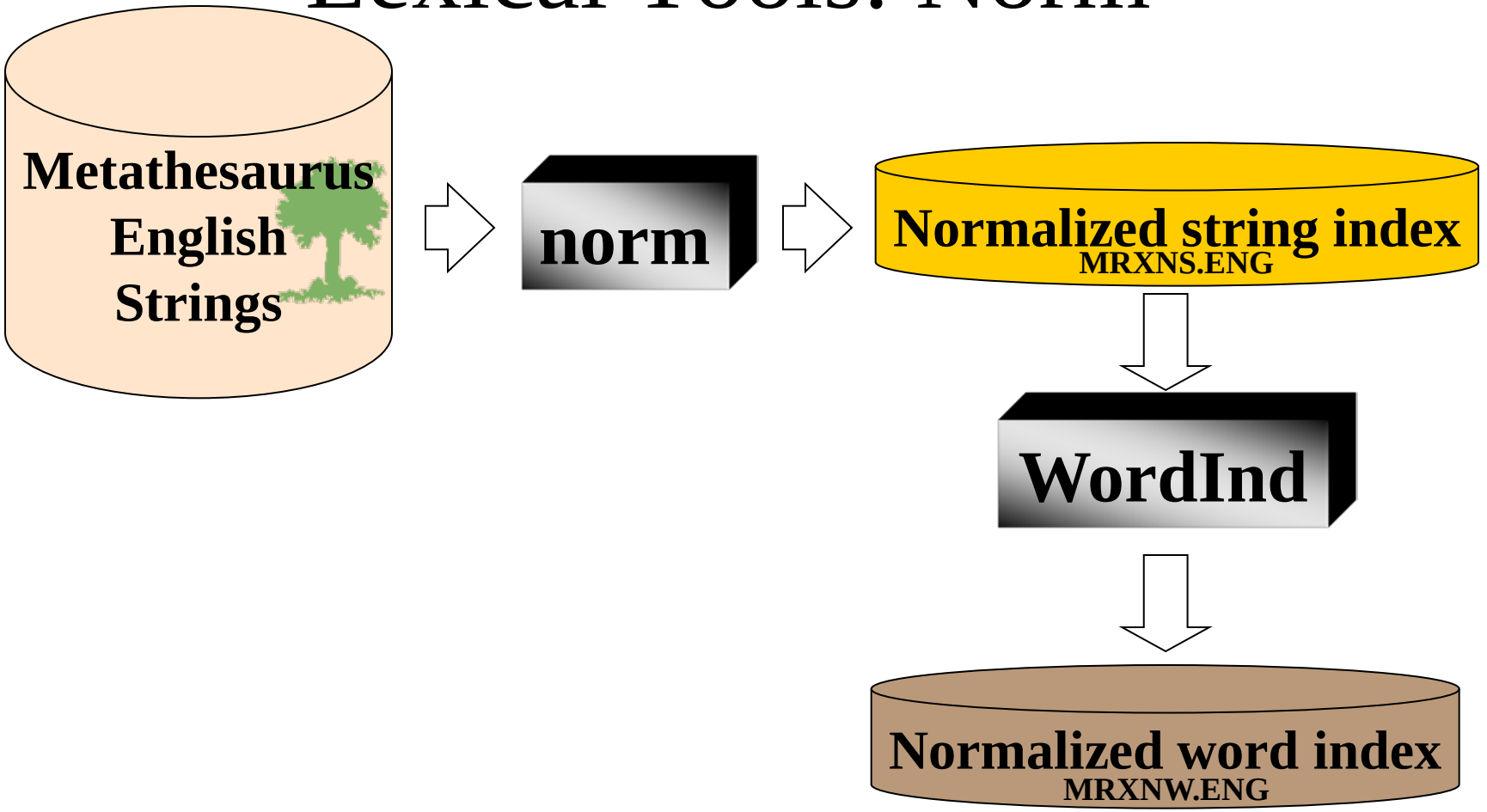


Lexical Tools: Introduction

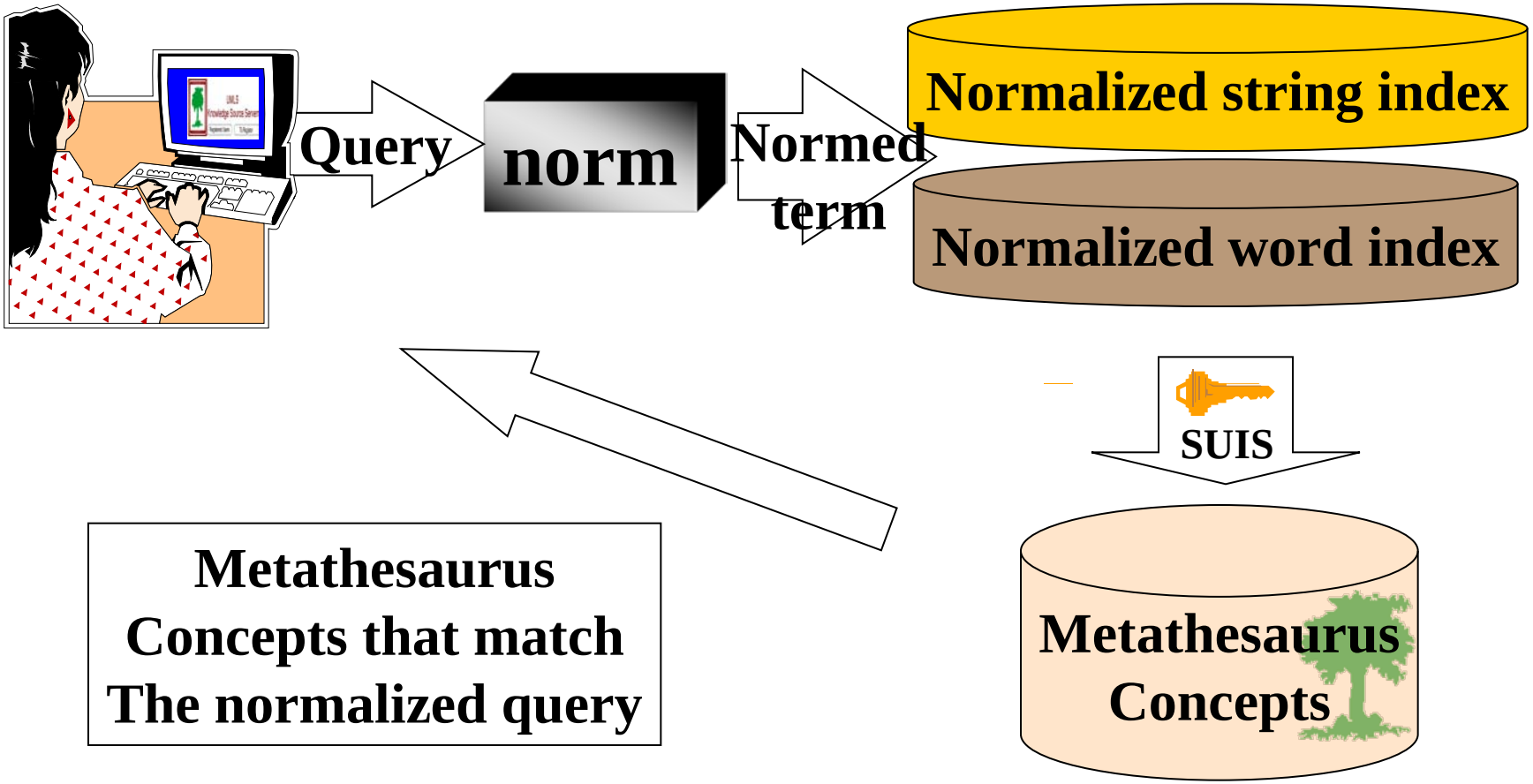
- Characteristics of all the command line tools
  - take input from the screen or a file
  - put their results to the screen or a file
  - Interpret fielded text
    - Can be told which fields contain what type of information



Lexical Tools: Norm

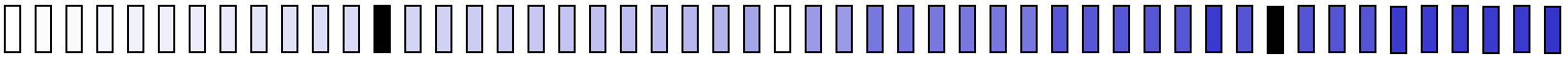


Lexical Tools: Norm



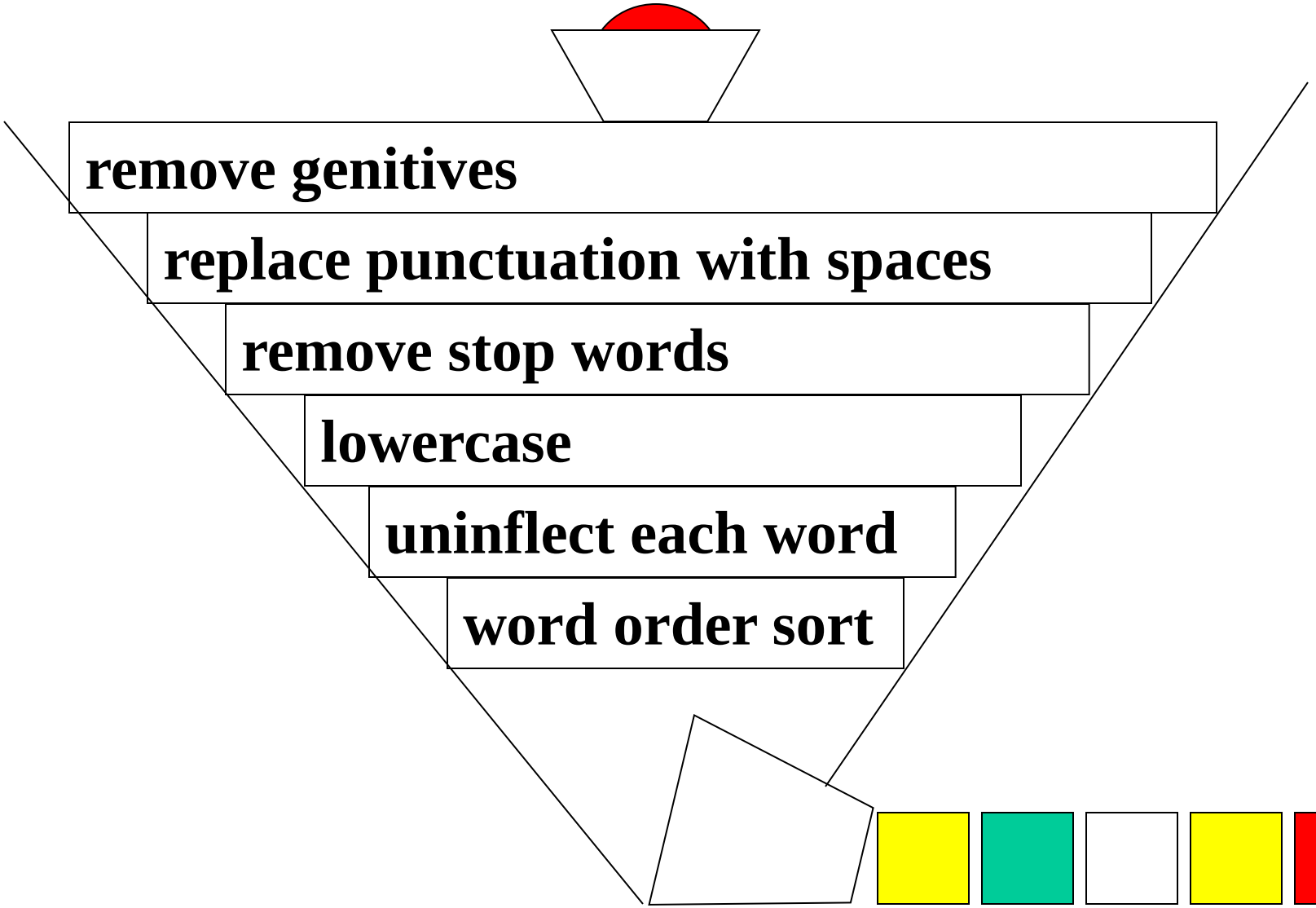
Lexical Tools: Norm

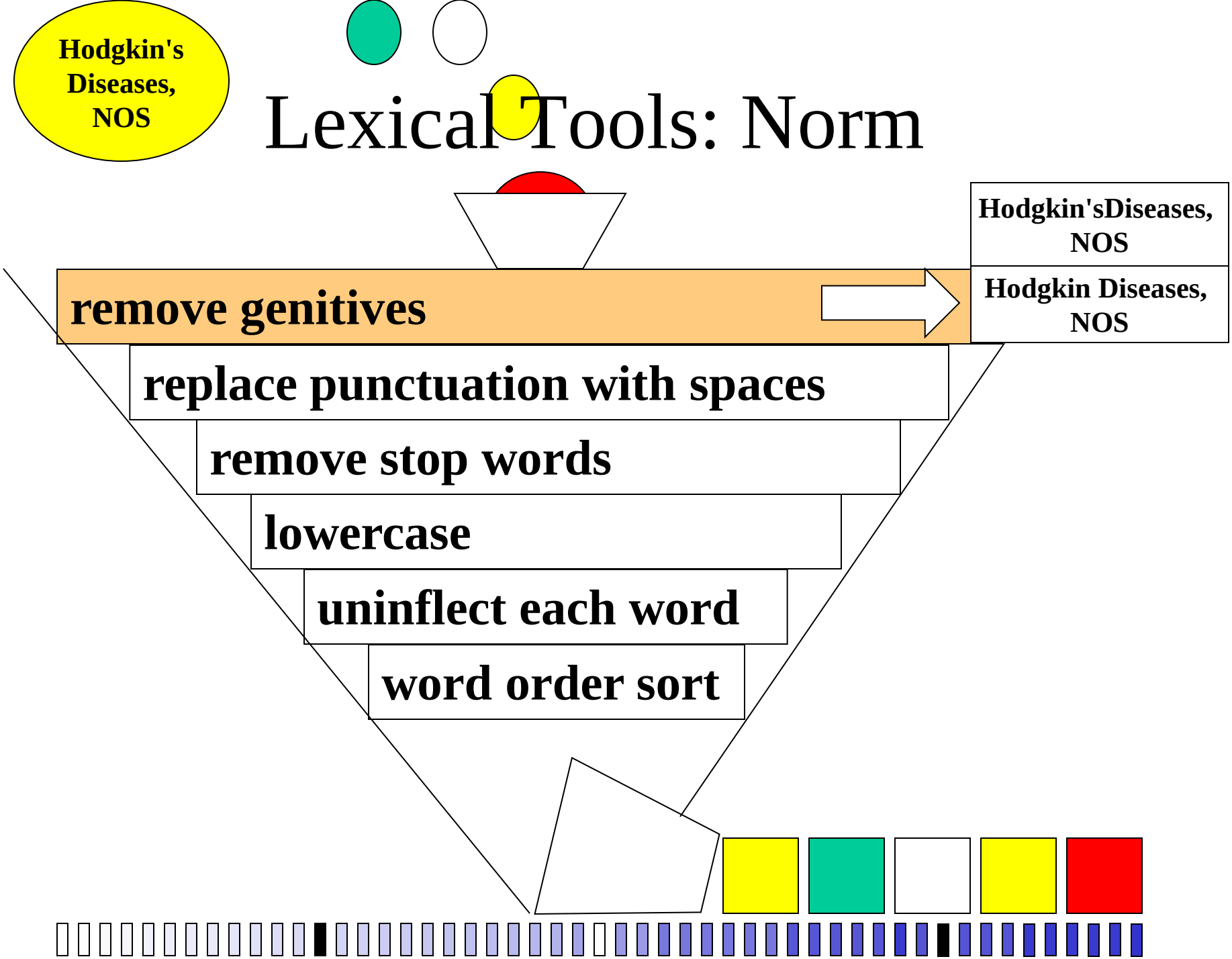
- Norm abstracts away from:
  - case
  - punctuation
  - word order
  - possessive forms
  - inflectional variation



Hodgkin's Diseases, NOS

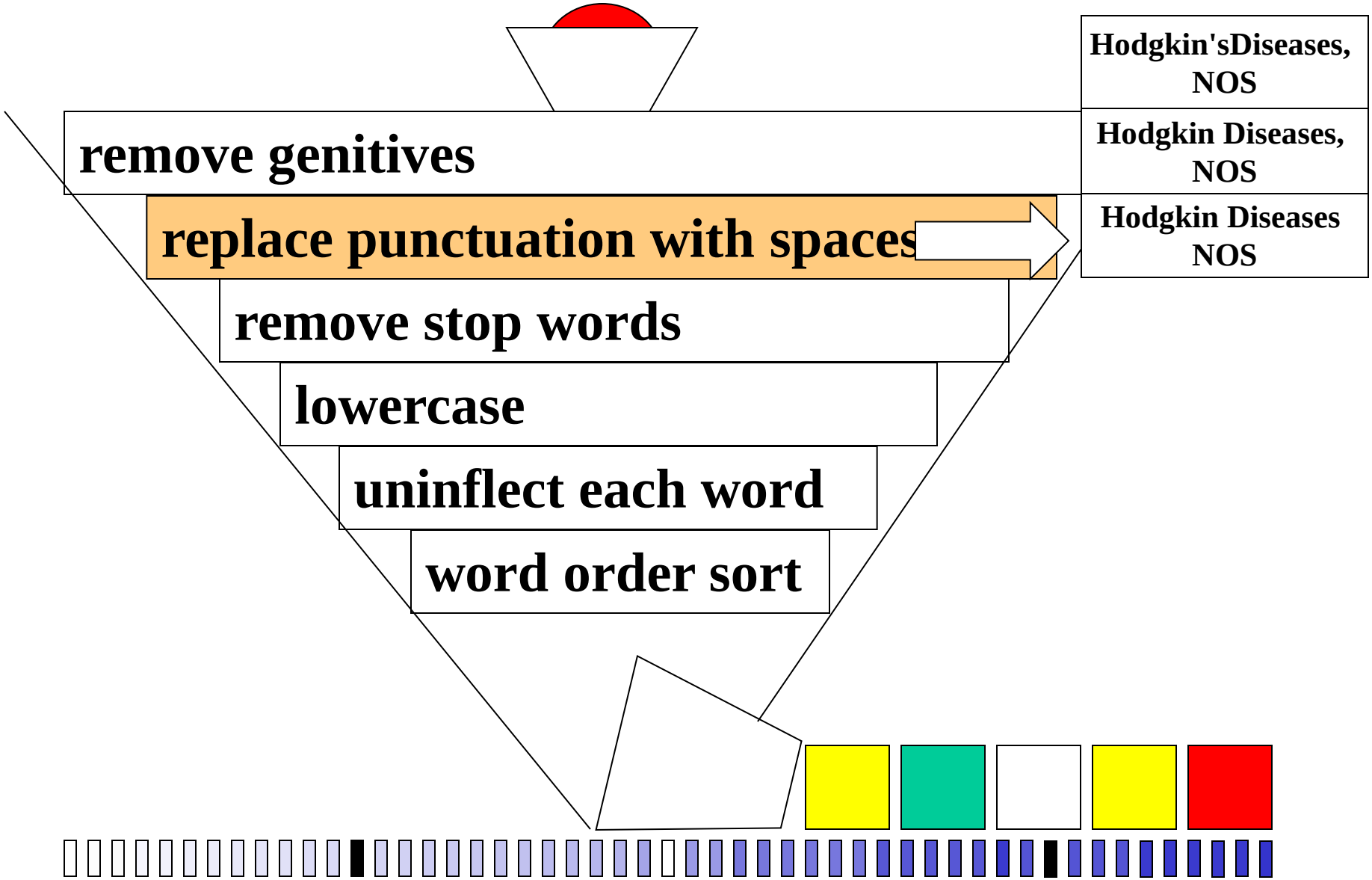
Lexical Tools: Norm




Hodgkin's Diseases, NOS
Lexical Tools: Norm
Hodgkin'sDiseases, NOS
remove genitives
Hodgkin Diseases, NOS
replace punctuation with spaces
remove stop words
lowercase
uninflect each word
word order sort

