

**800**  
A N N I  
1222-2022



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

# Indexing Hands-on

## Search Engines

Master Degree in Computer Engineering

Master Degree in Data Science

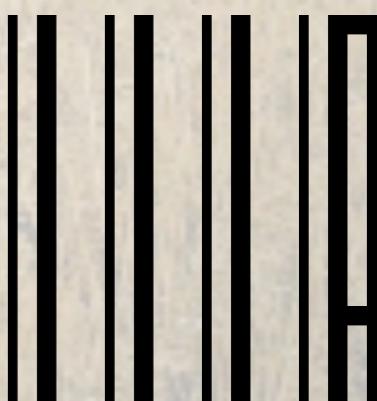
Academic Year 2023/2024

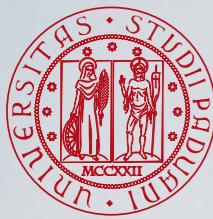


DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE

**Nicola Ferro**

Intelligent Interactive Information Access (IIIA) Hub  
Department of Information Engineering  
University of Padua



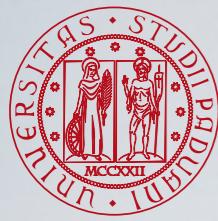


# Outline



- Indexing Options in Lucene
- Inside Lucene Indexes
- Lucene Codecs
- Extracting Information from a Lucene Index

# Indexing Options in Lucene

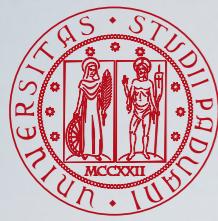


# Inverted Index: Just Documents

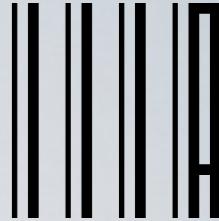


- It allows for retrieving documents containing any or all query terms
  - small marsupial → **intersection** if searching for all the terms;
  - **union** if searching for any terms
- No ranking possible
- No proximity match/phrase match possible

| Dictionary        | Posting List | Dictionary     | Posting List |
|-------------------|--------------|----------------|--------------|
| a                 | <u>1</u> 3   | muscular       | <u>2</u>     |
| about             | <u>1</u>     | native         | <u>2</u>     |
| and               | <u>3</u>     | of             | <u>1</u> 3   |
| approach          | <u>3</u>     | on             | <u>3</u>     |
| are               | <u>2</u>     | only           | <u>1</u>     |
| australia         | <u>2</u> 3   | particularly   | <u>3</u>     |
| cat               | <u>1</u>     | people         | <u>3</u>     |
| closely           | <u>3</u>     | population     | <u>3</u>     |
| commonly          | <u>3</u>     | prevalent      | <u>3</u>     |
| domestic          | <u>1</u>     | quadrupedal    | <u>2</u>     |
| exists            | <u>3</u>     | <b>quokka</b>  | <u>1</u>     |
| fear              | <u>3</u>     | <b>quokkas</b> | <u>3</u>     |
| genus             | <u>1</u>     | rottnest       | <u>3</u>     |
| have              | <u>3</u>     | setonix        | <u>1</u>     |
| humans            | <u>3</u>     | <b>short</b>   | <u>2</u>     |
| in                | <u>3</u>     | size           | <u>1</u>     |
| is                | <u>1</u>     | <b>small</b>   | <u>1</u> 2   |
| island            | <u>3</u>     | that           | <u>2</u>     |
| <b>legged</b>     | <u>2</u>     | the            | <u>1</u>     |
| little            | <u>3</u>     | to             | <u>2</u>     |
| <b>marsupial</b>  | <u>1</u>     | where          | <u>3</u>     |
| <b>marsupials</b> | <u>2</u>     | wombats        | <u>2</u>     |
| member            | <u>1</u>     |                |              |



# Inverted Index: Just Documents



Luke: Lucene Toolbox Project - v8.8.1

File Tools Help

Overview Documents Search Analysis Commits Logs

Index Path: /Users/ferro/Documents/progetti/software/search-engines/se-unipd/hello-ir/experiment/index

Number of Fields: 2

Number of Documents: 3

Number of Terms: 48

Has deletions? / Optimized?: No / Yes

Index Version: 8

Index Format: Lucene 8.6 or later

Directory implementation: org.apache.lucene.store.MMapDirectory

Currently opened commit point: segments\_2 (generation=2, segs=1)

Current commit user data: {}

Select a field from the list below, and press button to view top terms in the field.