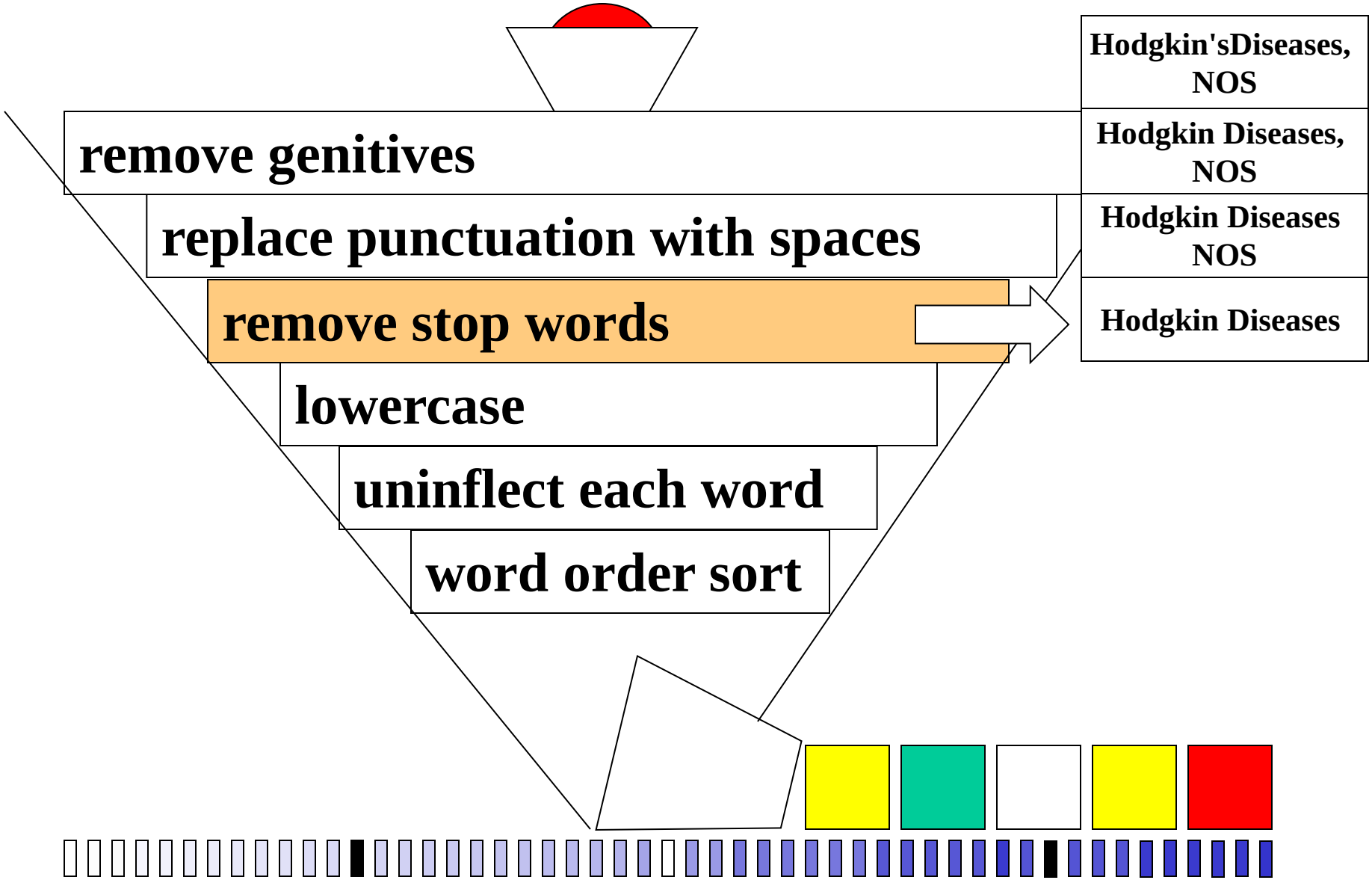
Hodgkin's Diseases, NOS

Lexical Tools: Norm



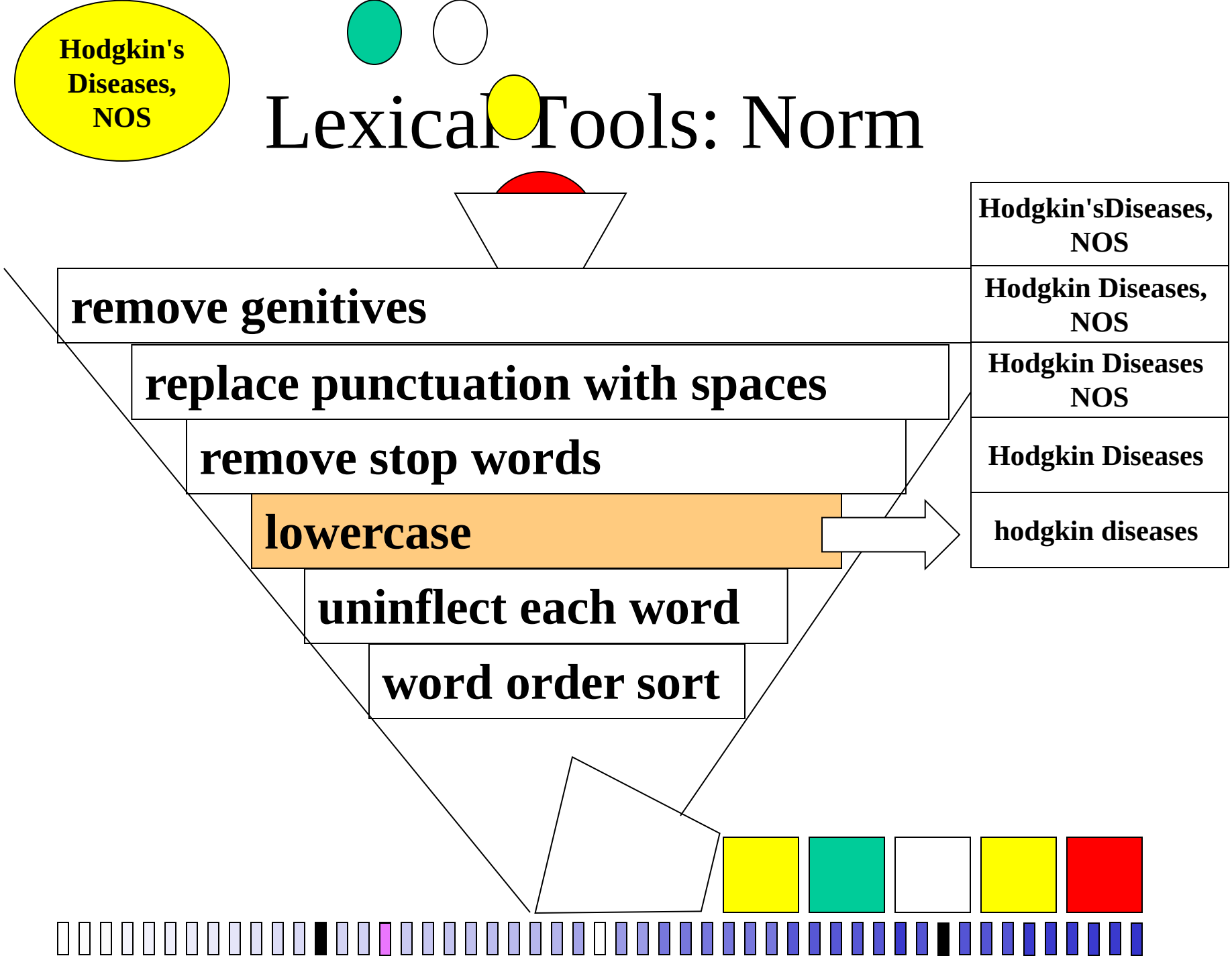
Hodgkin's Diseases, NOS

Lexical Tools: Norm



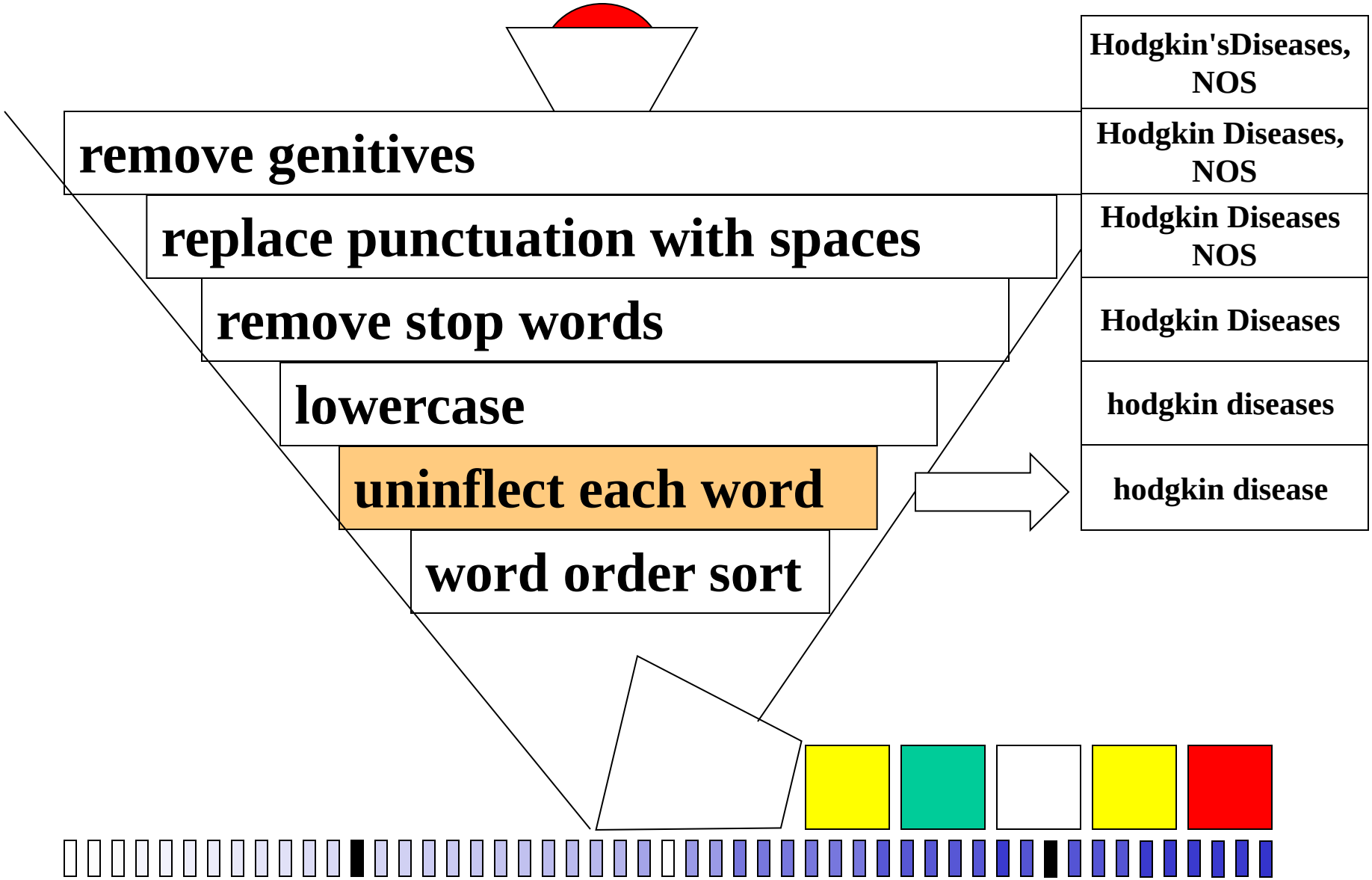
Hodgkin's Diseases, NOS

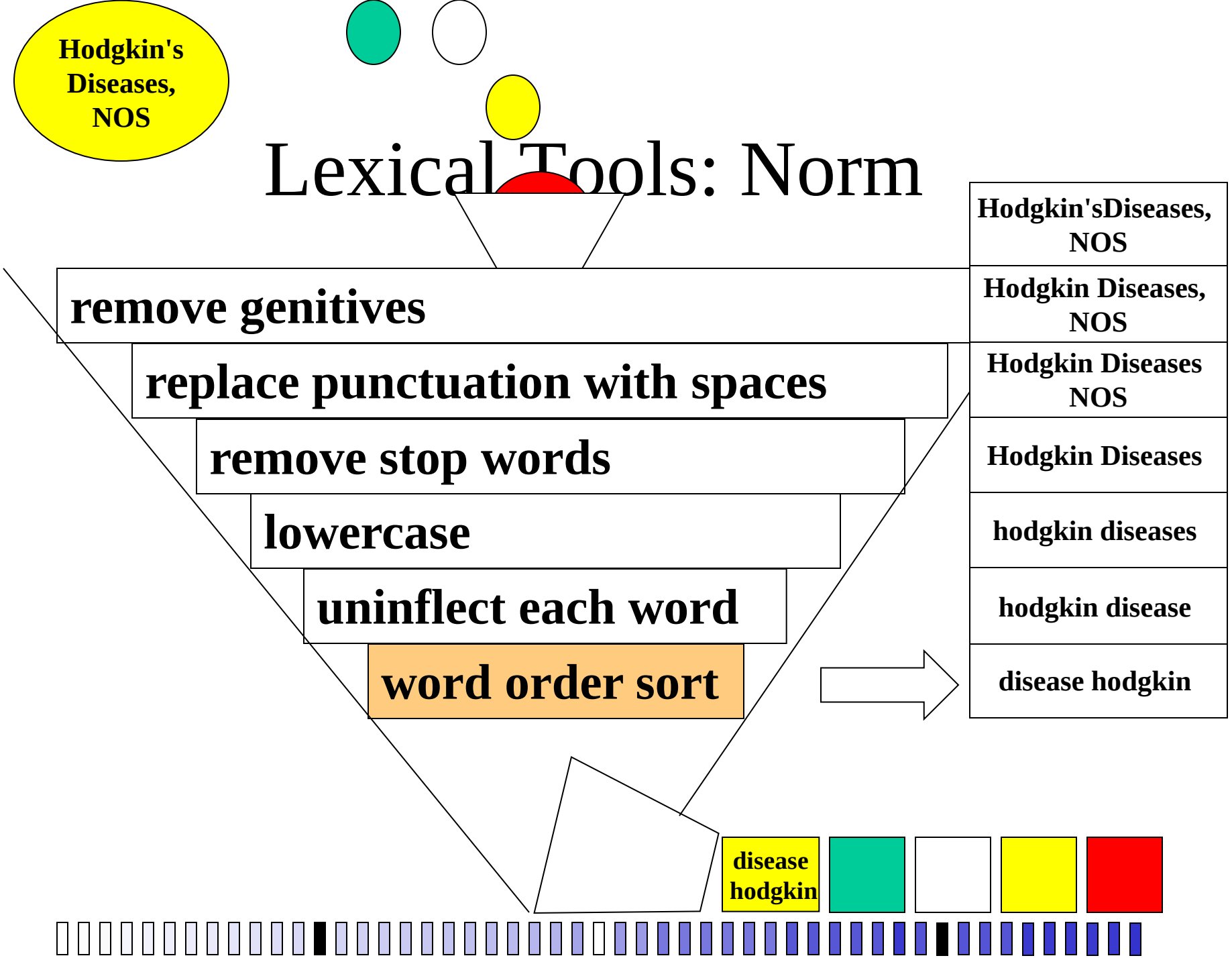
Lexical Tools: Norm



Hodgkin's Diseases, NOS

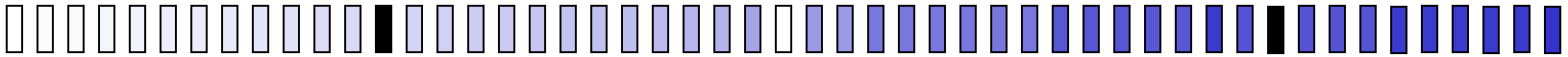
Lexical Tools: Norm

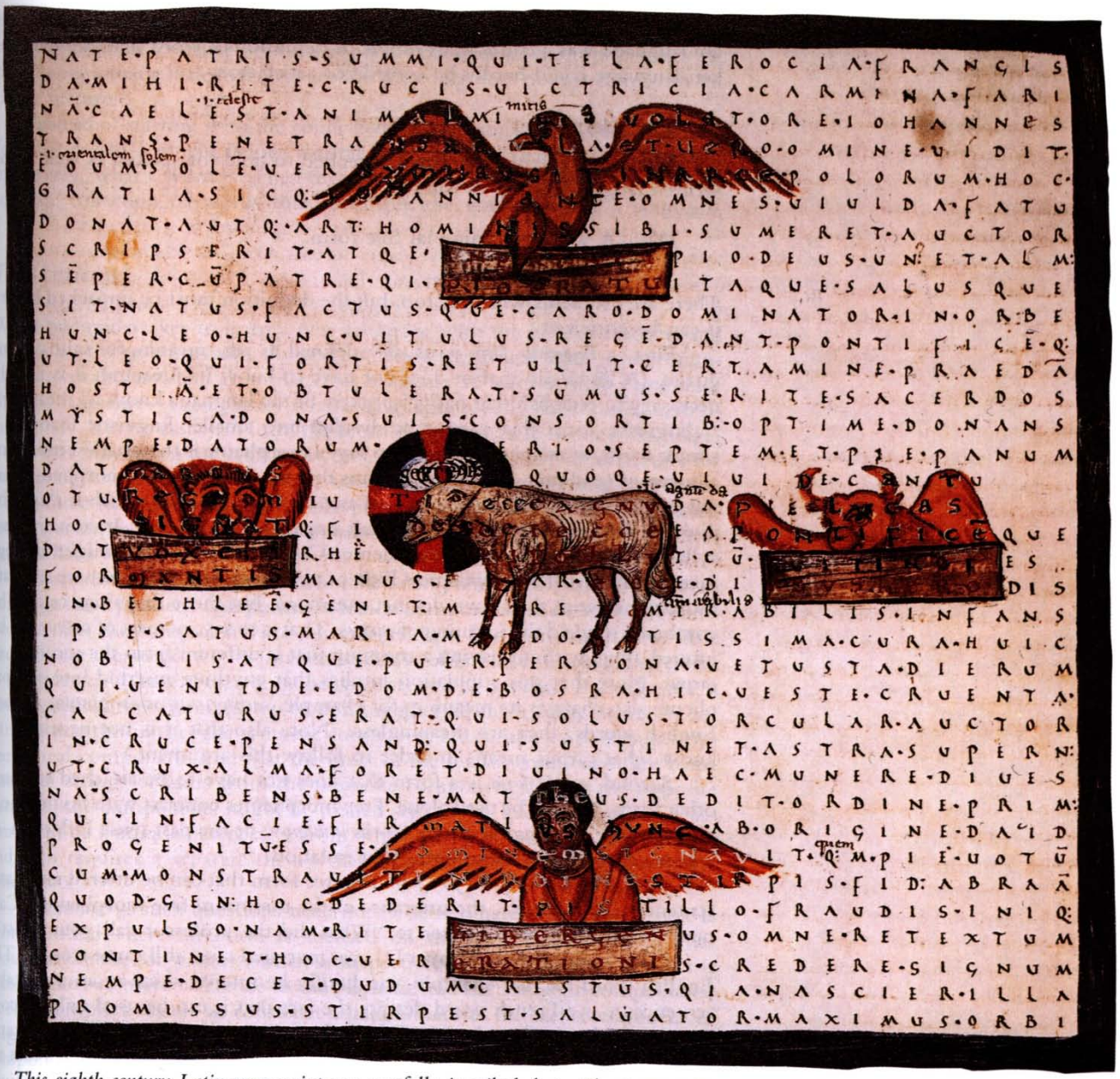



Hodgkin's Diseases, NOS
Lexical Tools: Norm
remove genitives
replace punctuation with spaces
remove stop words
lowercase
uninflect each word
word order sort
Hodgkin'sDiseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases NOS
Hodgkin Diseases
hodgkin diseases
hodgkin disease
disease hodgkin
disease hodgkin

Lexical Tools: Norm

Down's Syndrome Down Syndrome		down syndrome
Acetolyses acetolyses	**Norm to**	acetolysis
Lung cancer Cancer, lung		cancer lung





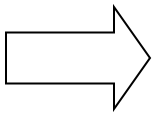
This eighth-century Latin manuscript was carefully inscribed, but without spaces to mark word boundaries.

Lexical Tools: WordInd

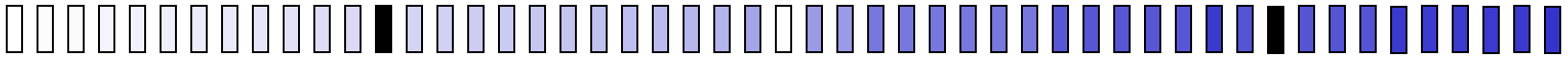
Wordind is a tool to break terms into words.

It is used to take a row from a Metathesaurus table that contains a term, sentence, paragraph, story,

and break the text part of that row into it's constituent words.

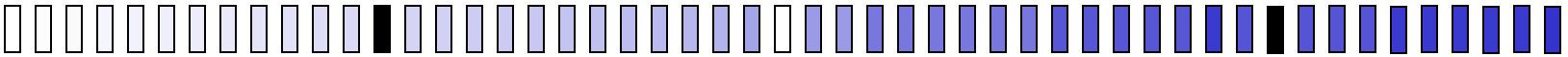


wordind
is
a
tool
to
break
terms
into
words
it
is
used
to



Lexical Tools: WordInd

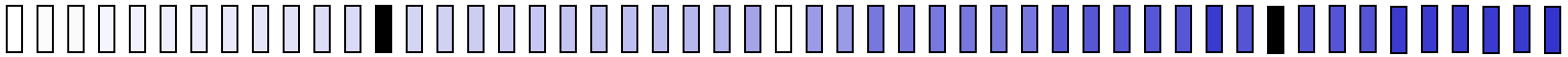
- Breaks words into tokens
- Passes other fields to output, untouched
- Lowercases
- Removes white space and punctuation



Lexical Tools: WordInd

Useful command line options for wordInd

-t[:Num]	Defines what field to tokenize
-f[:Num[:Num]]	Defines what fields get passed through



Lexical Tools: WordInd

> wordInd –t:7 –F:1:6

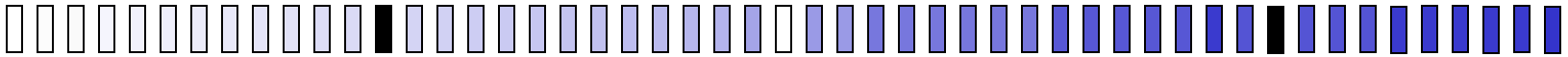
C0185495|ENG|P|L0223844|PF|S0298948|**Denis-Browne splint strapping**|3|

C0185495|S0298948|**denis**

C0185495|S0298948|**browne**

C0185495|S0298948|**splint**

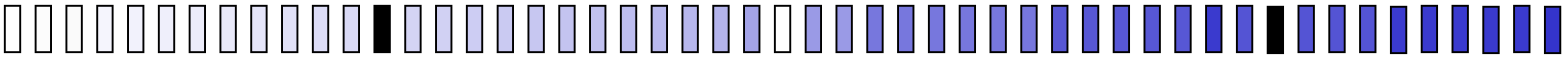
C0185495|S0298948|**strapping**



Lexical Tools: Lvg

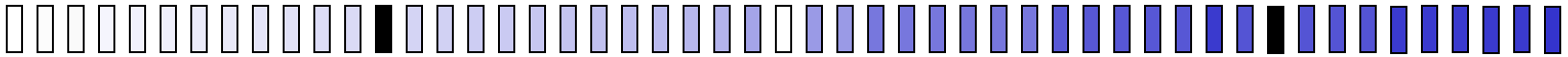
Lexical Tools: Flow Components

Mnemonic	Tool
A	Return known acronyms
a	Return known acronym expansions
B	Uninflect words in a term
b	Uninflect a term
C	Canonicalize
c	Tokenize a term into "words"
ca	Tokenize, keep everything
ch	Tokenize without breaking hyphens



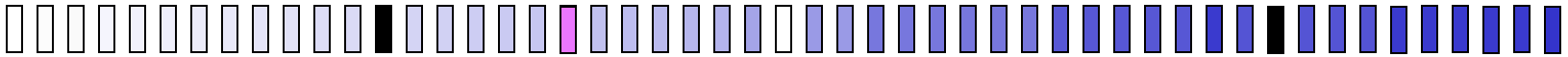
Lexical Tools: Flow Components

Mnemonic	Tool
Ct	**Retrieve the citation term**
d	**Generate derivational variants**
dc~N	**Generate derivational variants with specifying output categories**
R	**Generate derivational variants, recursively**
E	**Retrieve the unique EUI for a term**
f	**Filter output to contain only forms from lexicon**



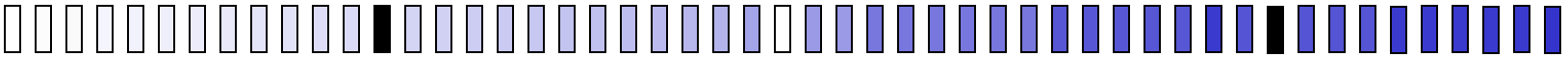
Lexical Tools: Flow Components

Mnemonic	Tool
Gn	**Generate known fruitful variants**
g	**Remove genitive**
i	**Generate inflectional variants**
ici~Cats+Infls	**Generate inflectional variants, by Categories and inflections**
is	**Generate inflectional variants, by Categories and inflections**
Si	**Inflections to simplified inflections**



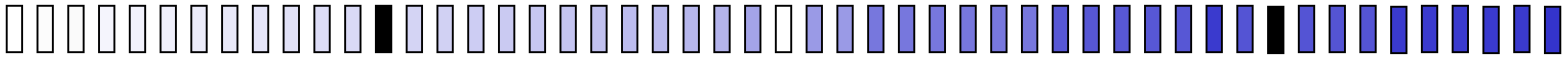
Lexical Tools: Flow Components

Mnemonic	Tool
L	**Retrieve category and inflection for a term**
l	**Lowercase**
N	**Normalize the input text in a non-canonical way (Norm)**
o	**Replace punctuations with spaces**
p	**Strip Punctuation**
q	**Strip diacritics**

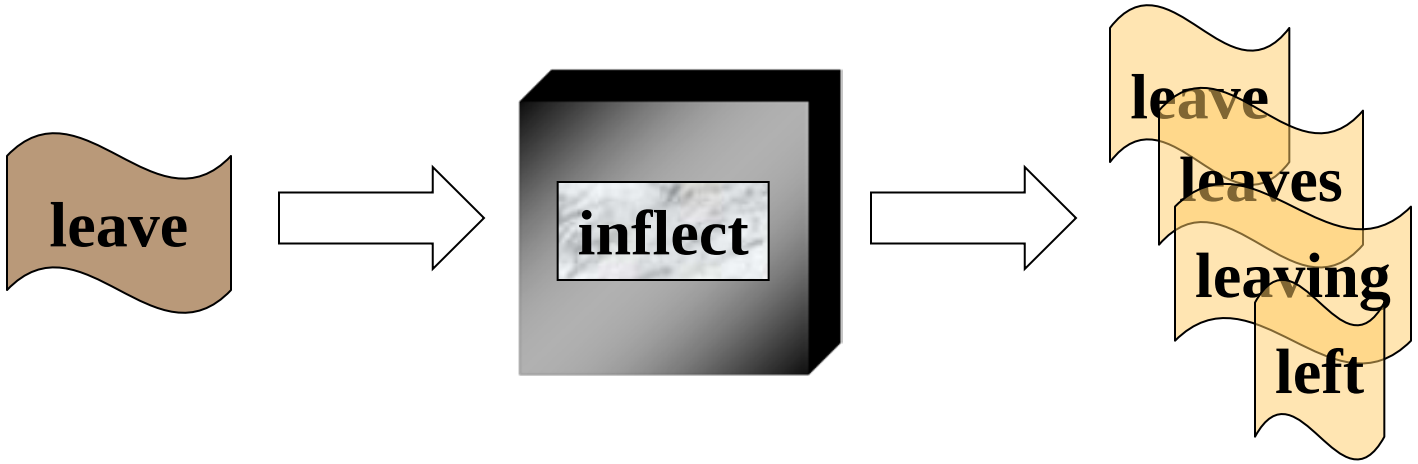


Lexical Tools: Flow Components

Mnemonic	Tool
s	**Generate known spelling variants**
t	**Strip stop words**
u	**Uninvert the input phrase around commas**
w	**Sort words by order**
y	**Generate synonyms**
r	**Generate synonyms, recursively**

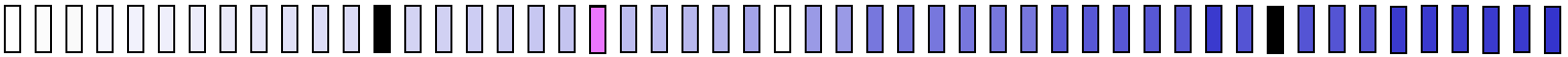


Lexical Tools: Flows

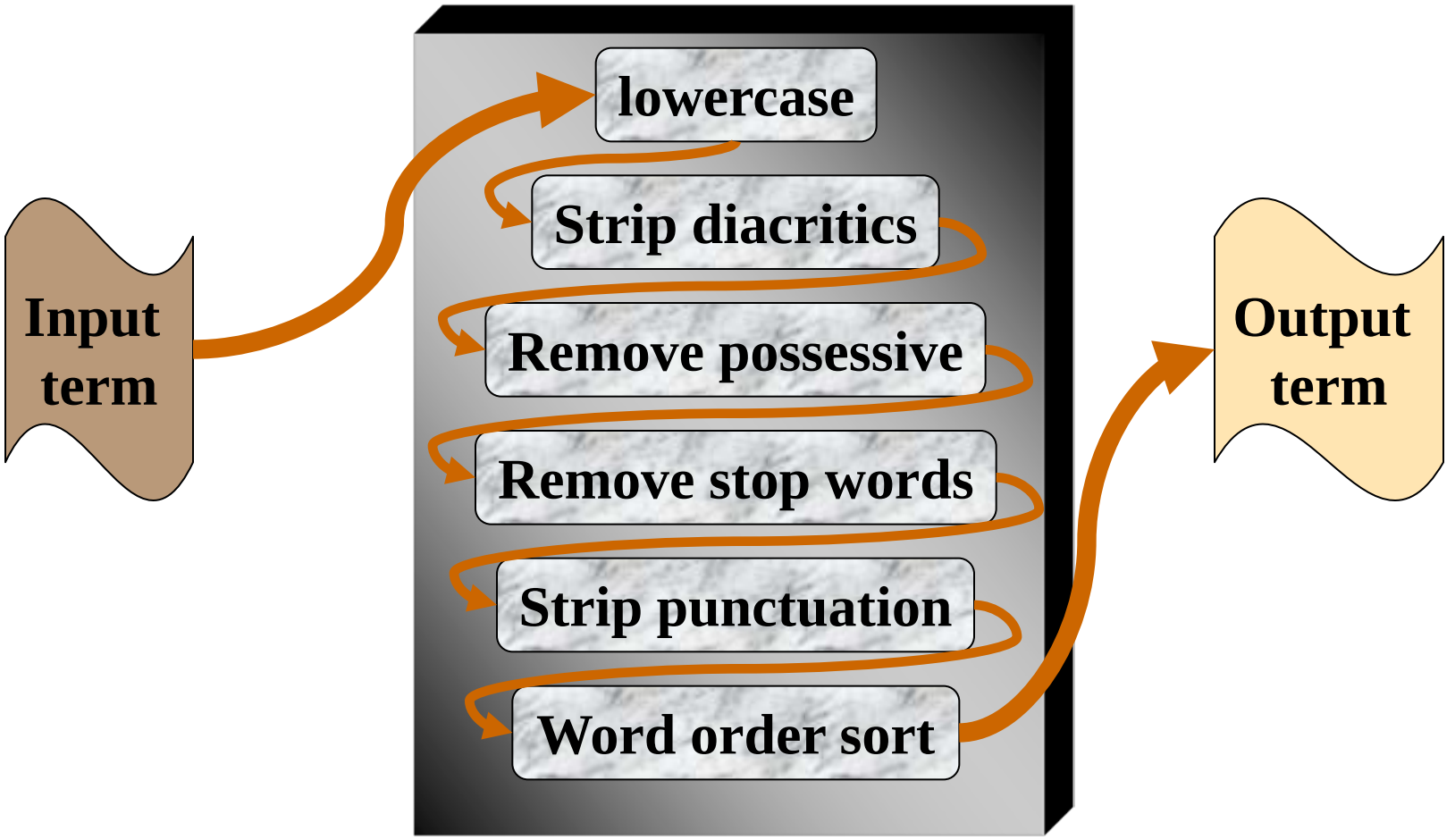


Lexical Tools: Flows

> lvg –f:i
leave
leave|**leave**|128|1|i|1|
leave|**leave**|128|512|i|1|
leave|**leaves**|128|8|i|1|
leave|**left**|1024|64|i|1|
leave|**left**|1024|32|i|1|
leave|**leave**|1024|1|i|1|
leave|**leave**|1024|262144|i|1|
leave|**leave**|1024|1024|i|1|
leave|**leaves**|1024|128|i|1|
leave|**leaving**|1024|16|i|1|



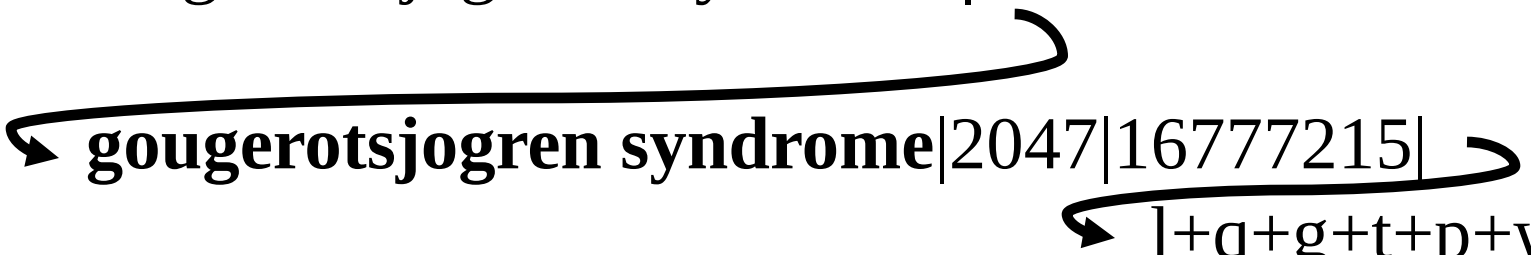
Lexical Tools: A Serial Flow



Flow components can be arranged so that the output of one is the input to another.

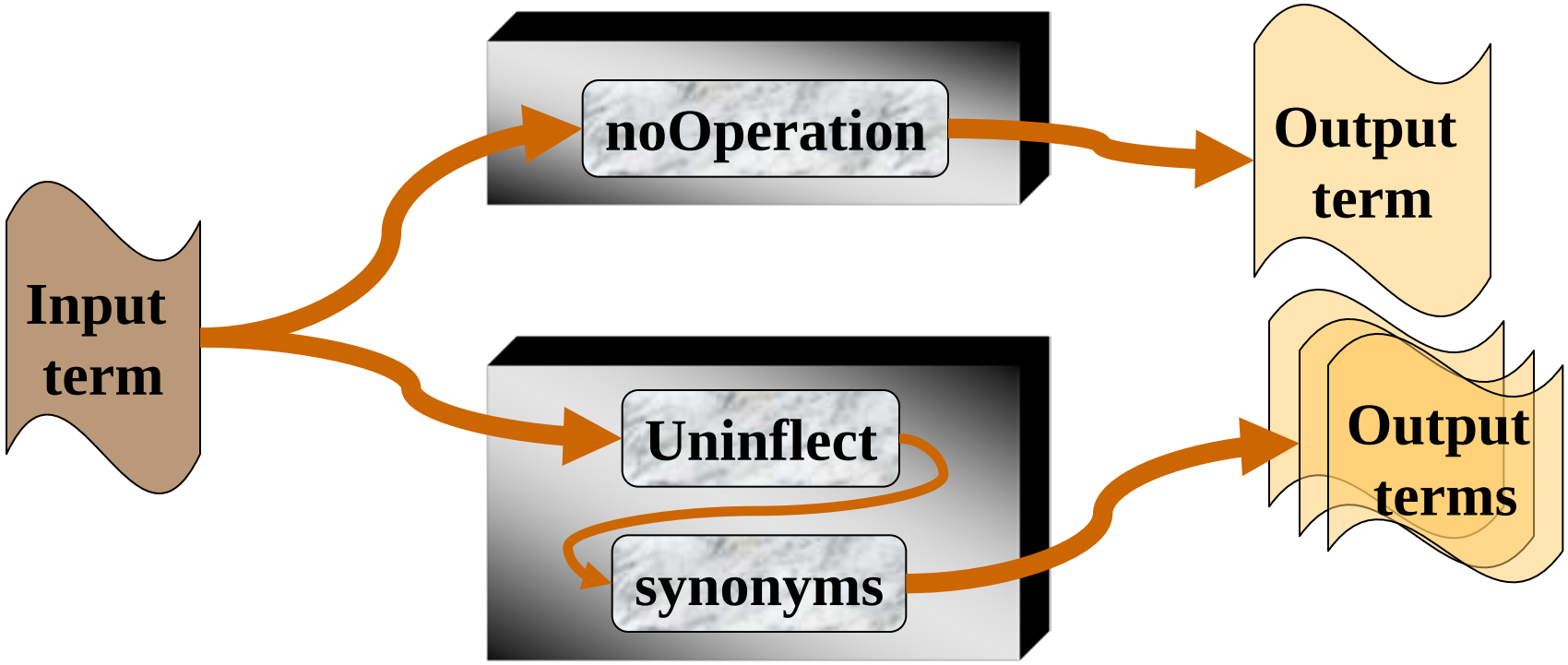
Lexical Tools: A Serial Flow

> lvg –f:l:q:g:t:p:w
The Gougerot-Sjögren's Syndrome
The Gougerot-Sjögren's Syndrome|



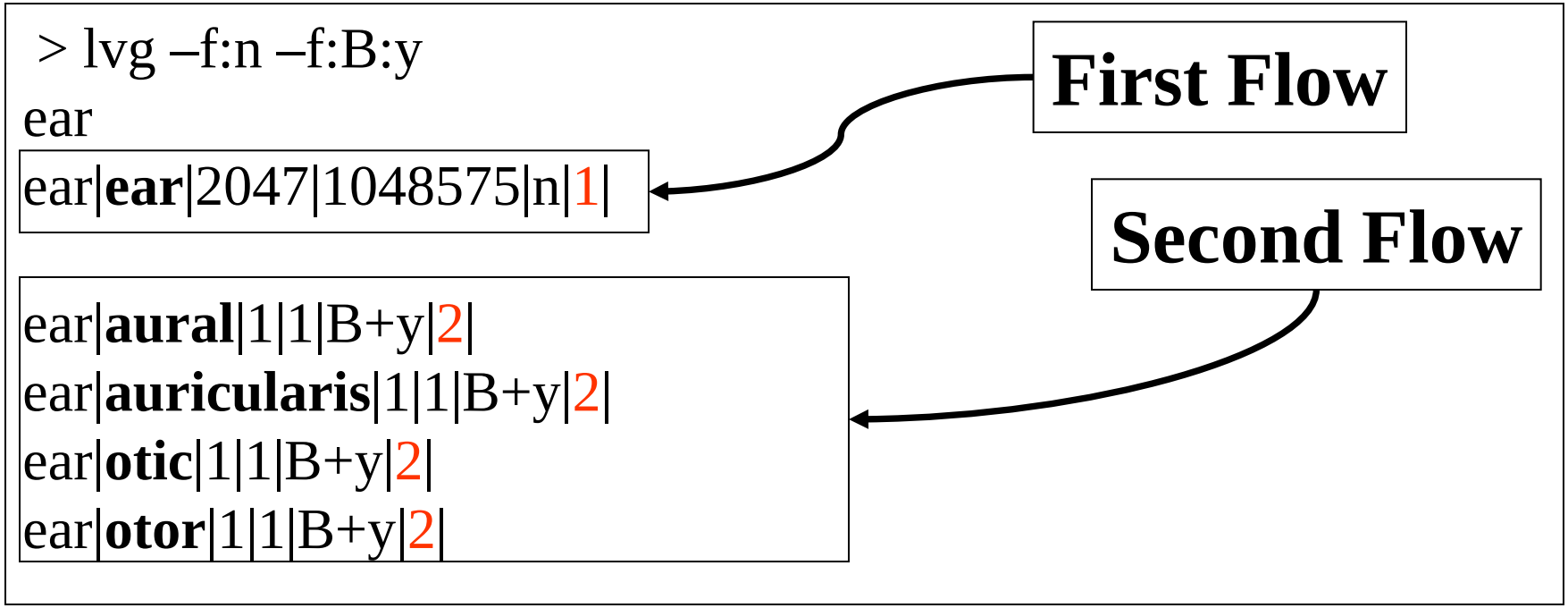
gougerotsjogren syndrome|2047|16777215|
l+q+g+t+p+w|1|

Lexical Tools: Parallel Flows



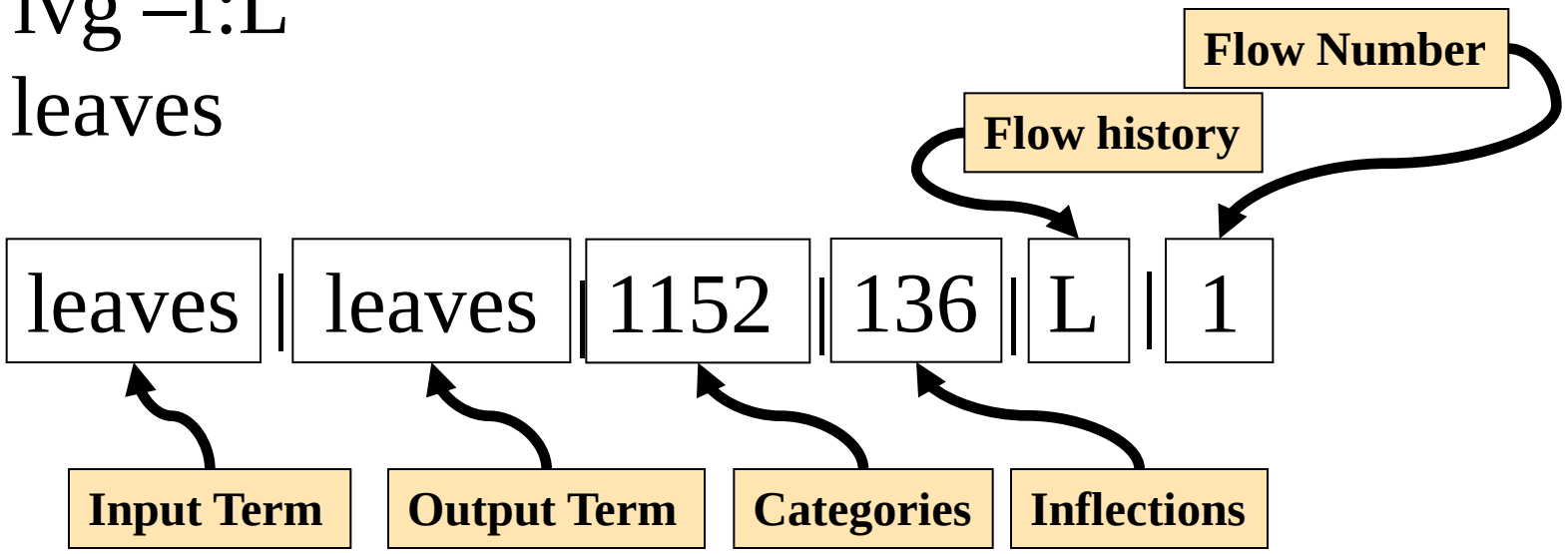
Multiple flows can be defined

Lexical Tools: Parallel Flows

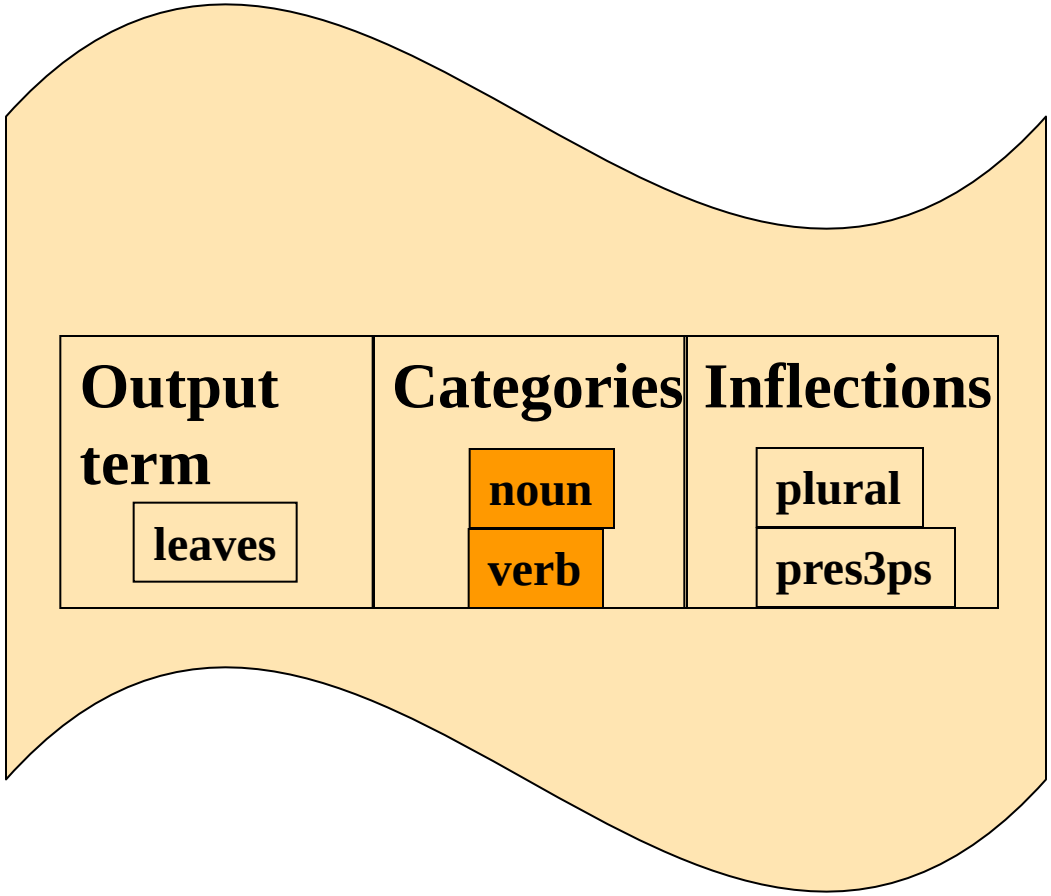


Lexical Tools: Fielded Output

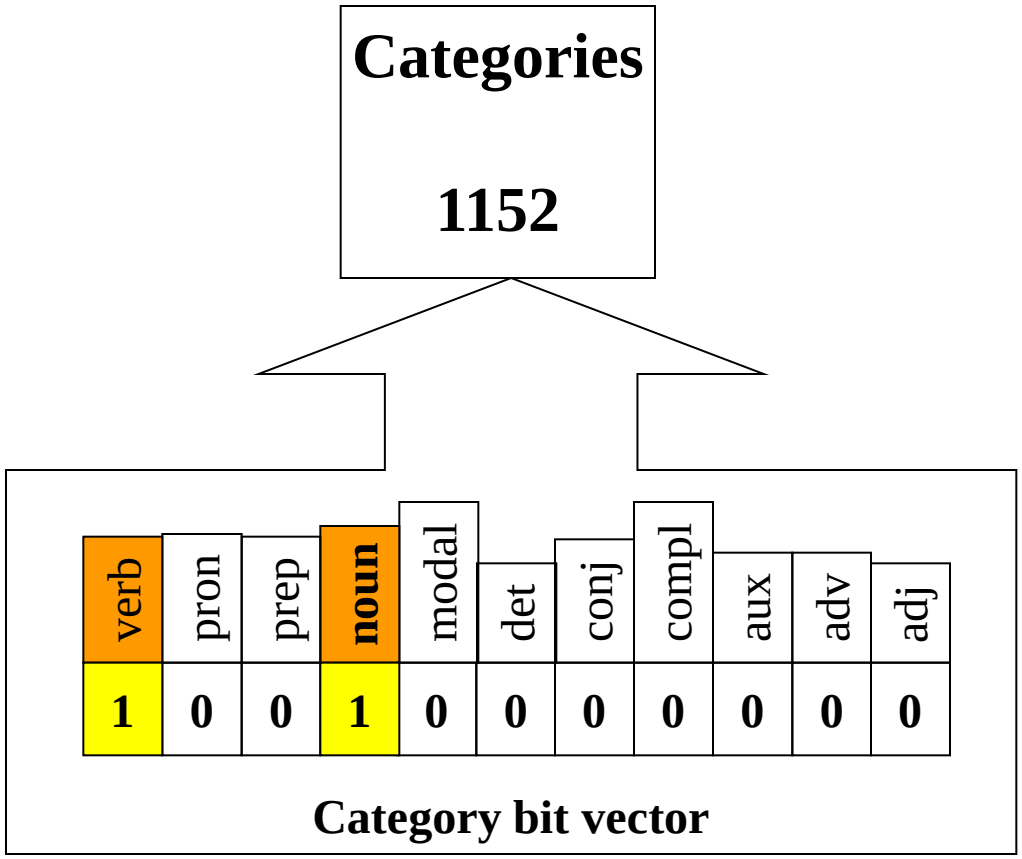
> lvg –f:L
leaves



Lexical Tools: Fielded Output



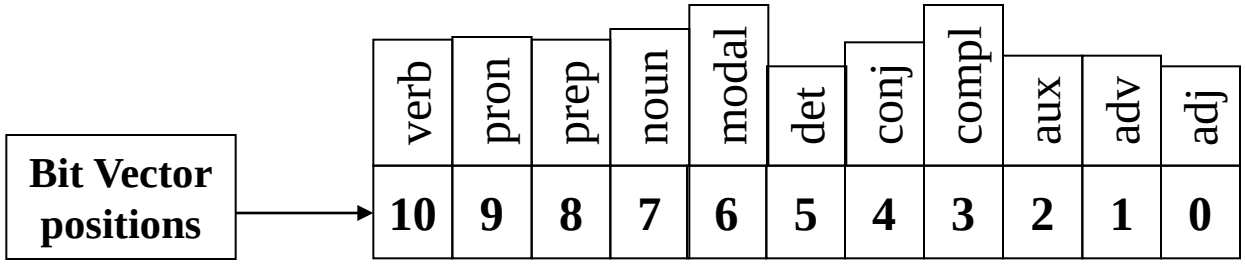
Lexical Tools: Categories



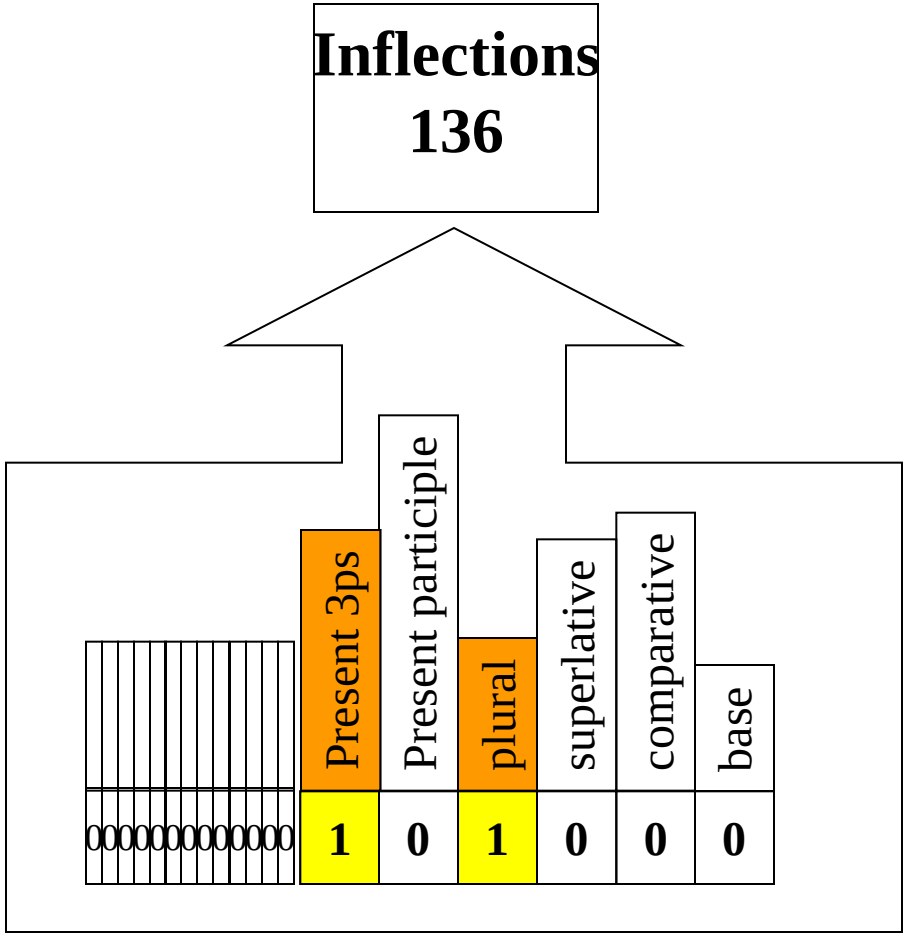
Lexical Tools: Categories

Adjective	1
Adverb	2
Auxiliary	4
Complement	8
Conjunction	16
Determiner	32

Modal	64
Noun	128
Preposition	256
Pronoun	512
Verb	1024

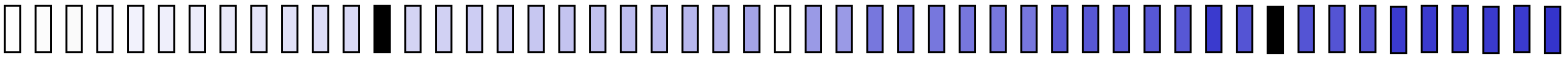


Lexical Tools: Inflections



Lexical Tools: Inflections

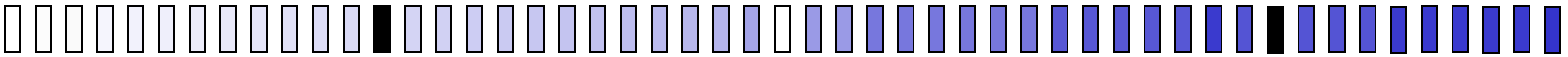
Base	1
Comparative	2
Superlative	4
Plural	8
Present Participle	16
Past	32
Past Participle	64
Present 3rd Person Singular	128