Available fields and term counts per field:

| Name | Term count | %       |
|------|------------|---------|
| body | 45         | 93.75 % |
| id   | 3          | 6.25 %  |

Selected field: body

Show top terms >

Num of terms: 68

Top ranking terms: (Double-click for more options.)

| Rank | Freq | Text         |
|------|------|--------------|
| 1    | 2    | small        |
| 2    | 2    | of           |
| 3    | 2    | australia    |
| 4    | 2    | a            |
| 5    | 1    | wombats      |
| 6    | 1    | where        |
| 7    | 1    | to           |
| 8    | 1    | the          |
| 9    | 1    | that         |
| 10   | 1    | size         |
| 11   | 1    | short        |
| 12   | 1    | setonix      |
| 13   | 1    | rottnest     |
| 14   | 1    | quokkas      |
| 15   | 1    | quokka       |
| 16   | 1    | quadrupedal  |
| 17   | 1    | prevalent    |
| 18   | 1    | population   |
| 19   | 1    | people       |
| 20   | 1    | particularly |
| 21   | 1    | only         |
| 22   | 1    | on           |
| 23   | 1    | native       |
| 24   | 1    | muscular     |
| 25   | 1    | member       |
| 26   | 1    | marsupials   |
| 27   | 1    | marsupial    |
| 28   | 1    | little       |
| 29   | 1    | legged       |
| 30   | 1    | island       |
| 31   | 1    | is           |
| 32   | 1    | in           |
| 33   | 1    | humans       |
| 34   | 1    | have         |
| 35   | 1    | genus        |
| 36   | 1    | fear         |
| 37   | 1    | exists       |
| 38   | 1    | domestic     |
| 39   | 1    | commonly     |
| 40   | 1    | closely      |
| 41   | 1    | cat          |
| 42   | 1    | are          |
| 43   | 1    | approach     |
| 44   | 1    | and          |
| 45   | 1    | about        |

Luke: Lucene Toolbox Project - v8.8.1

File Tools Help

Overview Documents Search Analysis Commits Logs

Browse terms in field: body

Browse documents by term: small

« First Term small Next

Hint: Edit the text field above and press Enter to seek to arbitrary terms.

Position Offsets Payload

11

Browse documents by Doc # 0 in 3 docs

Copy values More like this Add document

(Select a row and double-click for more options.)

(To copy all or arbitrary field value(s), unselect all rows or select row(s), and click 'Copy values' button.)

| Field | Flags <a href="#">Help</a> | Norm | Value   |
|-------|----------------------------|------|---|
| id    | Idfp-N-S-----              | 1    | d1  |
| body  | Idfp-N-S-----              | 20   | The quokka, the only member of the genus Setonix, is a small marsupial a... |

Luke: Lucene Toolbox Project - v8.8.1

File Tools Help

Overview Documents Search Analysis Commits Logs

Browse terms in field: body

Browse documents by term: small

« First Term small Next

Hint: Edit the text field above and press Enter to seek to arbitrary terms.

Position Offsets Payload

2

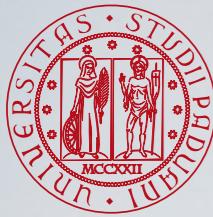
Browse documents by Doc # 1 in 3 docs

Copy values More like this Add document

(Select a row and double-click for more options.)

(To copy all or arbitrary field value(s), unselect all rows or select row(s), and click 'Copy values' button.)