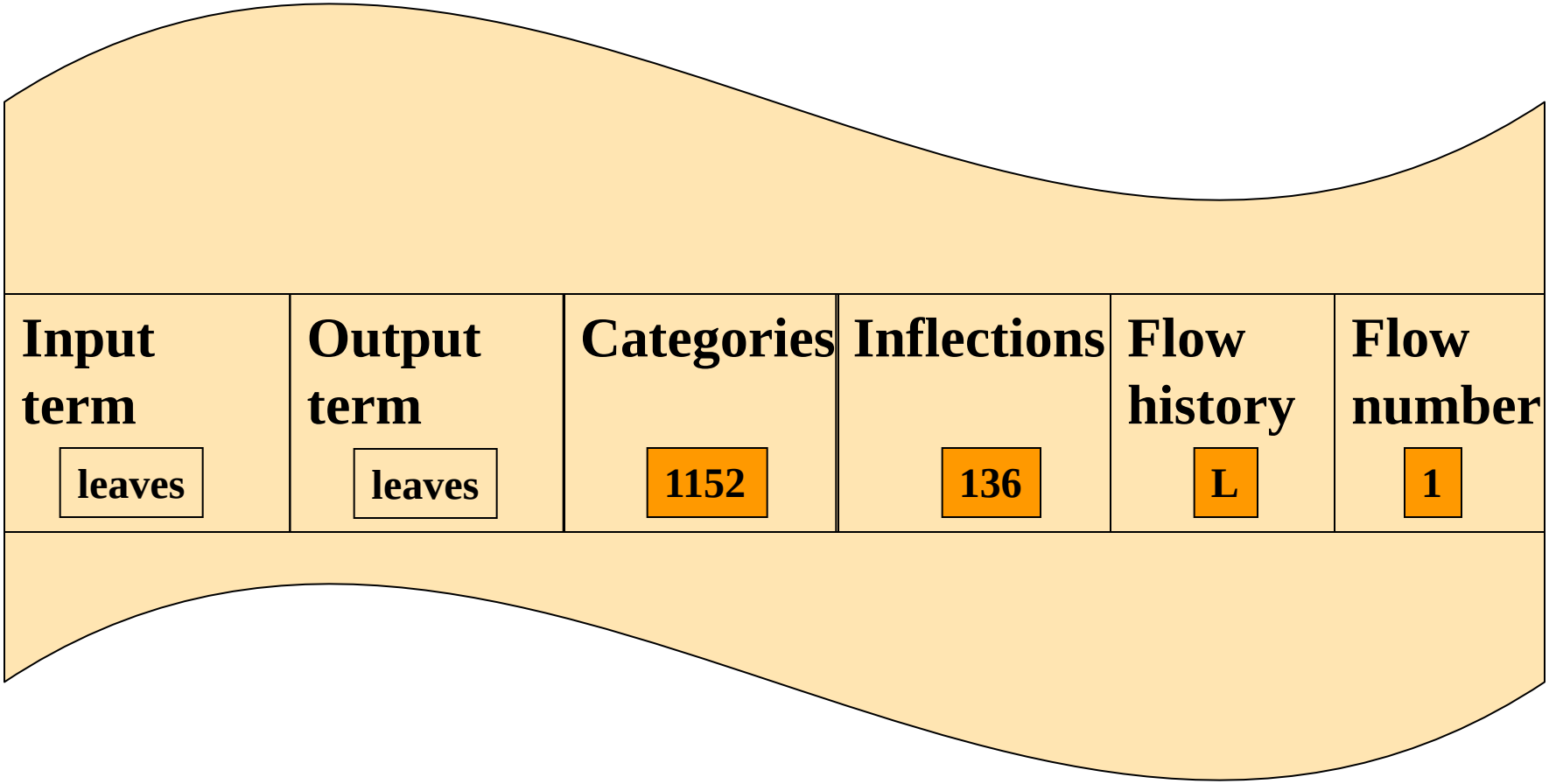
Lexical Tools: Inflections

Inflection	Expanded	Mapped
Positive	256	1
Singular	512	1
Infinitive	1024	1
Pres 123p	2048	128
Past Neg	4096	32
pres123pNeg	8192	128
Pres1s	16384	128
past1p23pNeg	32768	32
past1s3sNeg	65536	32

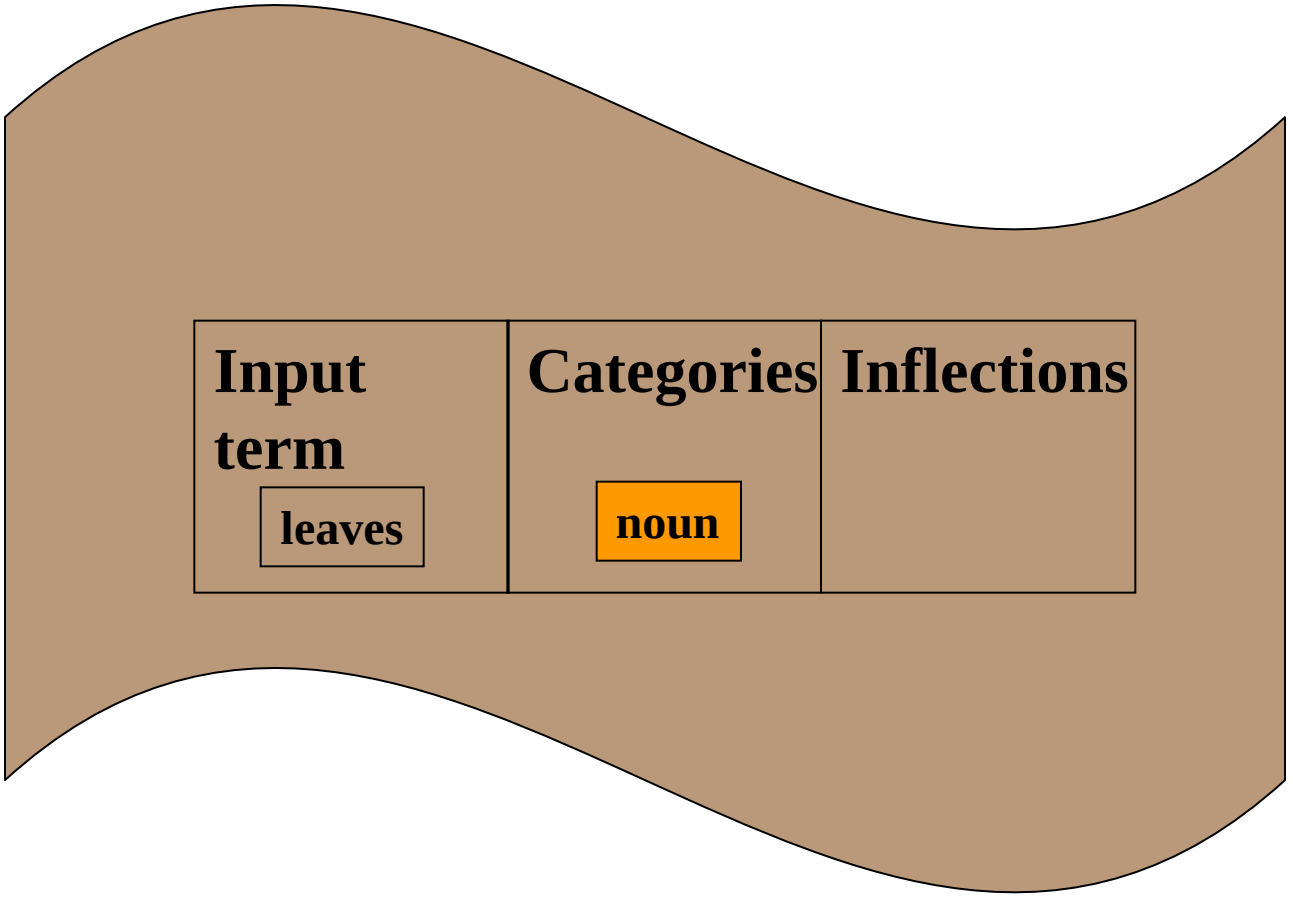
Inflection	Expanded	Mapped
Pres1p23p	131072	32
pres1p23p	262144	128
pres1p23pNeg	524288	128
past1s3s	1048576	32
pres	2097152	128
pres3sNeg	4194304	128
presNeg	8388608	128
all	16777215	255



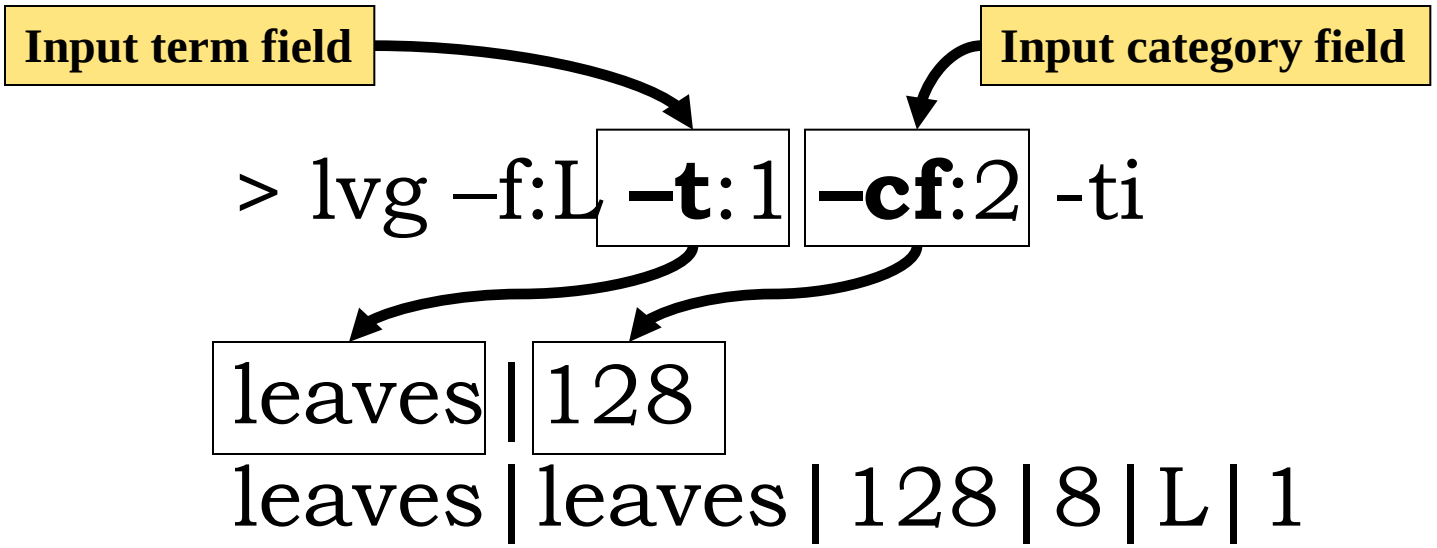
Lexical Tools: Fielded Output



Lexical Tools: Fielded Input

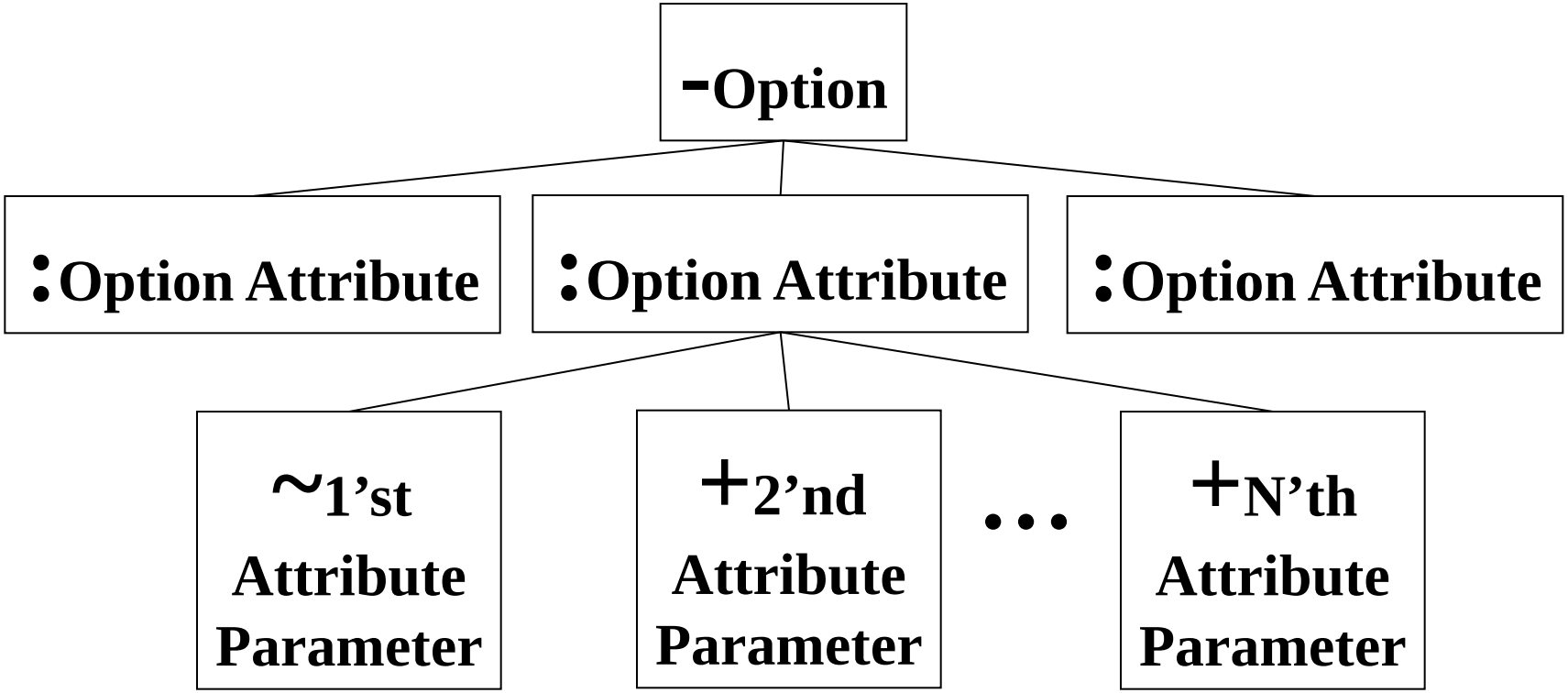


Lexical Tools: Fielded Input



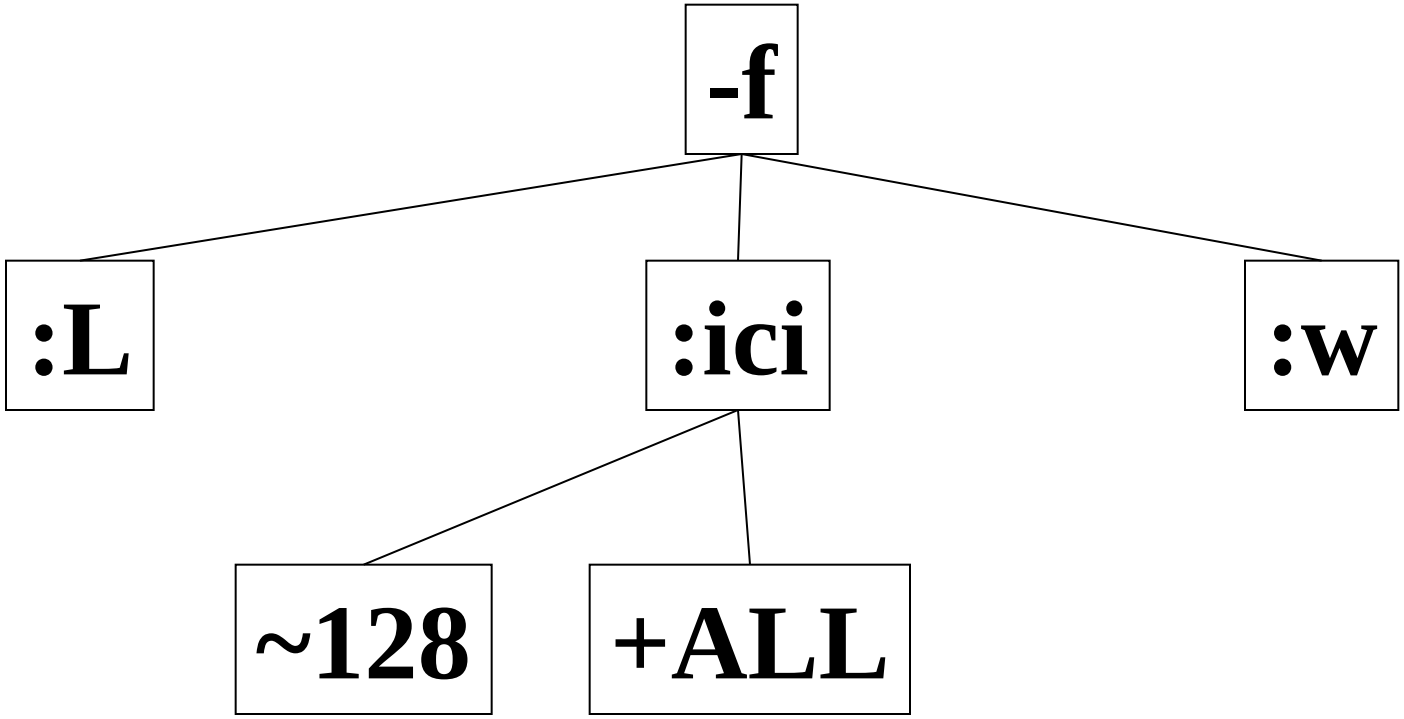
Lexical Tools: Command Line Syntax

- Hierarchical structure

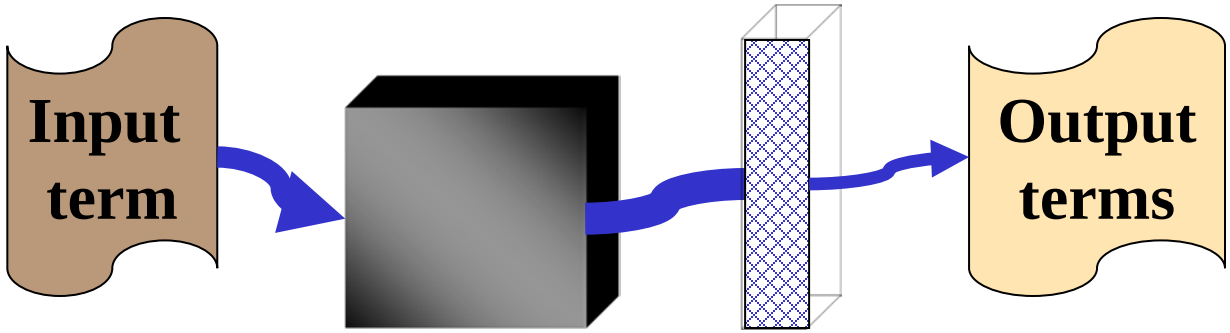


Lexical Tools: Command Line Syntax

-f:L:ici~128+ALL:w



Lexical Tools: Post Flow Options



SC	**Show category names**
SI	**Show inflection names**
ccgi	**Mark the end of the set of variants returned**
F:*Int[:Int]*	**Specify fields for outputs**
ti	**Display the only input term in the output when using fielded input**
R:*Int*	**Restrict the number of variants returned**

Lexical Tools: Post Flow Options

Show category names

Show inflection names

> lvg -f:L **-SC -SI**

phosphoprotein

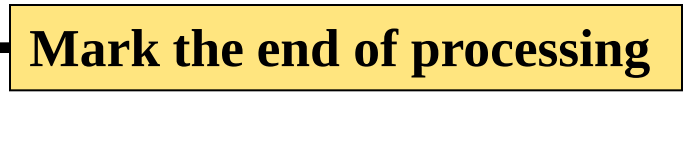
phosphoprotein|phosphoprotein|**<noun>**|**<base+singular>**|L|1|

sclerosing

sclerosing|sclerosing|**<adj+verb>**|**<base+presPart+positive>**|L|1|

Lexical Tools: Post Flow Options

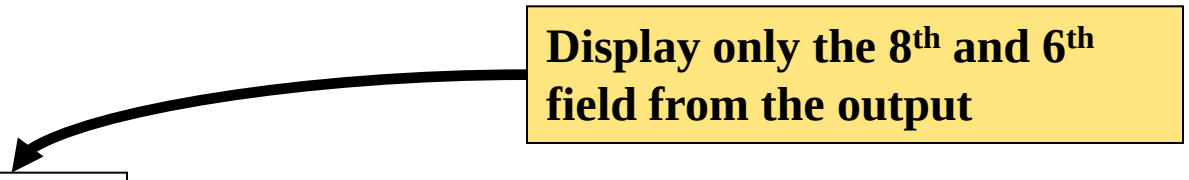
Mark the end of the set of variants returned




```
> lvg -f:L -ccgi
behavior
behavior|behavior|128|513|L|1|
__THE_END__
```

Lexical Tools: Post Flow Options

Specify fields for outputs



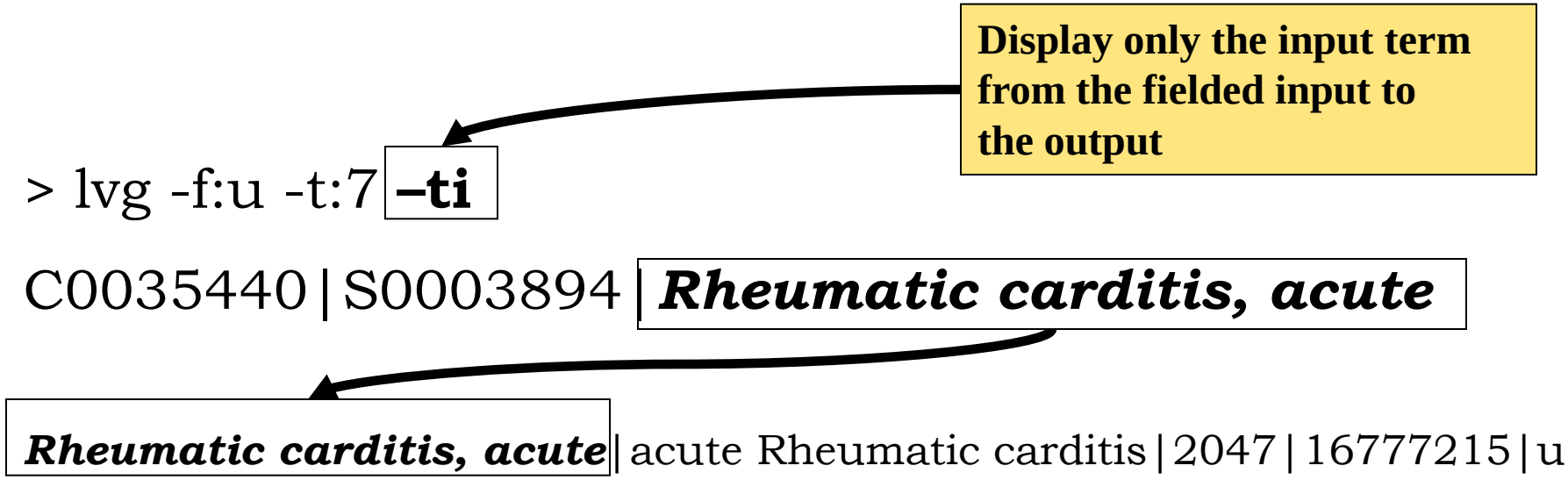
> lvg -f:u -t:7 **-F:8:6**

C0035440|ENG|S|L0035434|VW|S0003894|Rheumatic carditis, acute

acute Rheumatic carditis|S0003894

Lexical Tools: Post Flow Options

Display only the input term field when using fielded input



Lexical Tools: Post Flow Options

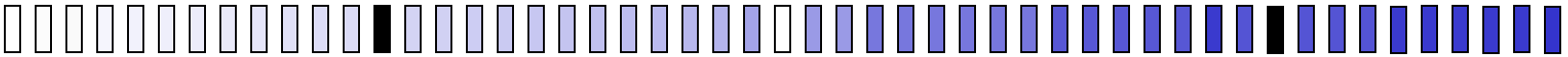
Restrict the number of variants returned

Limit the number of output terms to 2

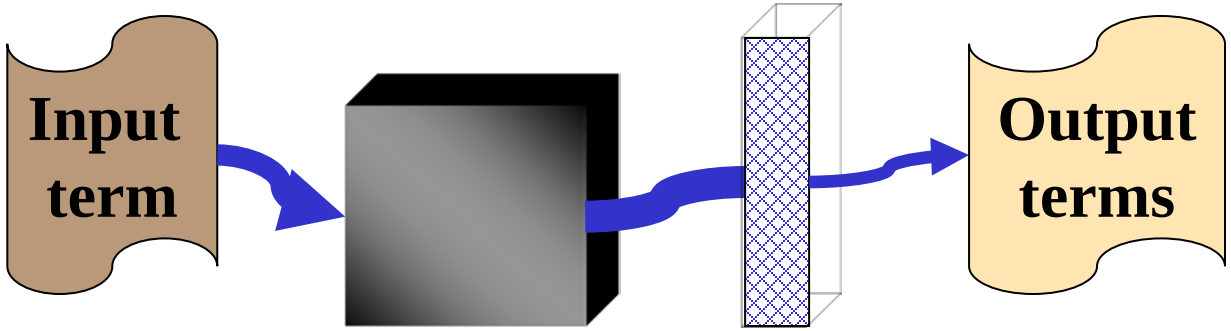
```
> lvg -f:i -R:2
foo
foo|foo|128|1|i|1|
foo|foos|128|8|i|1|
```

Note: Dangerous!
Do not try this at home!

Note: The unrestricted output would have produced 12 rows otherwise



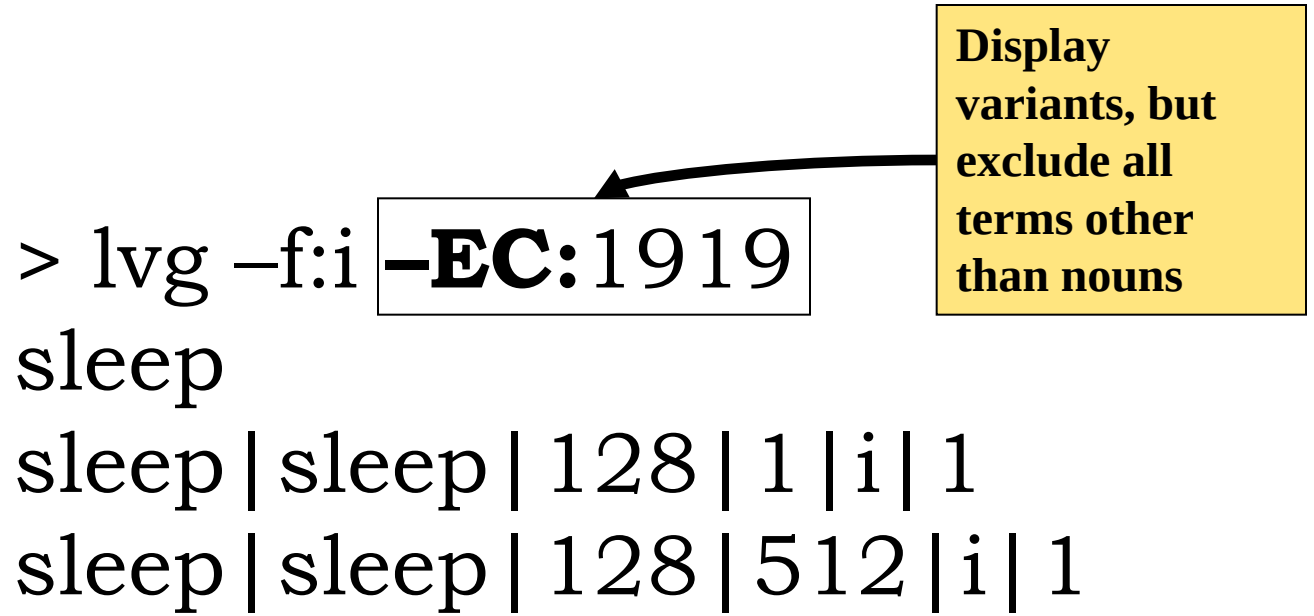
Lexical Tools: Post Flow Options



EC:*Long*	**Display variants: exclude categories specified**
EI:*Long*	**Display variants: exclude inflections specified**
DC:*Long*	**Display variants that only contain the categories specified**
DI:*Long*	**Display variants that only contain the inflections specified**

Lexical Tools: Post Flow Options

Display variants: exclude categories specified

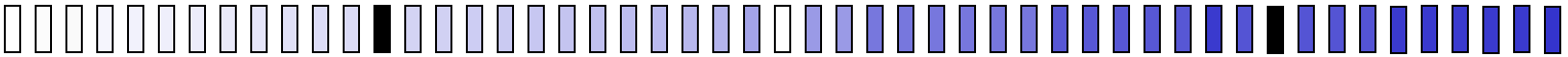


Lexical Tools: Post Flow Options

Display variants exclude inflection specified

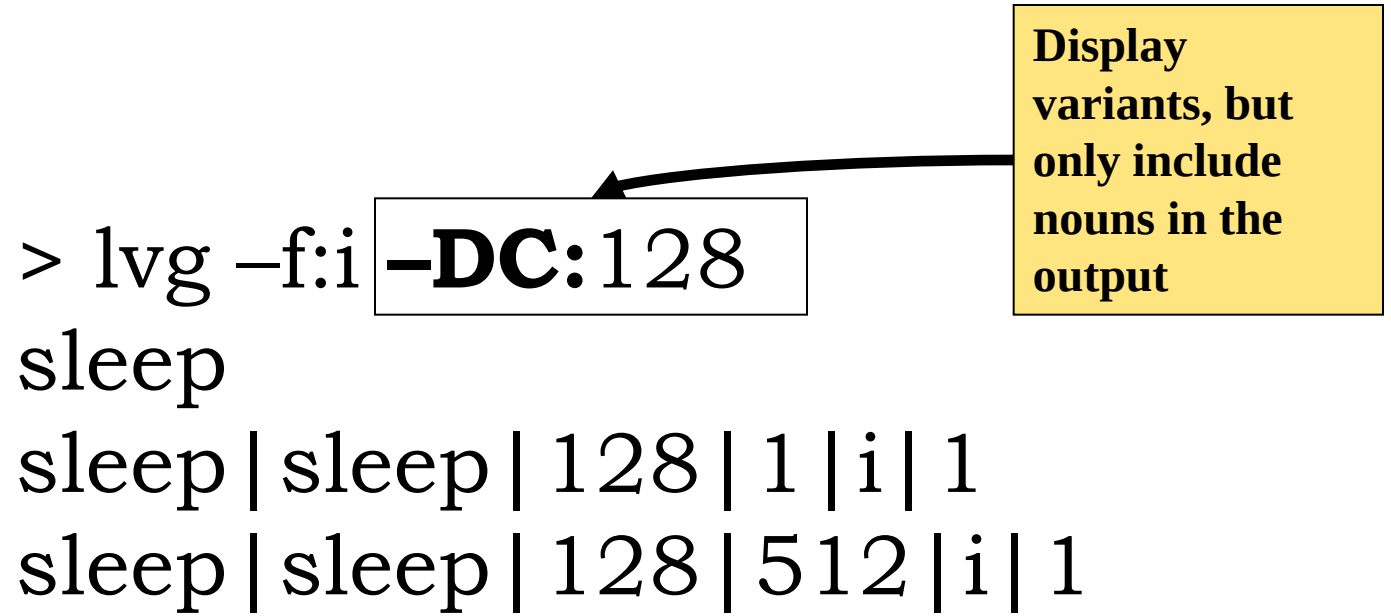
Display variants, but exclude base forms

> lvg –f:i –EC:1919 **–EI:**1
sleep
sleep|sleep|128|512|i|1



Lexical Tools: Post Flow Options

Display variants contain category specified

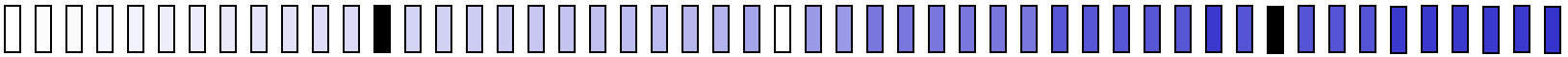


Lexical Tools: Post Flow Options

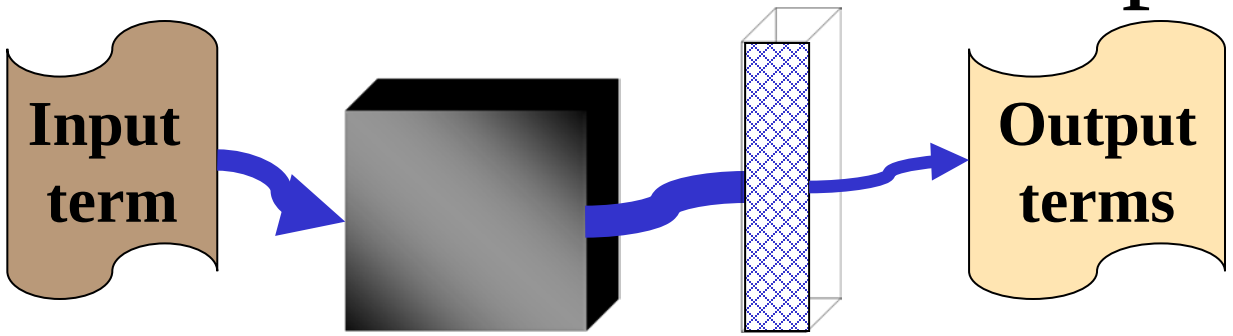
Display variants contain inflection specified

```
> lvg –f:i –DI:255
sleep
sleep|sleep|128|1|i|1|
sleep|sleep|1024|1|i|1|
sleep|slept|1024|64|i|1|
sleep|slept|1024|32|i|1|
sleep|sleeps|1024|128|i|1|
sleep|sleeping|1024|16|i|1|
```

Display variants, but only include “simplified” inflections



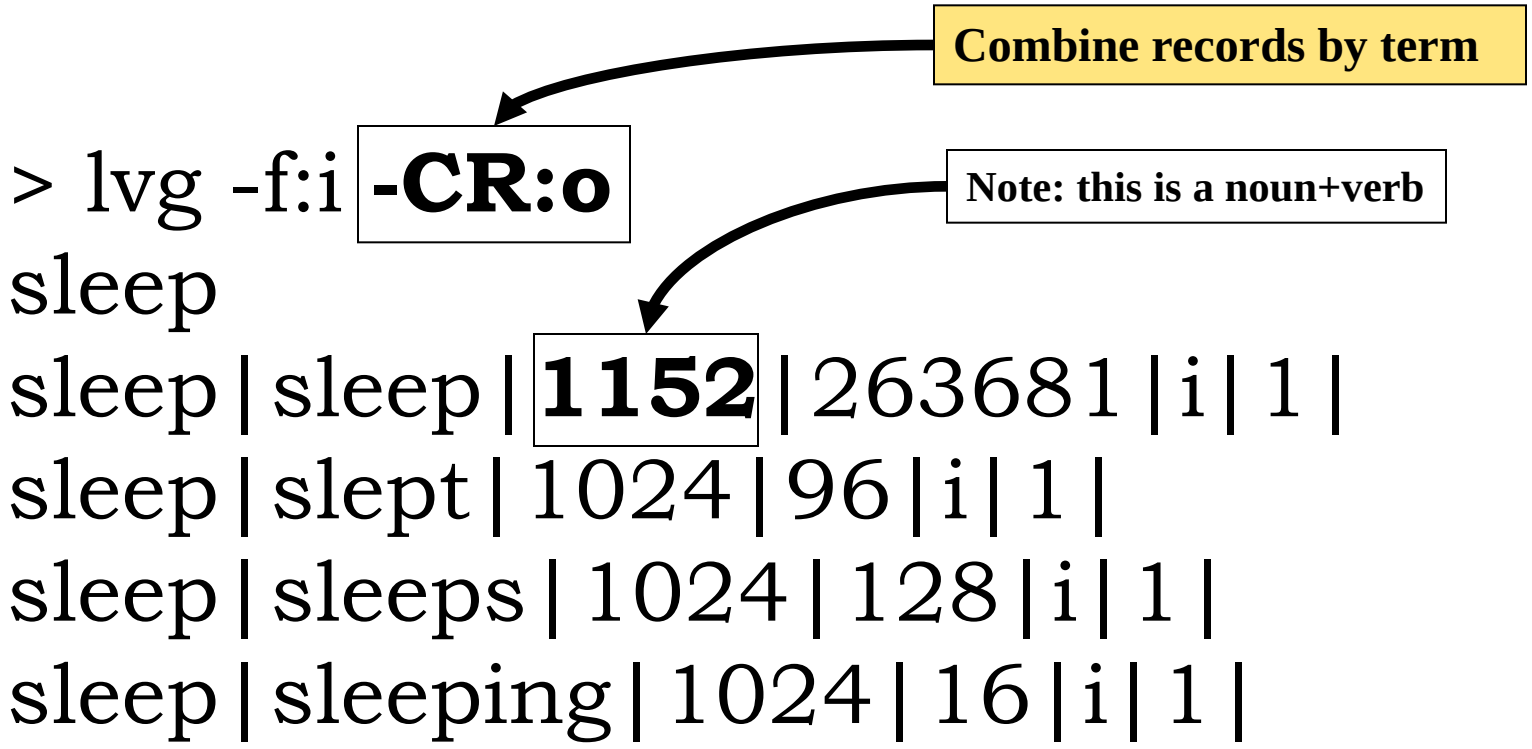
Lexical Tools: Post Flow Options



CR:o	**Combine record by output term**
CR:oc	**Combine record by output term and category**
CR:oi	**Combine record by output term and inflection**
St:o	**Sort outputs by terms in an alphabetical order**
St:oc	**Sort outputs by term and category**
St:oci	**Sort outputs by term, category, and inflection**

Lexical Tools: Post Flow Options

Combine record by output term



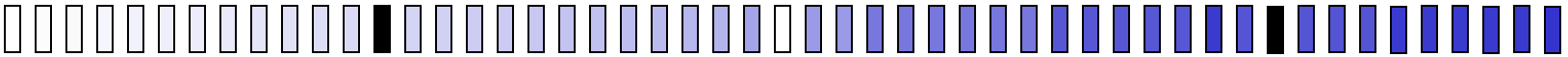
Lexical Tools: Post Flow Options

Combine record by output term and category

```
> lvg -f:i -CR:oc
sleep
sleep|sleep|128|513|i|1|
sleep|sleep|1024|263169|i|1|
sleep|slept|1024|96|i|1|
sleep|sleeps|1024|128|i|1|
sleep|sleeping|1024|16|i|1|
```

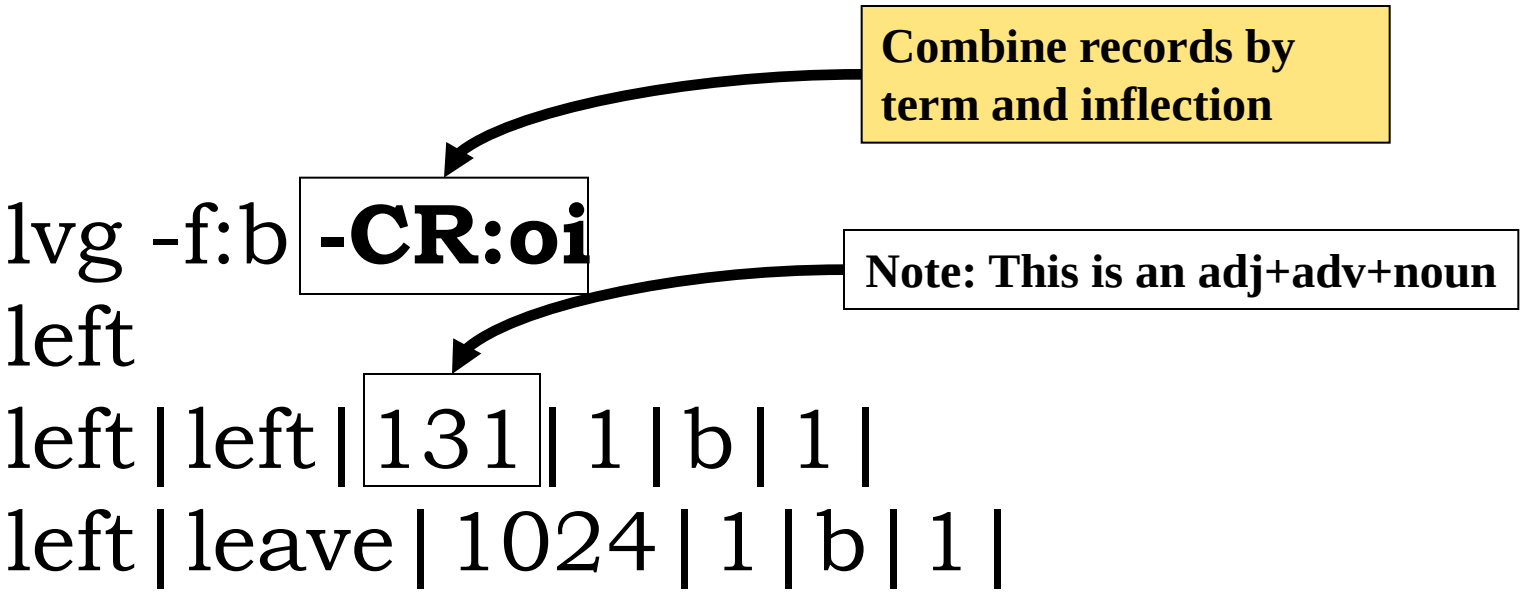
Combine records by term and category.

Note: this is both the base+singular inflections combined



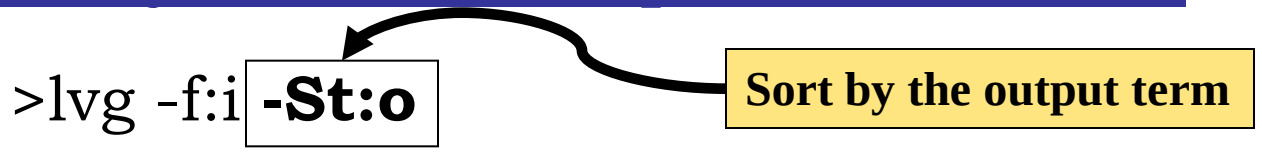
Lexical Tools: Post Flow Options

Combine record by output term and inflection



Lexical Tools: Post Flow Options

Sort outputs by terms in an alphabetical order



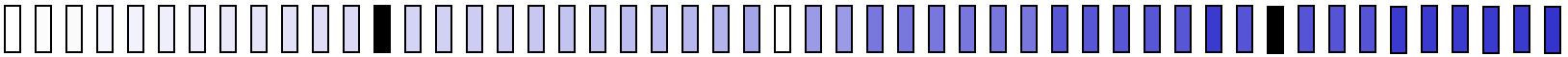

```
>lvg -f:i -St:o
see
see|saw|1024|32|i|1|
see|see|1024|1|i|1|
see|see|1024|262144|i|1|
see|see|1024|1024|i|1|
see|seeing|1024|16|i|1|
see|seen|1024|64|i|1|
see|sees|1024|128|i|1|
```

Lexical Tools: Post Flow Options

Sort outputs by term and category

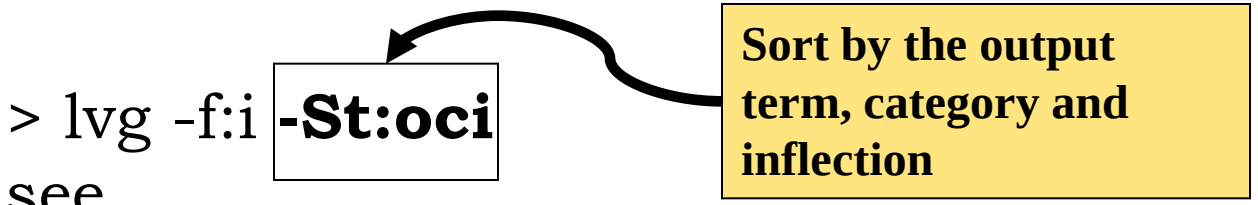
```
> lvg -f:i -St:oc
left
...
left|left|1|1|i|1|
left|left|1|256|i|1|
left|left|2|1|i|1|
left|left|2|256|i|1|
left|left|128|1|i|1|
left|left|128|512|i|1|
left|left|128|8|i|1|
left|left|1024|64|i|1|
left|left|1024|32|i|1|
left|lefts|128|8|i|1|
```

Sort by the output term and category



Lexical Tools: Post Flow Options

Sort outputs by term, category, and inflection

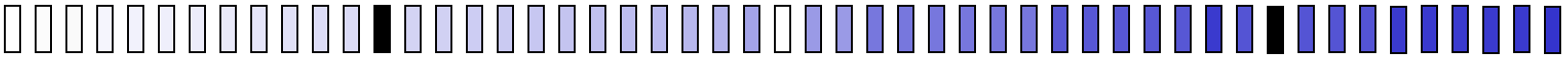



```
> lvg -f:i -St:oci
see
see|saw|1024|32|i|1|
see|see|1024|1|i|1|
see|see|1024|1024|i|1|
see|see|1024|262144|i|1|
see|seeing|1024|16|i|1|
see|seen|1024|64|i|1|
see|sees|1024|128|i|1|
```

Lexical Tools: Flow or Filter

Map the expanded inflections to the simplified inflections

```
>lvg -f:i:Si -CR:oc -St:oci
paint
paint|paint|128|1|i+Si|1|
paint|paint|1024|129|i+Si|1|
paint|painted|1024|96|i+Si|1|
paint|painting|1024|16|i+Si|1|
paint|paints|128|8|i+Si|1|
paint|paints|1024|128|i+Si|1|
```



Lexical Tools: Why Base+Singular

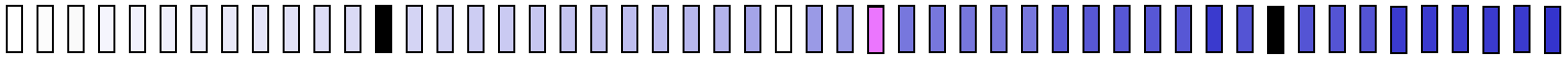
- Base form for Nouns:
  - Singular form
    - There are exceptions:
      - Police

      lvg -f:i -SC -SI -DC:128

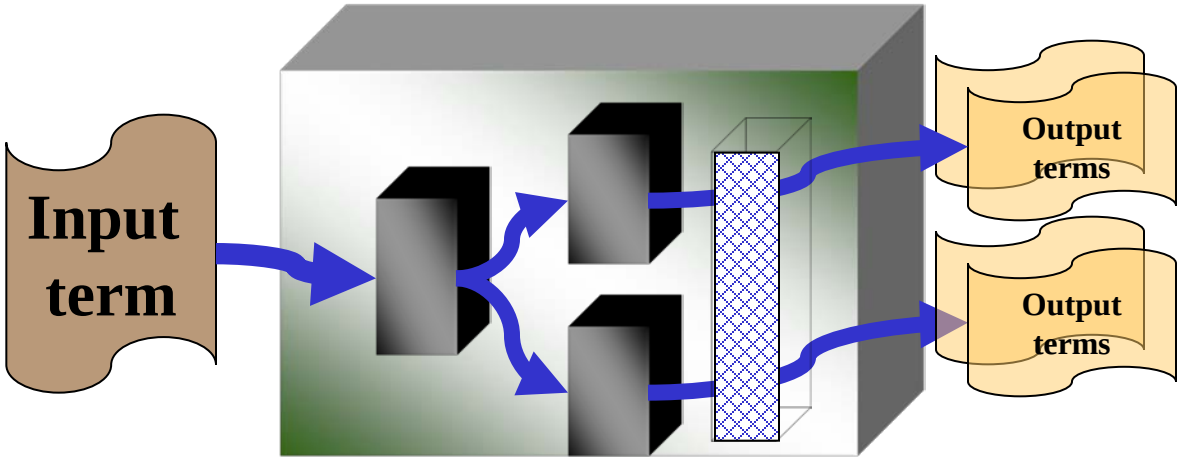
      police

      police|police|<noun>|<base>|i|1|

      police|police|<noun>|<plural>|i|1|



Lexical Tools: Global Behaviors



i:*filename*	Define input file name
o:*filename*	Define output file name
x:*filename*	Loading an alternative configuration file
p	Interactive prompt
m	Print extra information of flow mutations
s:*Str*	Defines a field separator.