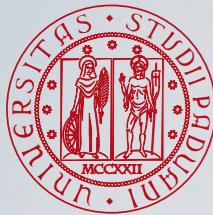
| Field | Flags <a href="#">Help</a> | Norm | Value   |
|-------|----------------------------|------|---|
| id    | Idfp-N-S-----              | 1    | d2  |
| body  | Idfp-N-S-----              | 13   | Wombats are small, short-legged, muscular quadrupedal marsupials that ar... |



# Inverted Index: Documents and Positions

- It allows for ranking documents by their relevance
- Proximity match/phrase match possible
  - “small marsupial”
  - find **small** within 5 words of **marsupials**

| Dictionary        | Posting List            | Dictionary   | Posting List             |
|-------------------|-------------------------|--------------|--------------------------|
| a                 | <u>1:10,17</u>          | muscular     | <u>2:5</u>               |
| about             | <u>1:13</u>             | native       | <u>2:10</u>              |
| and               | <u>3:6</u>              | of           | <u>1:5,16</u> <u>3:4</u> |
| approach          | <u>3:8</u>              | on           | <u>3:12</u>              |
| are               | <u>2:1,9</u>            | only         | <u>1:3</u>               |
| australia         | <u>2:12</u> <u>3:16</u> | particularly | <u>3:11</u>              |
| cat               | <u>1:19</u>             | people       | <u>3:9</u>               |
| closely           | <u>3:10</u>             | population   | <u>3:20</u>              |
| commonly          | <u>3:7</u>              | prevalent    | <u>3:19</u>              |
| domestic          | <u>1:18</u>             | quadrupedal  | <u>2:6</u>               |
| exists            | <u>3:21</u>             | quokka       | <u>1:1</u>               |
| fear              | <u>3:3</u>              | quokkas      | <u>3:0</u>               |
| genus             | <u>1:7</u>              | rottnest     | <u>3:13</u>              |
| have              | <u>3:1</u>              | setonix      | <u>1:8</u>               |
| humans            | <u>3:5</u>              | short        | <u>2:3</u>               |
| in                | <u>3:15</u>             | size         | <u>1:15</u>              |
| is                | <u>1:9</u>              | <b>small</b> | <u>1:11</u> <u>2:2</u>   |
| island            | <u>3:14</u>             | that         | <u>2:8</u>               |
| legged            | <u>2:4</u>              | the          | <u>1:0,2,6,14</u>        |
| little            | <u>3:2</u>              | to           | <u>2:11</u>              |
| <b>marsupial</b>  | <u>1:12</u>             | where        | <u>3:17</u>              |
| <b>marsupials</b> | <u>2:7</u>              | wombats      | <u>2:0</u>               |
| member            | <u>1:4</u>              |              |                          |



# Inverted Index: Documents and Positions

Luke: Lucene Toolbox Project - v8.8.1

File Tools Help

Overview Documents Search Analysis Commits Logs

Browse terms in field:

body

« First Term the Next

Hint:  
Edit the text field above and press Enter to seek to arbitrary terms.

Browse documents by term:

the

« First Doc 1 Next in 1 docs

| Position | Offsets | Payload |
|----------|---------|---------|
|          | 0       |         |
|          | 2       |         |
|          | 6       |         |
|          | 14      |         |

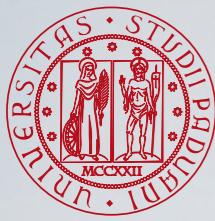
Browse documents by Doc # 0 in 3 docs

Copy values More like this Add document

(Select a row and double-click for more options.)

(To copy all or arbitrary field value(s), unselect all rows or select row(s), and click 'Copy values' button.)

| Field | Flags <small>Help</small> | Norm | Value   |
|-------|---------------------------|------|---|
| id    | Idfp-N-S-----             | 1    | d1  |
| body  | Idfp-N-S-----             | 20   | The quokka, the only member of the genus Setonix, is a small marsupial about the size of a domestic cat |



# Inverted Index: Options in Lucene

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | ENUM CONSTANTS | FIELD | METHOD DETAIL: ENUM CONSTANTS | FIELD | METHOD

org.apache.lucene.index

## Enum IndexOptions

java.lang.Object  
java.lang.Enum<IndexOptions>  
org.apache.lucene.index.IndexOptions

All Implemented Interfaces:

Serializable, Comparable<IndexOptions>

public enum IndexOptions  
extends Enum<IndexOptions>

Controls how much information is stored in the postings lists.