Lexical Tools: Global Behaviors

Interactive prompt

> lvg -f:s -CR:oc **-p**

- Please input a term (type "Ctl-d" to Quit) >

color

color|color|128|513|s|1|

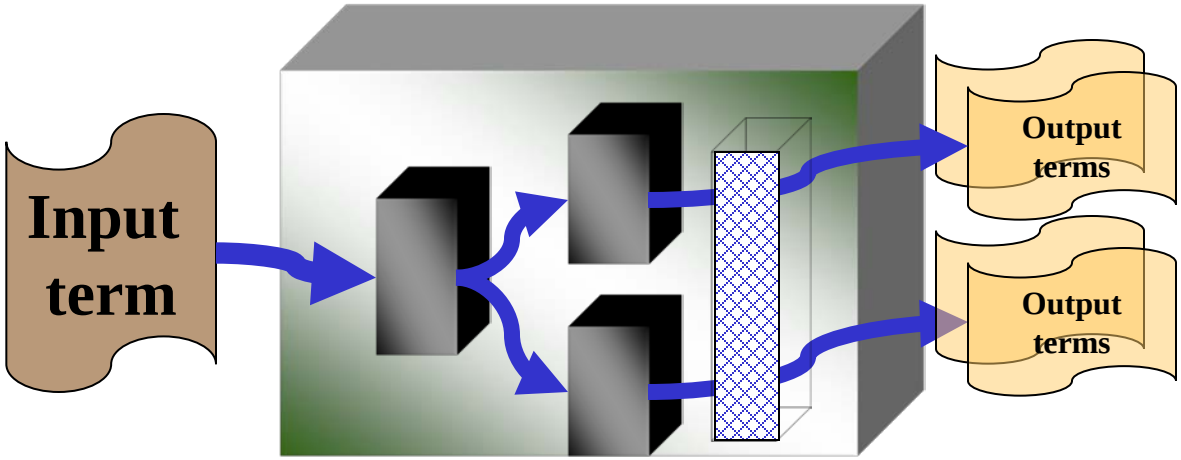
color|color|1024|263169|s|1|

color|colour|128|513|s|1|

color|colour|1024|263169|s|1|

- Please input a term (type "Ctl-d" to Quit) >

Lexical Tools: Global Behaviors

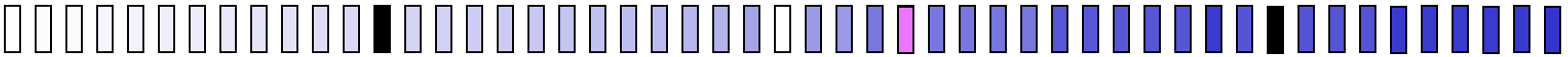


Kd:*Int*	**Configure the derivation morphology behavior**
Ki:*Int*	**Configure the inflection morphology behavior**

Lexical Tools: Global Behaviors

Configure the derivation morphology behavior

1	**Restrict the output to those variants which are known to the lexicon (default).**
2	**Restrict the output to those variants which are known to the lexicon, unless none of the variants are found in the lexicon, in which case the entire (rule-generated) list is returned.**
3	**No restriction on the output of the morphology. Both facts and rules generated variants are displayed.**



Lexical Tools: Global Behaviors

Configure the derivation morphology behavior

Restrict the derivations to terms that are known to the Lexicon

> lvg -f:d **-kd:**1 -m

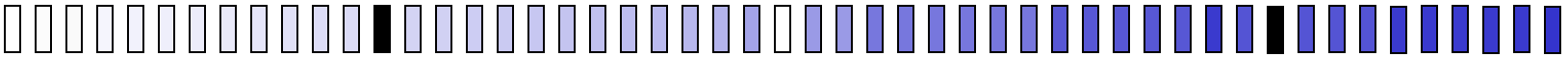
breath

breath|breather|128|1|d|1|RULE|$|verb|base|er$|noun|base|

breath|breathlessly|2|1|d|1|FACT|breath|128|breathlessly|2|

breath|breathy|1|1|d|1|RULE|$|noun|base|y$|adj|base|

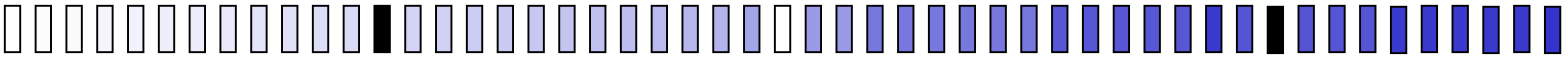
breath|breathless|1|1|d|1|FACT|breath|128|breathless|1|



Lexical Tools: Global Behaviors

Configure the inflection morphology behavior

1	**Restrict the output to those variants which are known to the lexicon.**
2	**Restrict the output to those variants which are known to the lexicon, unless none of the variants are found in the lexicon, in which case the entire (rule-generated) list is returned (Default).**
3	**No restriction on the output of the morphology. Both facts and rules generated variants are displayed**

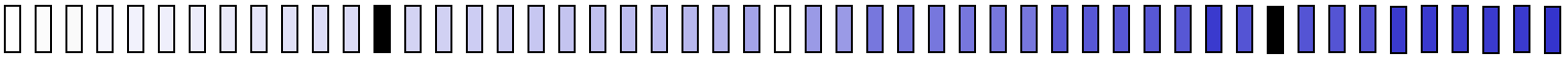


Lexical Tools: Global Behaviors

Configure the inflection morphology behavior

```
> lvg -f:i -m -ki:3 -R:5 -F:1:2:7:3:4 -SC -SI
Amish
Amish|Amish|FACT|<noun>|<base>
Amish|Amish|FACT|<noun>|<plural>
Amish|Amishs|RULE|<noun>|<plural>
Amish|Amishes|RULE|<noun>|<plural>
Amish|Amishs|RULE|<verb>|<pres>
```

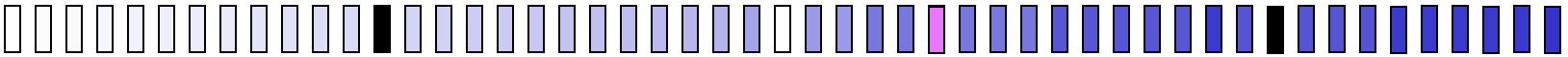
Return all inflected variants, both fact and rule generated



Lexical Tools: No Operation

-f:n	**Copies the input term to the output with no transformation**

> lvg -f:**n** -f:d -f:y -SC -SI
force
force|**force**|<all>|<all>|**n**|1|
force|forcefully|<adv>|<base>|d|2|
force|forceful|<adj>|<base>|d|2|
force|forcible|<adj>|<base>|d|2|
force|dynamic|<adj>|<base>|y|3|



Lexical Tools: Inflect

-f:i	**Generate inflectional variants**

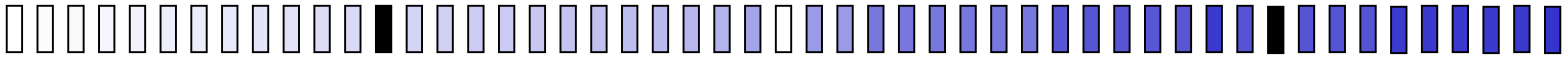
> lvg -f**:i** -SC -SI

West Nile Virus

West Nile Virus|**West Nile virus**|<noun>|<base>|i|1|

West Nile Virus|**West Nile virus**|<noun>|<singular>|i|1|

West Nile Virus|**West Nile viruses**|<noun>|<plural>|i|1|



Lexical Tools: Inflect

-f:ici~cats+infls	**Generate inflections, filter by cat and/or inflection**

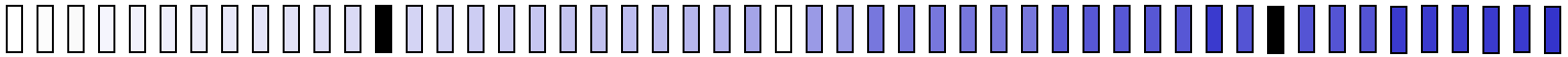
> lvg -f:**ici~128+ALL** -SC -SI

bioassay

bioassay|**bioassay**|<noun>|<base>|ici|1|

bioassay|**bioassay**|<noun>|<singular>|ici|1|

bioassay|**bioassays**|<noun>|<plural>|ici|1|

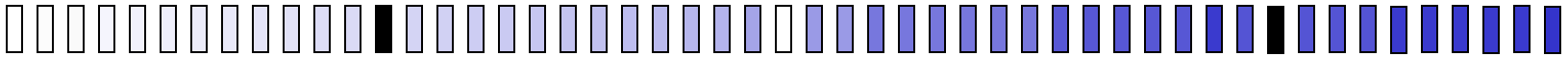


Lexical Tools: Inflect Simple

-f:is	Generate inflections noting simplified inflection tags

> lvg -f:**is**

cut|cut|1024|1|is|1|
cut|cutting|1024|16|is|1|
cut|cut|1024|32|is|1|
cut|cut|1024|64|is|1|
cut|cut|1024|128|is|1|
cut|cuts|1024|128|is|1|
cut|cut|128|1|is|1|
cut|cuts|128|8|is|1|



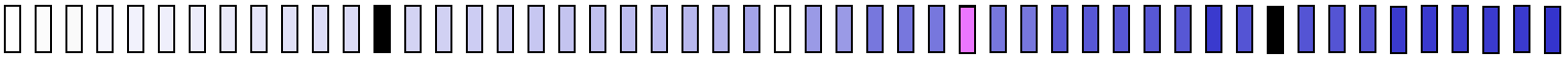
Lexical Tools: Uninflect by Term

-f:b	**Uninflect by term**

> lvg -f:**b** -SC -SI

left atria

left atria|**left atrium**|<noun>|<base>|b|1|



Lexical Tools: Derivations

-f:d	**Generate derivations**

> lvg -f:**d** -SC -SI

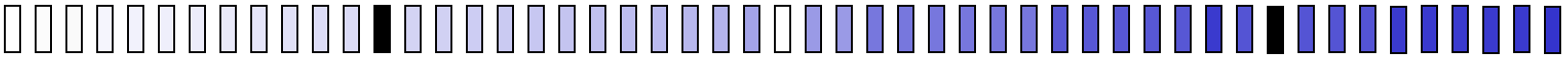
diagnostic

diagnostic|**diagnosis**|<noun>|<base>|d|1|

diagnostic|**diagnostics**|<noun>|<base>|d|1|

diagnostic|**diagnose**|<verb>|<base>|d|1|

diagnostic|**diagnostical**|<adj>|<base>|d|1|



Lexical Tools: Derivations

-f:dc~cats	**Generate derivations, filter by category**

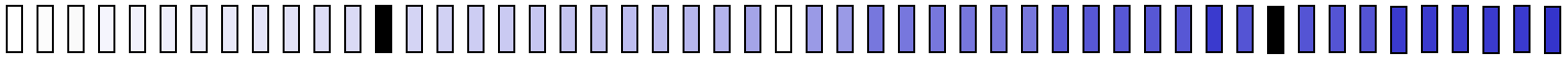
> lvg -f:**dc~129** -SC -SI

Reduce

reduce|**reducer**|<noun>|<base>|d|1|

reduce|**reduction**|<noun>|<base>|d|1|

reduce|**reducible**|<adj>|<base>|d|1|



Lexical Tools: Synonyms

-f:y	**Generate synonyms**

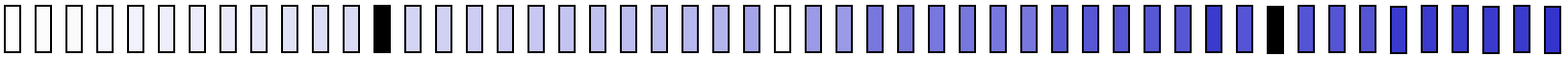
> lvg -f:**y** -SC -SI

kidney

kidney|**nephric**|<adj>|<base>|y|1|

kidney|**nephritic**|<adj>|<base>|y|1|

kidney|**renal**|<adj>|<base>|y|1|



Lexical Tools: Normalize (norm)

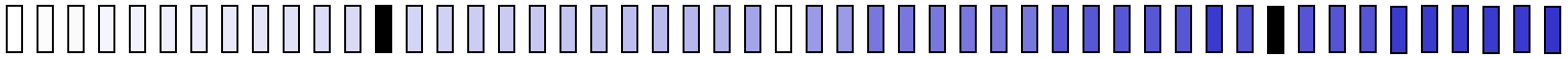
-f:N

Remove stop words, then remove genitives, then replace punctuation with spaces, then lowercase, then uninflect each word, then take each of the uninflected words, then word order sort.

> lvg -f:N

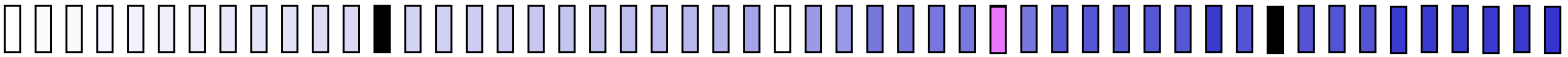
Syndrome, Dry Eyes

Syndrome, Dry Eyes|**dry eye syndrome**|2047|1|g+o+t+l+B+w|1|



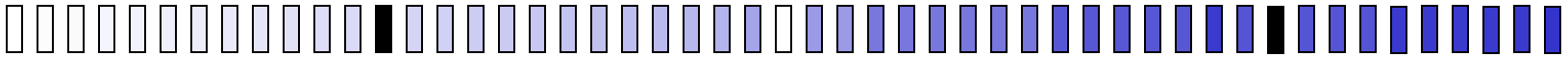
Lexical Tools: Installation

- Requirements
  - 2 gigabytes of space
  - Solaris/NT/Linux
  - Tar, gzip or WinZip
  - Minimum of 36 MB Memory
- Java (1.4) JREs included

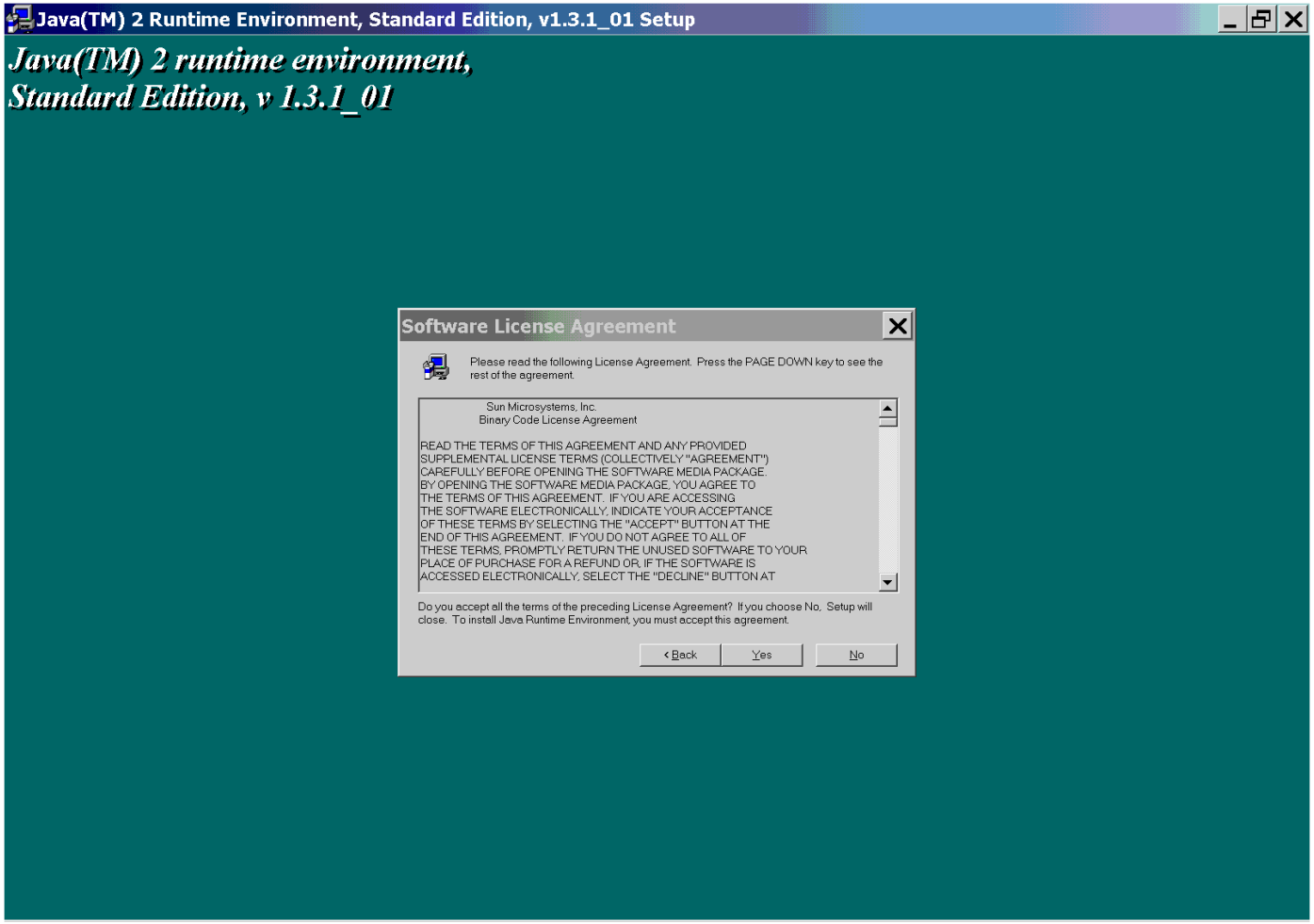


Lexical Tools: Installation

```
> ./install/bin/install_solaris_sparc.sh

> .\install\bin\install_win
```



Lexical Tools: Installation



Lexical Tools: Installation

Welcome to the Java Lexical Tools Installation!

Please read the ***installationNotes.html***
prior to invoking this script.

This script will configure the
_LVG_DIR_/data/config/lvg.properties
file.

This script will create configured
lvg, norm, luiNorm, and wordind
scripts in the _LVG_DIR_ directory.

This script, as an option, will configure
and load the lexical tools into a pre-existing
MySql database. This option is only available
if you already have a MySql database running.

~~~~~~

~~~~~~

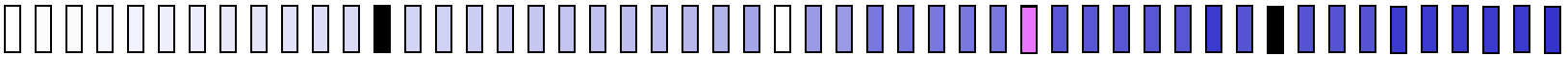
The Java Lexical tools use or (may use) the
following third party software packages:

- Instant DB from Enhydra.org
  (see http://instantdb.enhydra.org)
  This package is covered under an Enhydra
  Public License, which allows for redistribution
  and use with a "use notice".

- The installation script uses some GNU CYGWIN
  programs that have been distributed along with
  this script. These commands are only used when
  this installation is on a Windows platform.
  The commands that have been distributed
  include tar, gunzip, rm, find, hosthame,
  tee, and cmp.

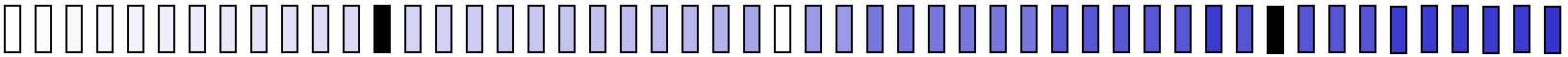
  The CYGWIN package is licensed under the
  GNU Public License. The entire CYGWIN package,
  along with the sources to these commands can be
  found at http://sources.redhat.com/cygwin/

Enhydra is a trademark of Lutris Technologies, Inc.



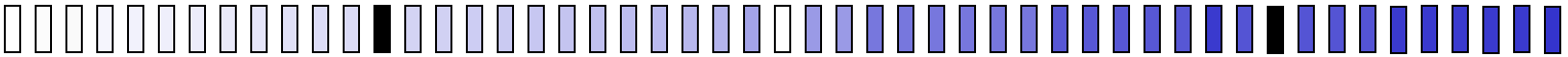
Lexical Tools: Installation

```
+----------------------Note---------------------------+
|Although we do not support this feature, and it      |
|is very likely that we will change the underlying    |
|implementation, we have found it useful for our      |
|applications to use an existing database system      |
|other than IDB. We often use Mysql as the underlying |
|database for our applications. Since we needed a     |
|way to get the lexical tools data into MySql, we     |
|figured that we would incorporate the MYSQL loader   |
|into this install.                                   |
|                                                     |
|This option requires that an existing mysql database |
|(version 3.23 or higher) have already been installed,|
|that you have database root access privileges,       |
|that the database has enough room to load this       |
|data, and that the database is running now.          |
+-----------------------------------------------------+

+-------------- Question -------------------+
|  Do you want to have the lexical tools use |
|  your mysql database [y/n]  [n]? n
```



Lexical Tools: Installation

```
+-----------------------------------------------+
| Verifying the installation ...                |
+-----------------------------------------------+

Enhydra InstantDB - Version 3.26

The Initial Developer of the Original Code is Lutris Technologies
Inc. Portions created by Lutris are Copyright (C) 1997-2001
Lutris Technologies, Inc.  All Rights Reserved.

Database sample is shutting down...

Database sample shutdown complete.
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```



Lexical Tools: Installation

~~~~~~

Congratulations!

~~~~~~

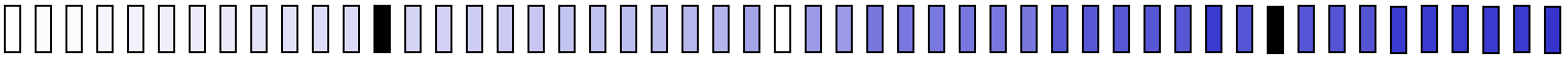
This script has completed configuring the java lexical tools. You may invoke these tools from a command line. These tools are found in the __LVG_DIR__/bin directory.
You can add this __LVG_DIR__/bin path to your $PATH environment variable. This would enable you to find and run these tools from any location.
In UNIX, this would be done by adding this path to your ~/.cshrc or ~/.profile startup script.

In Windows, this would be done by appending this path to the PATH variable from the control panel/System/Environment variables menus.

~~~~~~

The lexical tools are ready to be used!

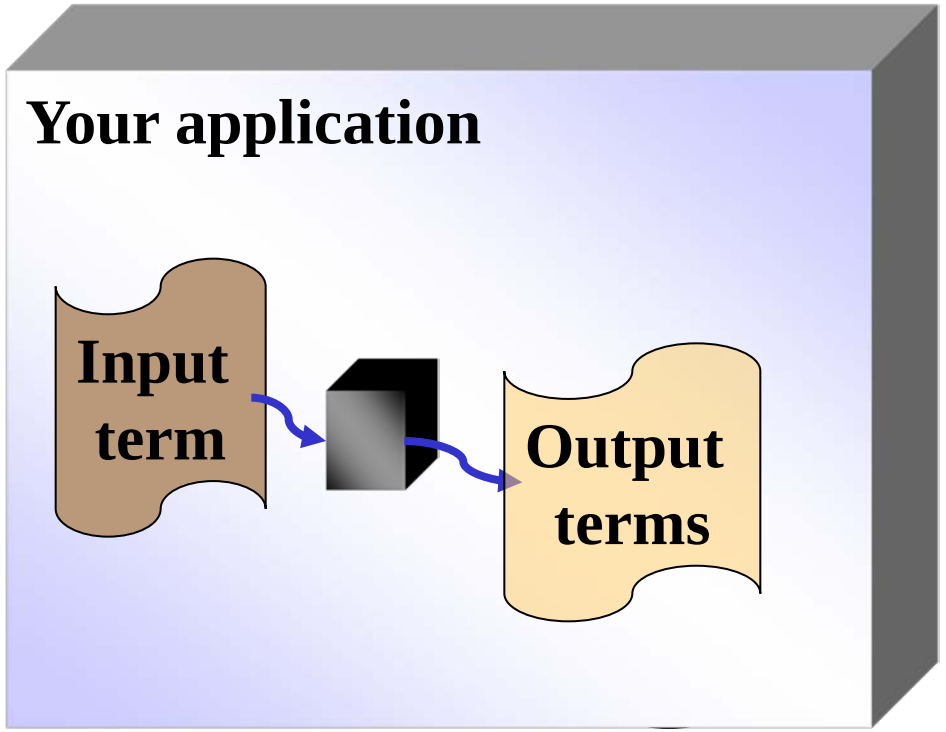
~~~~~~



Lexical Tools:

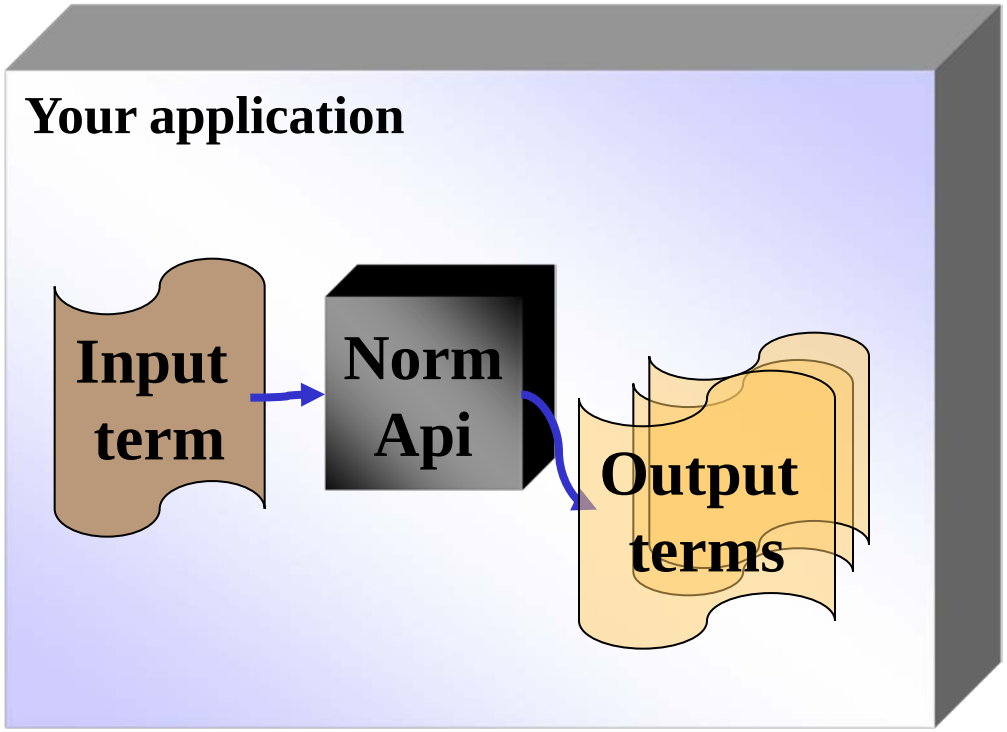
Embedding These Tools into Your Application

- Classpath
- NormApi()
- LvgCmdApi()
- LexItem
- LvgLexItemApi()



Lexical Tools:

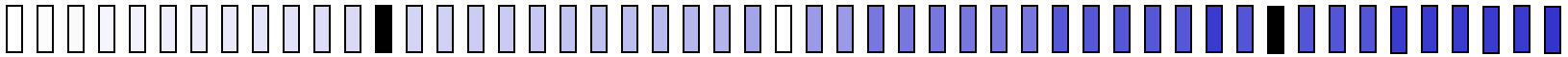
Embedding Norm into Your Application



Lexical Tools:

Embedding These Tools into Your Application

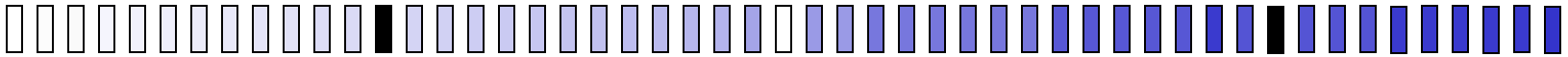
CLASSPATH = ${CLASSPATH}:
${LVG_DIR}:
${LVG_DIR}/lib/*lvg2003dist.jar:*



Lexical Tools:

Embedding Norm into Your Application

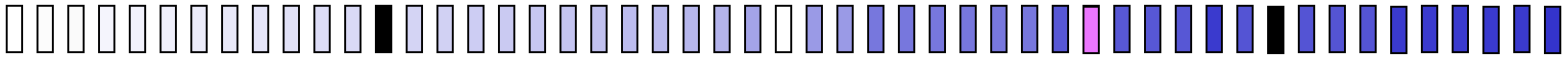
```
import gov.nih.nlm.nls.lvg.Api.*;


NormApi       normalize = new NormApi();
String          input2Norm = null;
Vector  outputFromNorm = null;
```



Lexical Tools:

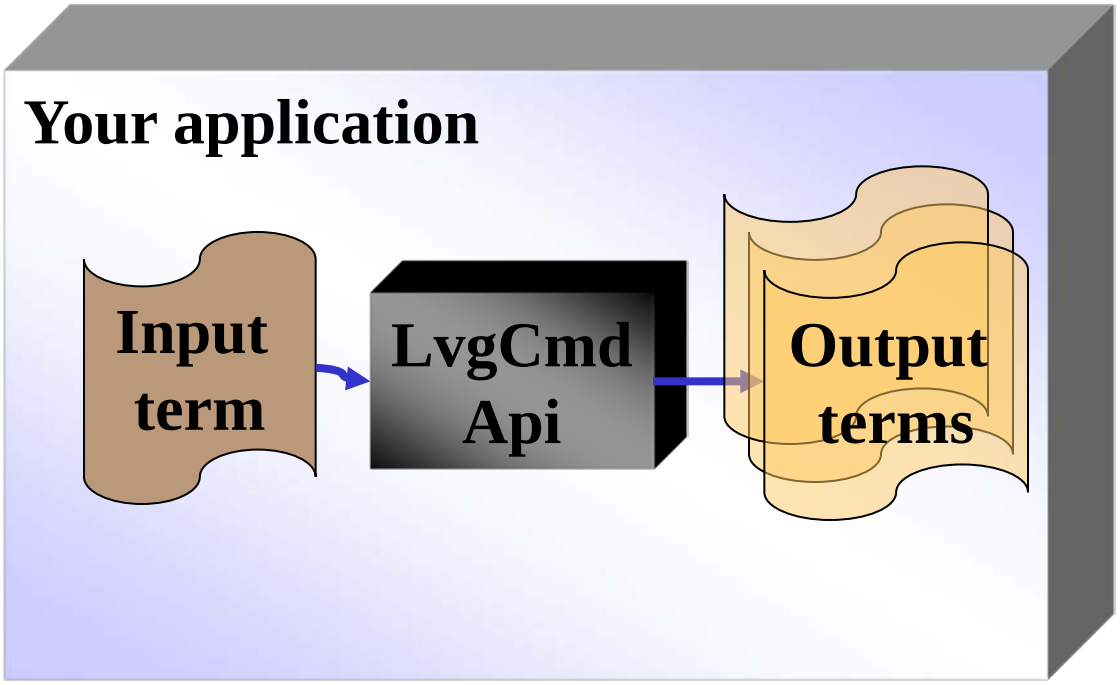
Embedding Norm into Your Application

```
while ( (input2Norm = stdIn.readLine() ) != null ) {

   outputFromNorm= normalize.Mutate(input2Norm);

   for ( int i = 0; i < outputFromNorm.size(); i++ ) {

       System.out.println((String) outputFromNorm.get(i));
    }
}
normalize.CleanUp();
```



Lexical Tools:

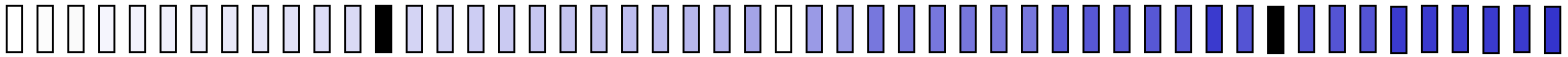
Embedding Lvg into Your Application



Lexical Tools:

Embedding Lvg into Your Application

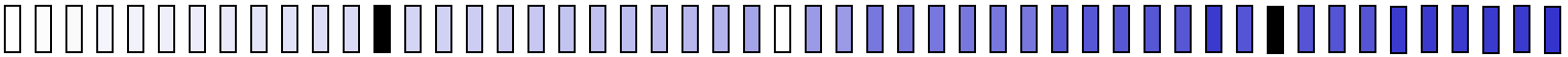
```
import gov.nih.nlm.nls.lvg.Api.*;

LvgCmdApi lvgApi = new LvgCmdApi("-f:b –CR:o –SC –SI");
String          input2Lvg = null;
Vector   outputFromLvg = null;
```



Lexical Tools:

Embedding Lvg into Your Application

```
while ( (input2Lvg = stdIn.readLine() ) != null ) {
   outputFromLvg= lvgApi.Mutate(input2Lvg);
    for ( int i = 0; i < outputFromLvg.size(); i++ ) {
        System.out.println((String) outputFromLvg.get(i));
    }
}
lvgApi.CleanUp();
```

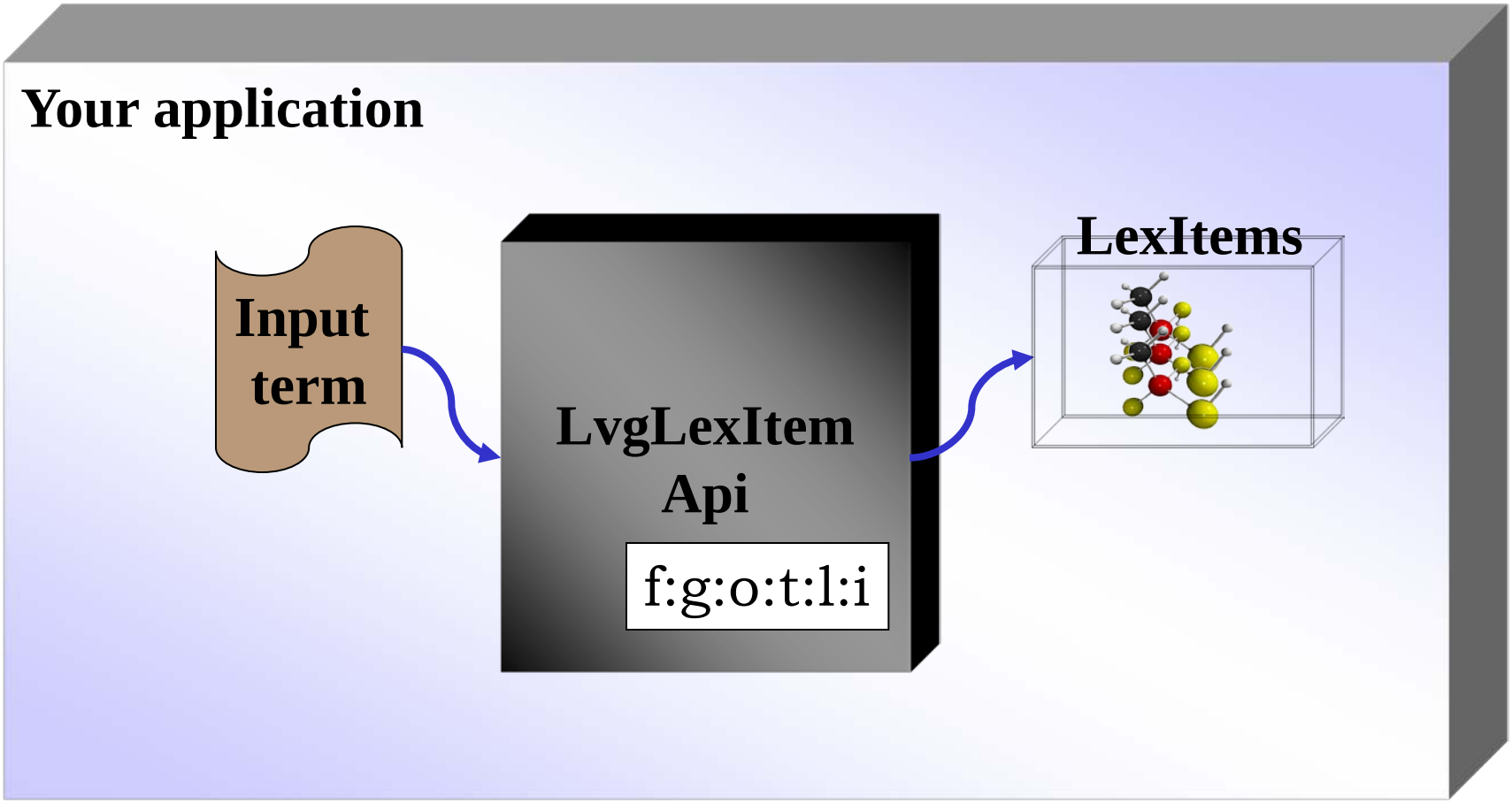


Lexical Tools:

Embedding Lvg into Your Application

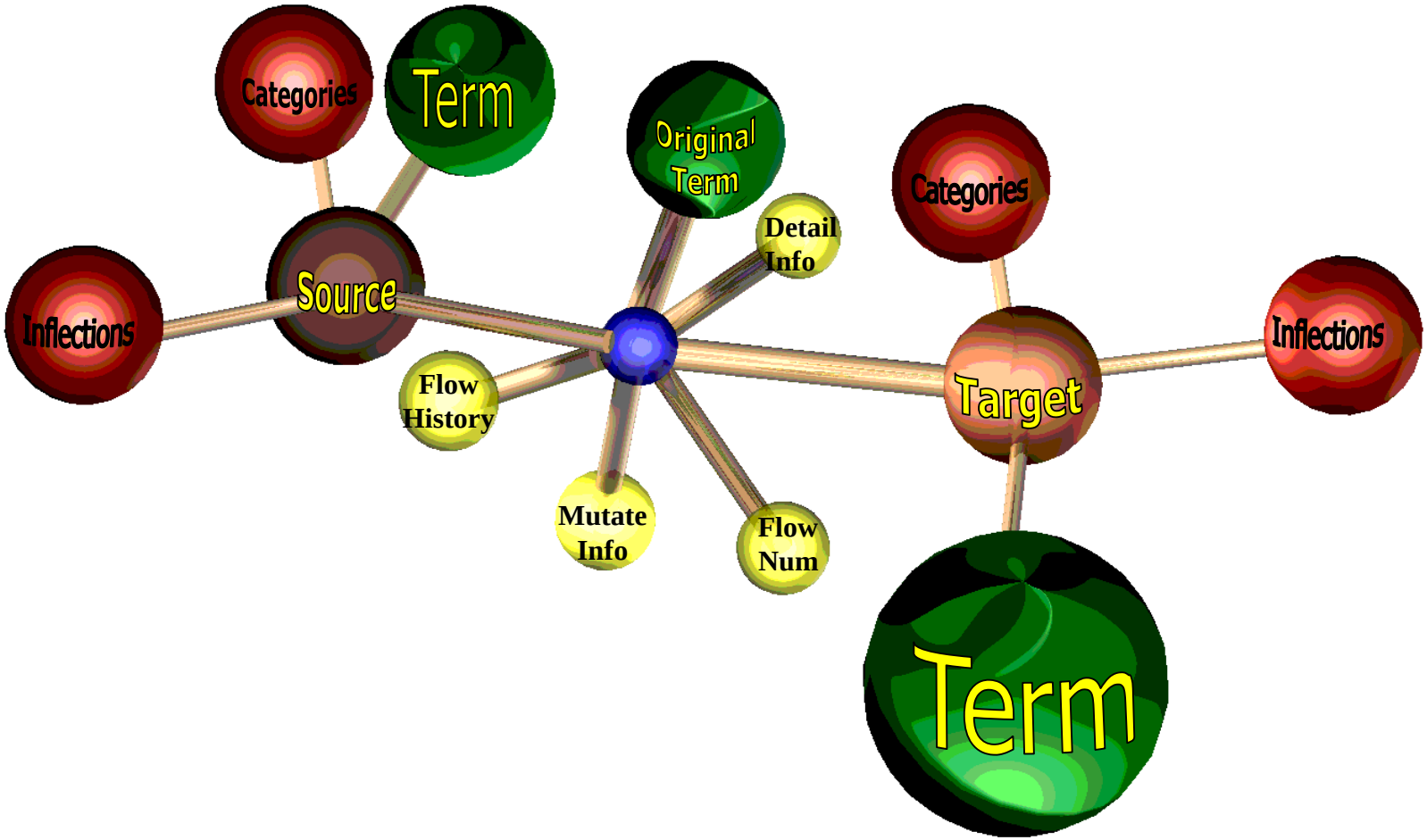
Lexical Tools:

Embedding Lvg into Your Application



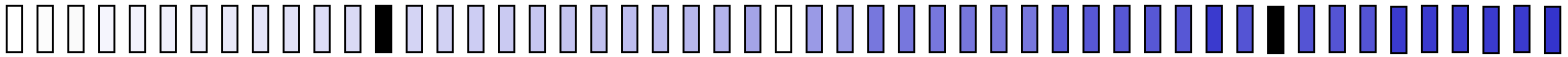
Lexical Tools:

The LexItem Class



Lexical Tools:
Embedding Lvg into Your Application

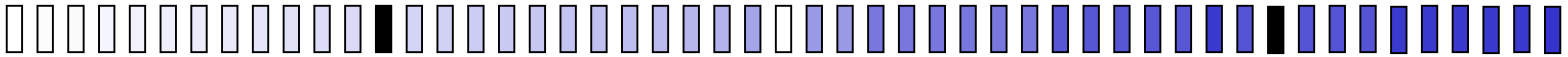
```
import gov.nih.nlm.nls.lvg.Api.*;
import gov.nih.nlm.nls.lvg.Lib.*;

LvgCmdApi    lvgApi = new LvgLexItemApi("-f:g:o:t:l:i");
String          input2Lvg = null;
Vector outputFromLvg = null;
LexItem         aLexItem = null;
```



Lexical Tools:

Embedding Lvg into Your Application

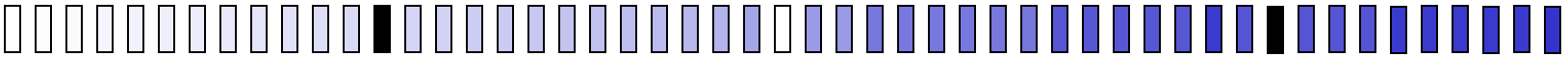
```
while ( (input2Lvg = stdIn.readLine() ) != null ) {

   outputFromLvg=lvgApi.MutateLexItem(input2Lvg);

    for ( int i = 0; i < outputFromLvg.size(); i++ ) {

       aLexItem = (LexItem) outputFromLvg.get(i);
       System.out.println(aLexItem.GetSourceTerm() + "|" +
             aLexItem.GetTargetTerm() + "|" +
             aLexItem.GetTargetCategory().GetName()+ "|" +
             aLexItem.GetTargetInflection().GetName() + "|" +
             aLexItem.GetTargetInflection().GetValue() );
    }
}
lvgApi.CleanUp();
```



Lexical Tools:

Java API Documentation

Packages	
gov.nih.nlm.nls.lvg.Api	This package provides APIs
gov.nih.nlm.nls. lvg.CmdLineSyntax	Provides Java classes necessary to create a command line system.
gov.nih.nlm.nls. lvg.Db	Provides a higher level interface to LVG database.
gov.nih.nlm.nls. lvg.Flows	This package provides API of all Lvg flow components.
gov.nih.nlm.nls. lvg.Lib	Contains LVG general library classes of BitMaskBase, Category, Gender, Inflection, Configuration, CombineRecords, OutputFilter, Flow, GlobalBehavior, LexItem, LexItemComparator
gov.nih.nlm.nls. lvg.Trie	Provides the classes necessary to generate inflections, uninflections, and derivations using LVG rules tries
gov.nih.nlm.nls. lvg.Util	Contains LVG general utility classes of comparators, Bit operations, Case, In, Out, Strip operations, token operation, Char, Str, and Word.



Lexical Resources

UMLS Knowledge Source Server (UMLSKS)

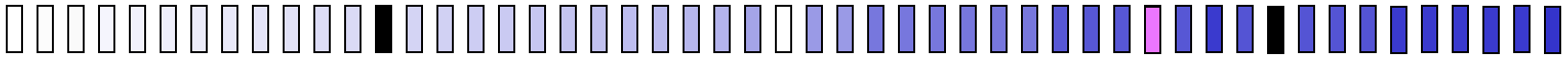
UMLSKS Version 2.1 UMLS Releases: 2002 2002AB

Metathesaurus Semantic Network SPECIALIST Lexicon

Documentation Resources Views/Profiles Logout

The following resources are provided as part of the UMLSKS suite of tools.

Download UMLS Knowledge Sources	Source files for the Metathesaurus, Semantic Network and SPECIALIST Lexicon.
Metathesaurus Resources	Tools and resources for the UMLS Metathesaurus.
NLP and Lexical Resources	Generic configurable Natural Language Processing tools and utilities for manipulating lexical data.



References

McCray AT, Loane RF, Browne AC, Bangalore AK. Terminology Issues in User Access to Web-based Medical Information. Proc AMIA Symp. 1998;:107-111

Divita G, Browne AC, Rindflesch T. Evaluating Lexical Variant Generation to Improve Information Retrieval, Proc AMIA Symp. 1998;:775-9

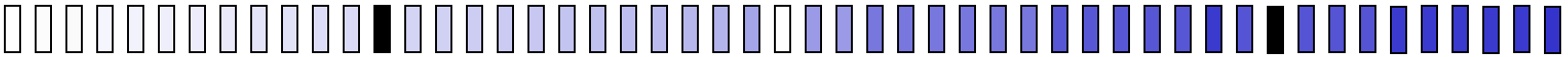
McCray AT, Browne AC. Discovering the modifiers in a terminology data set. Proc AMIA Symp. 1998;:780-4

McCray AT. The nature of lexical knowledge. Methods Inf Med. 1998 Nov;37(4-5):353-60.

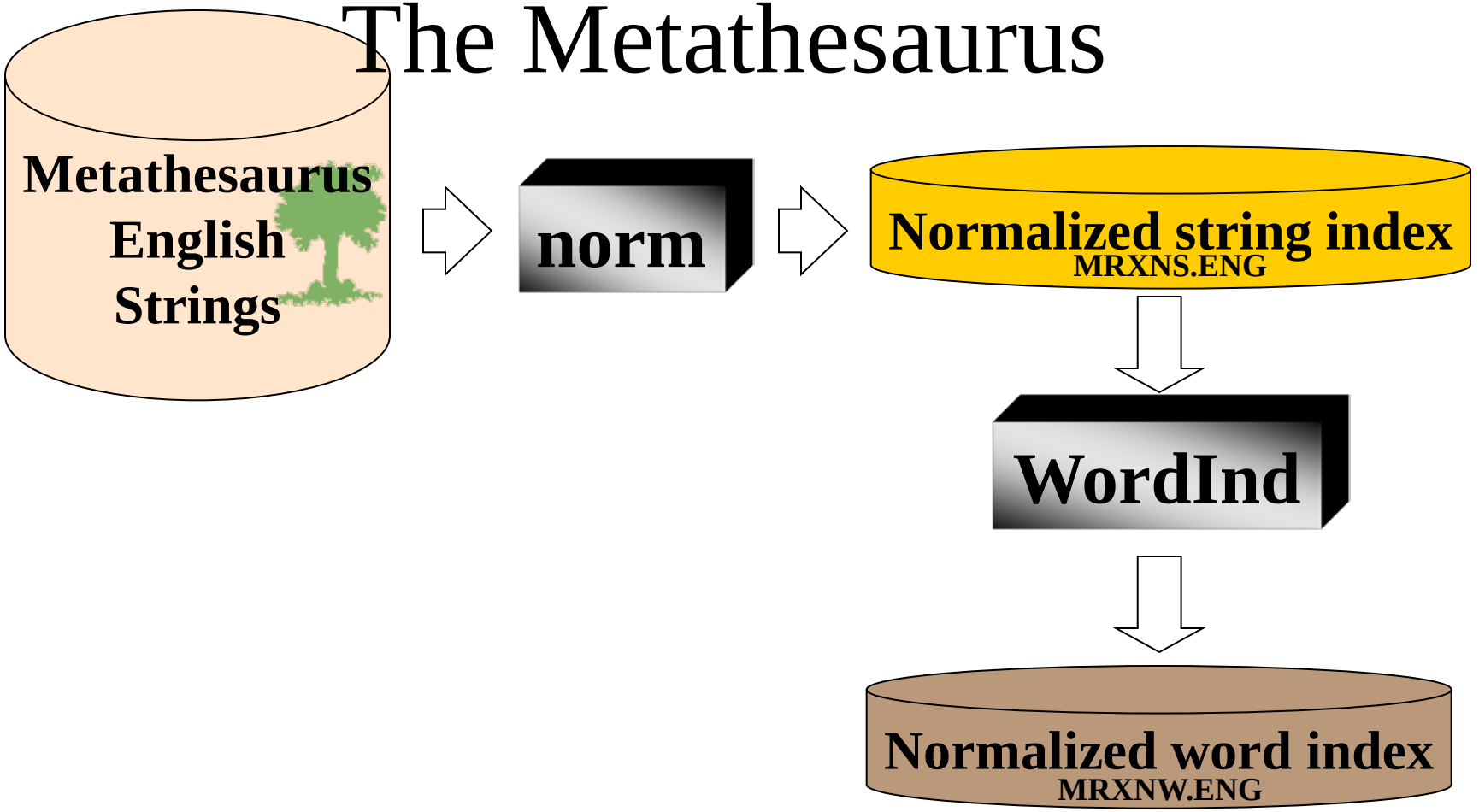
McCray AT, Cheh ML, Bangalore AK, Rafei K, Razi AM, Divita G, Stavri PZ. Conducting the NLM/AHCPR Large Scale Vocabulary Test: a distributed Internet-based experiment. Proc AMIA Annu Fall Symp. 1997;:560-4.

McCray AT, Razi, AM, Bangalore AK, Browne AC, Stavri PZ. The UMLS Knowledge Source Server: A versatile Internet-based research tool. Proc AMIA Symp. 1996;164-8.

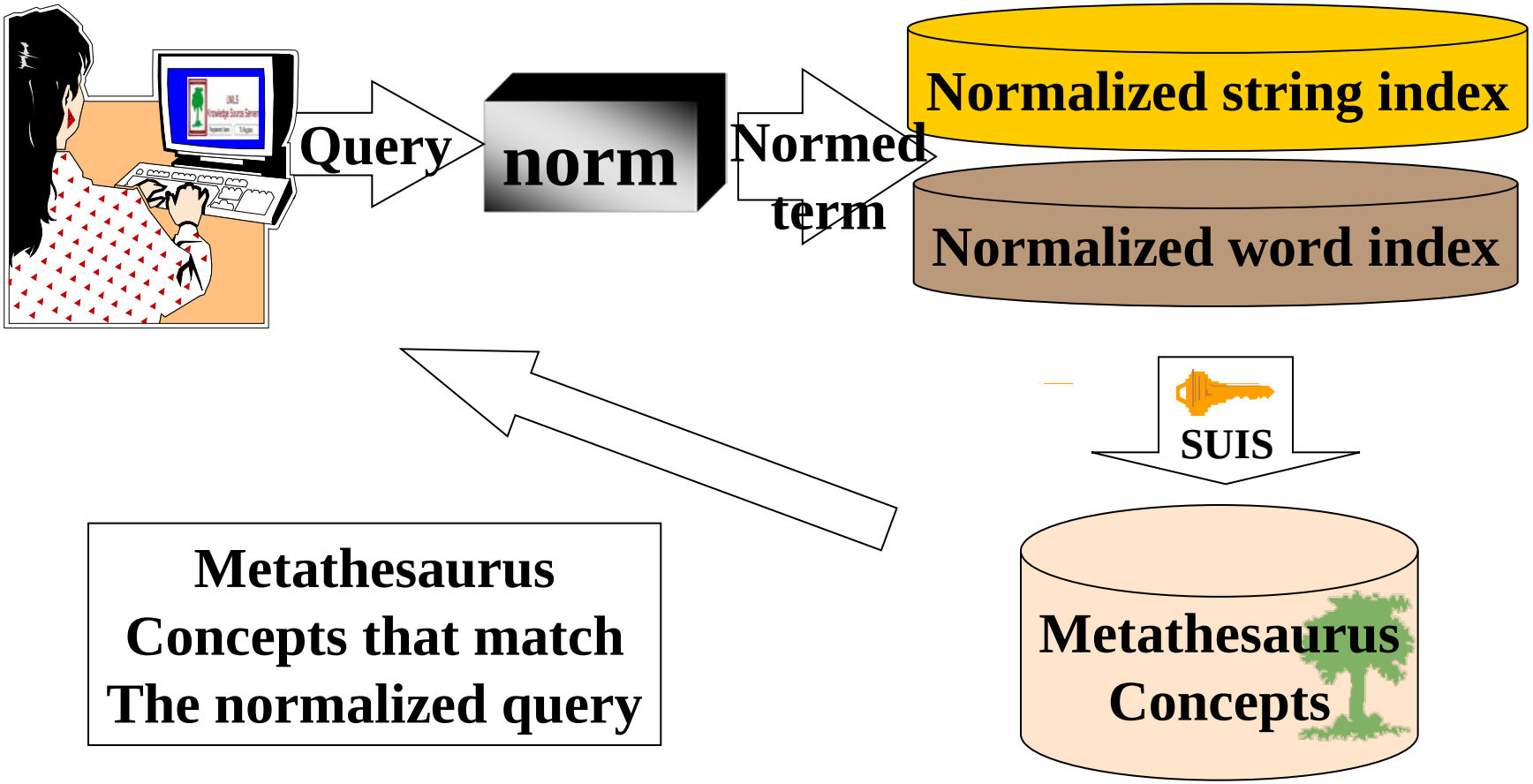
McCray AT, Srinivasan S, Browne AC. Lexical methods for managing variation in biomedical terminologies. Proc Annu Symp Comput Appl Med Care. 1994;:235-9.



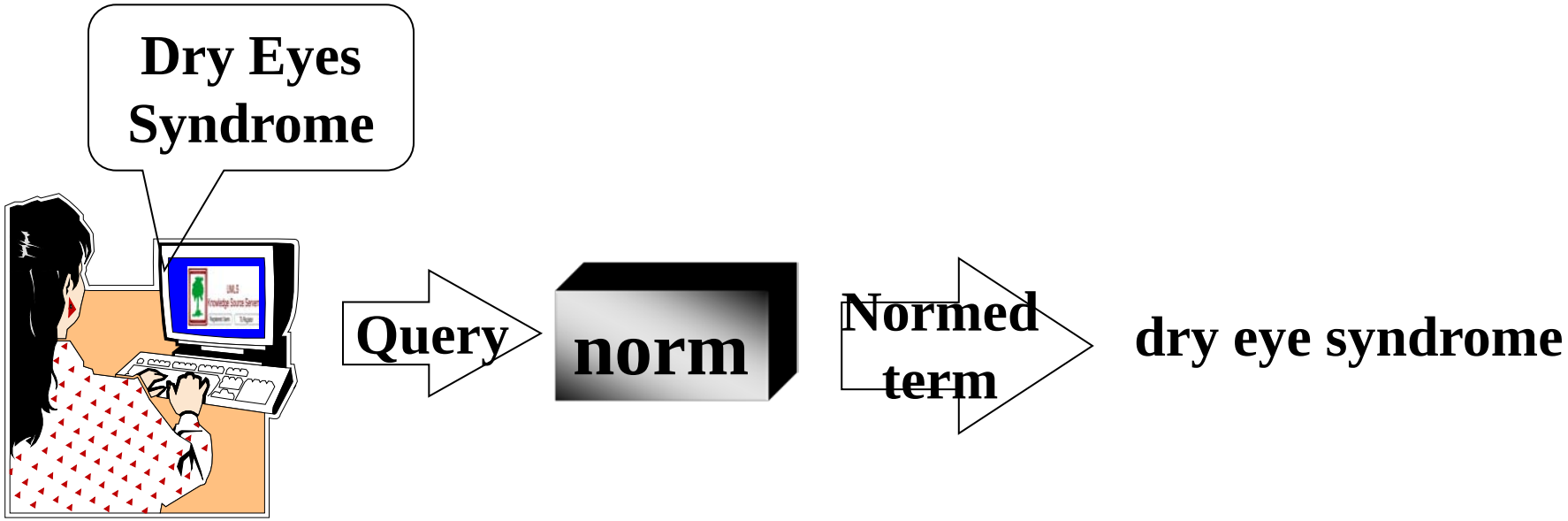
Using The Lexical Tools with The Metathesaurus



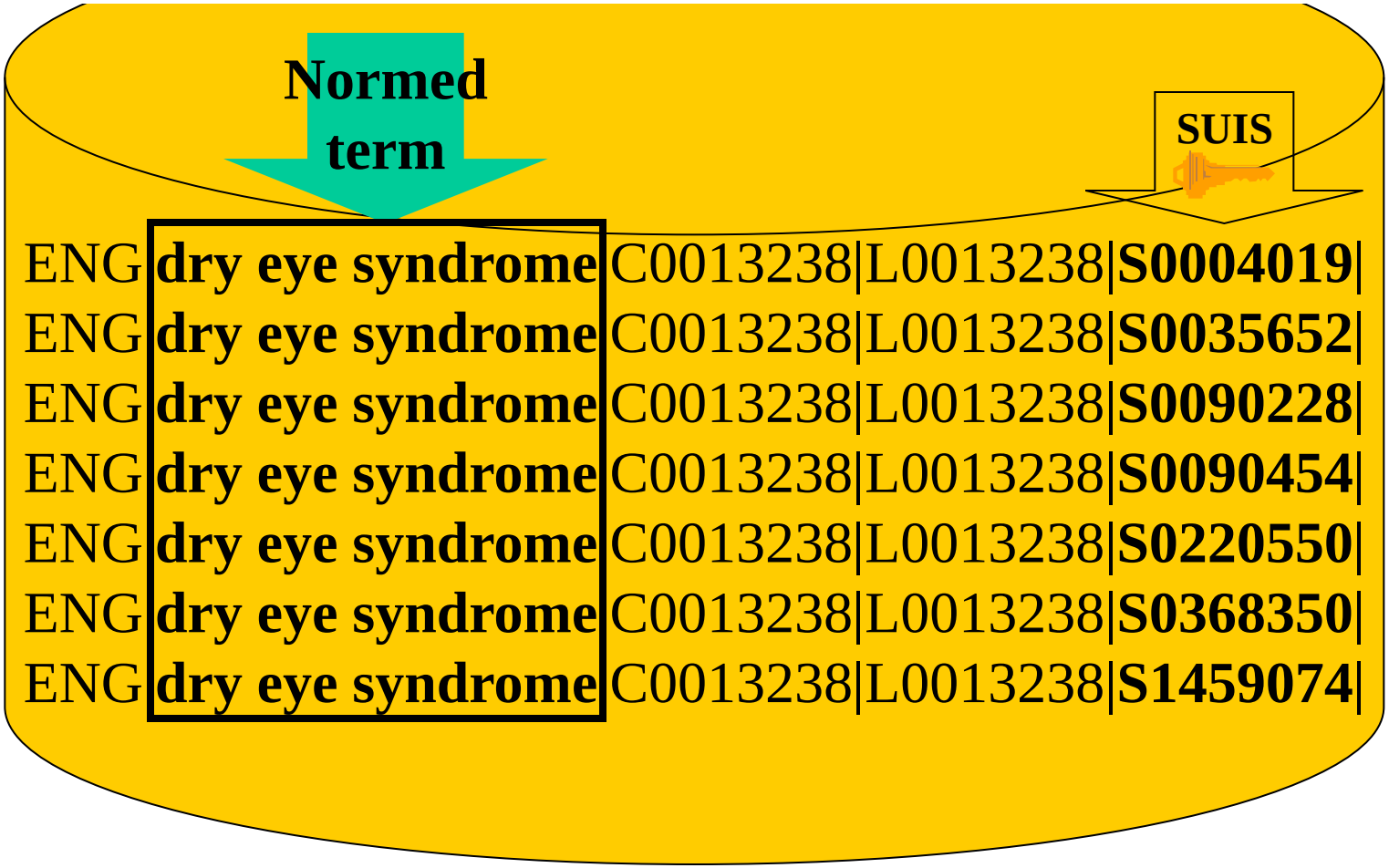
Using The Lexical Tools with The Metathesaurus



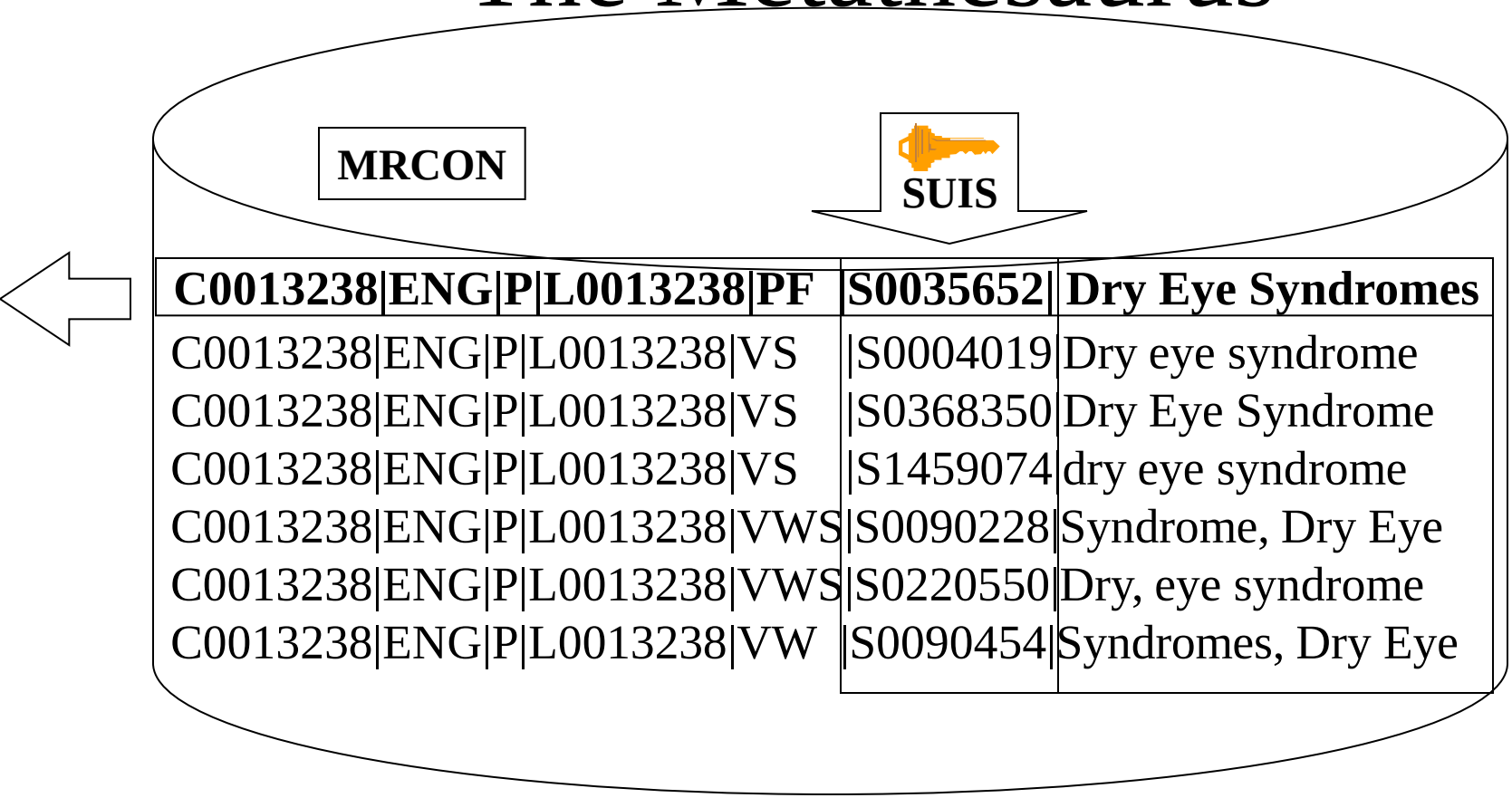
Using The Lexical Tools with The Metathesaurus



Using The Lexical Tools with The Metathesaurus



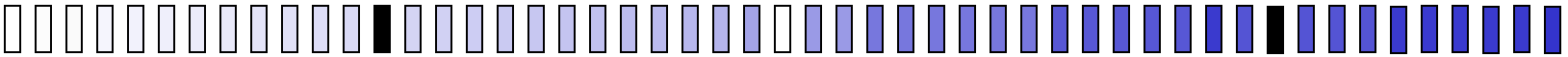
Using The Lexical Tools with The Metathesaurus



C0013238|ENG|P|L0013238|PF	S0035652|	Dry Eye Syndromes
C0013238|ENG|P|L0013238|VS	|S0004019	Dry eye syndrome
C0013238|ENG|P|L0013238|VS	|S0368350	Dry Eye Syndrome
C0013238|ENG|P|L0013238|VS	|S1459074	dry eye syndrome
C0013238|ENG|P|L0013238|VWS	|S0090228	Syndrome, Dry Eye
C0013238|ENG|P|L0013238|VWS	|S0220550	Dry, eye syndrome
C0013238|ENG|P|L0013238|VW	|S0090454	Syndromes, Dry Eye

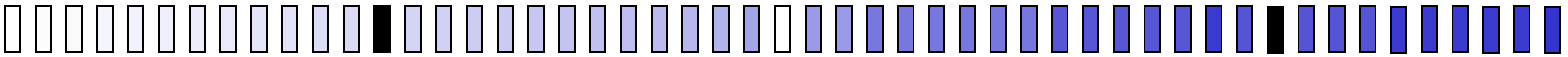
Building an Index Using The Lexical Tools

- Can we build a tool that increases recall?
- Can we build a tool that increases precision?



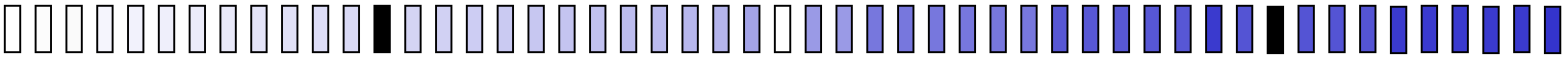
Building an Index Using The Lexical Tools

- Can we build a tool that increases precision?
  - Case
  - Constrain by part of speech
  - Filter to the lexicon

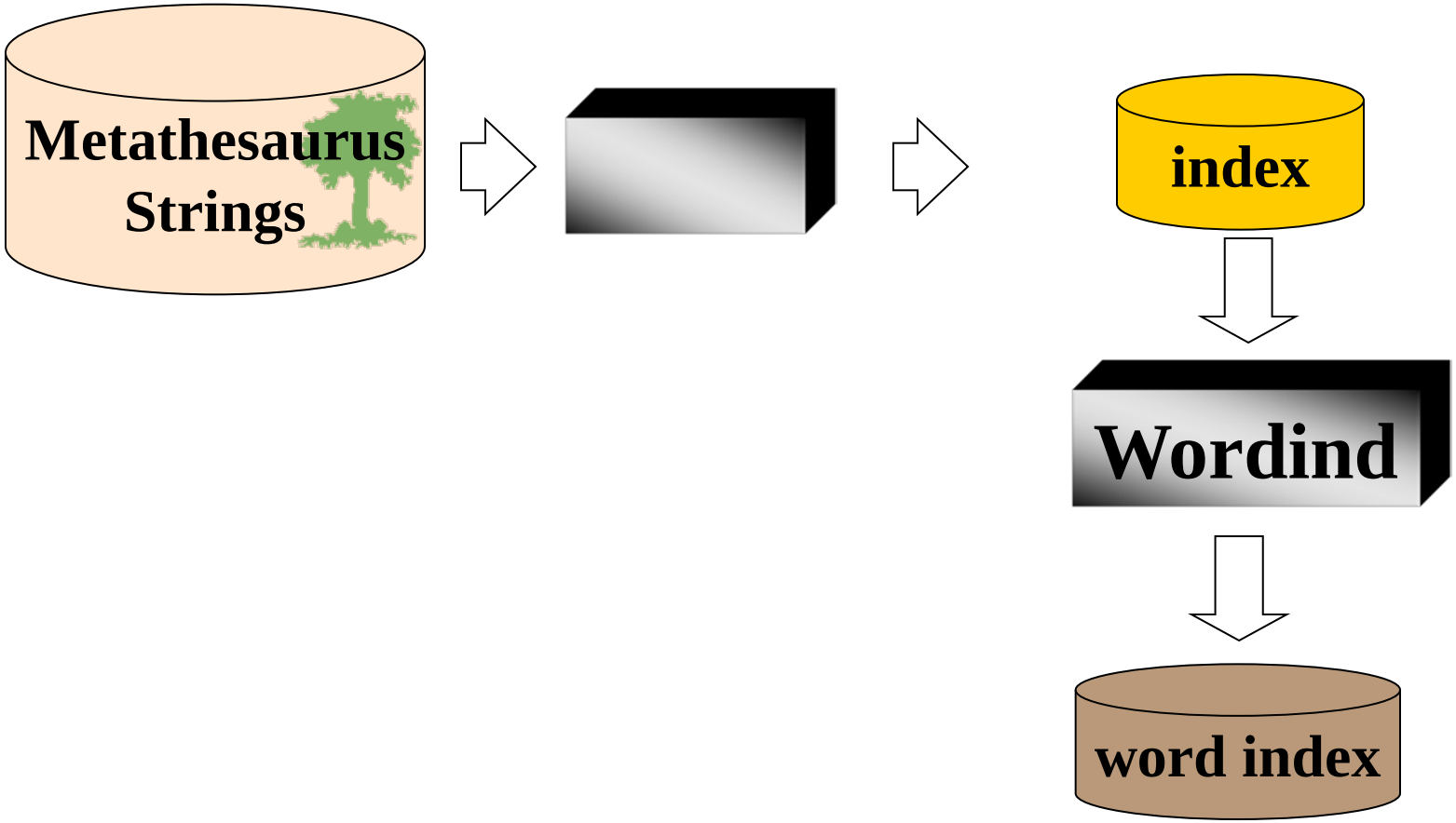


Building an Index Using The Lexical Tools

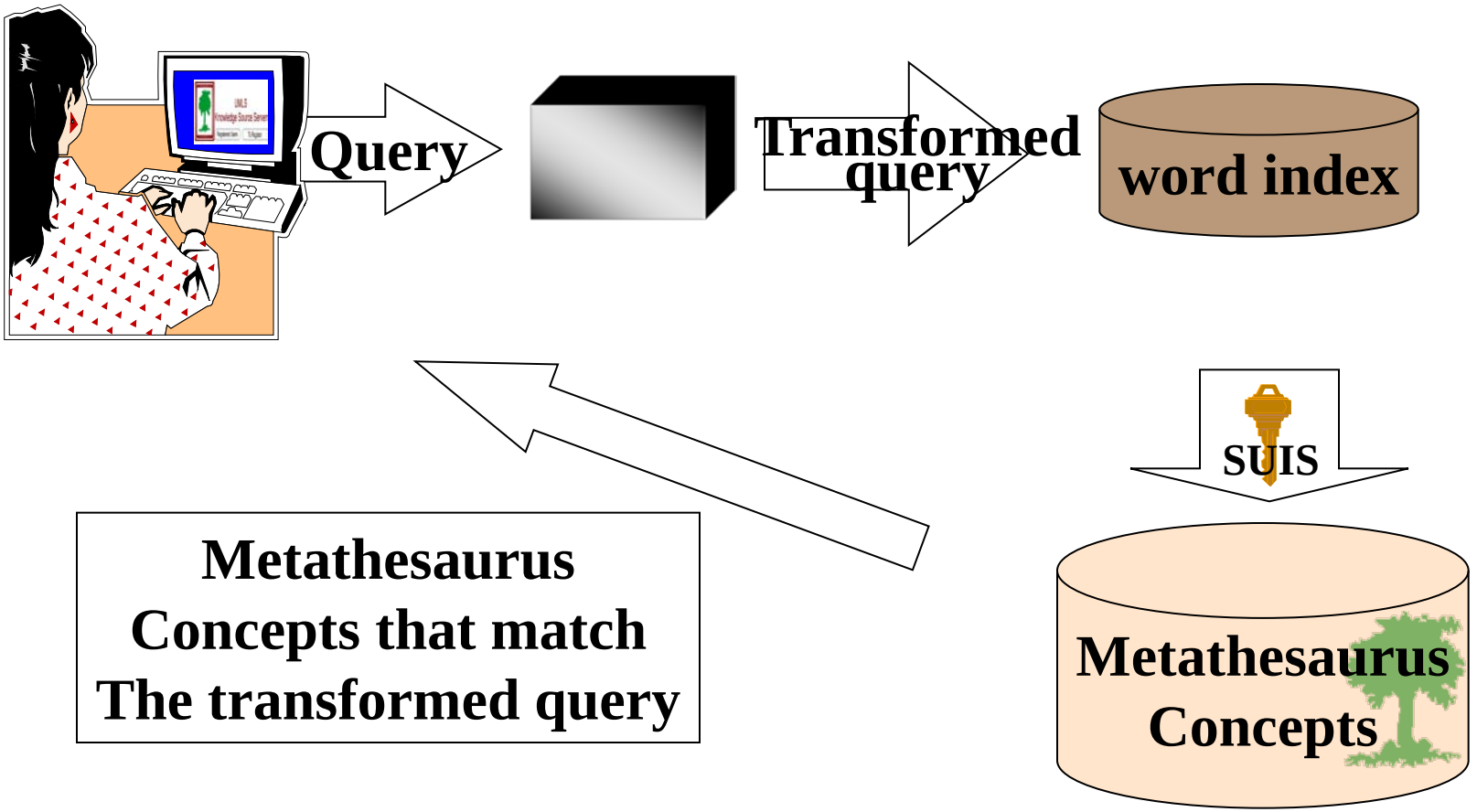
- Can we a tool that increases recall?
  - Include
    - synonyms
    - derivations
    - acronyms and their expansions
    - spelling variants



Building an Index Using The Lexical Tools



Building an Index Using The Lexical Tools



Lexical Tools:
2003

- Performance issues addressed
- Graphic User Interface
  - Lexical GUI tool
  - On-line Web based tool
- New flow components
  - Simplify Inflection flows –f:Si, –f:is
- Additional API
  - configuration file parameter in constructor