**WARNING:** This API is experimental and might change in incompatible ways in the next release.

### Enum Constant Summary

#### Enum Constants

##### Enum Constant and Description

###### DOCS

Only documents are indexed: term frequencies and positions are omitted.

###### DOCS\_AND\_FREQS

Only documents and term frequencies are indexed: positions are omitted.

###### DOCS\_AND\_FREQS\_AND\_POSITIONS

Indexes documents, frequencies and positions.

###### DOCS\_AND\_FREQS\_AND\_POSITIONS\_AND\_OFFSETS

Indexes documents, frequencies, positions and offsets.

###### NONE

Not indexed

### Method Summary

#### All Methods Static Methods Concrete Methods

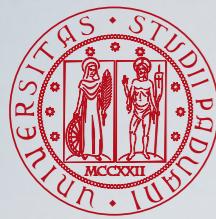
##### Modifier and Type

a che carattere  
⇒ in doc appare  
TOKEN  
↓  
es. usare per  
estrarre snippet

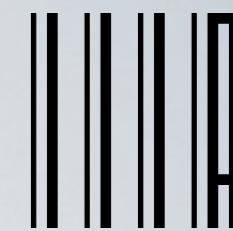
```

25
26  /**
27   * Represents a {@link Field} for containing the body of a document.
28   * <p>
29   * It is a tokenized field, not stored, keeping only document ids and term frequencies (see {@link
30   * IndexOptions#DOCS_AND_FREQS} in order to minimize the space occupation.
31   *
32   * @author Nicola Ferro (ferro@dei.unipd.it)
33   * @version 1.00
34   * @since 1.00
35   */
36  public class BodyField extends Field {
37
38      /**
39       * The type of the document body field
40       */
41      private static final FieldType BODY_TYPE = new FieldType();
42
43      static {
44          BODY_TYPE.setIndexOptions(IndexOptions.DOCS_AND_FREQS);
45          BODY_TYPE.setTokenized(true);
46          BODY_TYPE.setStored(false);
47      }
48
49
50      /**
51       * Create a new field for the body of a document.
52       *
53       * @param value the contents of the body of a document.
54       */
55      public BodyField(final Reader value) { super(ParsedDocument.FIELDS.BODY, value, BODY_TYPE); }
56
57
58      /**
59       * Create a new field for the body of a document.
60       *
61       * @param value the contents of the body of a document.
62       */
63      public BodyField(final String value) { super(ParsedDocument.FIELDS.BODY, value, BODY_TYPE); }
64
65
66
67
68
69

```

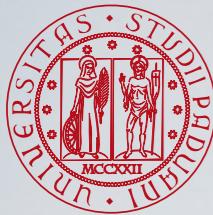


# Some Indexing Statistics

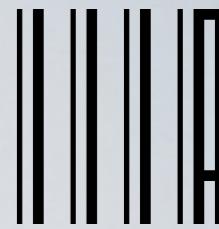


| No Stop<br>No Stem                        | Size      | Time        |
|---|-----------|-------------|
| Documents                                 | 151 MByte | 92 seconds  |
| Documents<br>and<br>Frequencies           | 209 MByte | 101 seconds |
| Documents<br>and Positions                | 555 Mbyte | 105 seconds |
| Documents<br>and Positions<br>and Offsets | 999 MByte | 136 seconds |

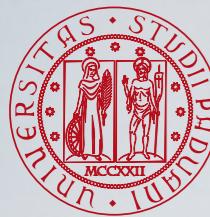
# Inside Lucene Indexes



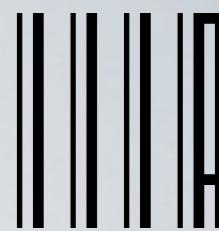
# Lucene Indexes



- Lucene indexes may be composed of multiple sub-indexes, or **segments**
  - Each segment is a fully independent index, which could be searched separately
- Indexes evolve by
  - Creating new segments for newly added documents
  - Merging existing segments
- Segments allow for
  - **incremental indexing**
  - **distributed indexing**
- Searches may involve multiple segments and/or multiple indexes, each index potentially composed of a set of segments
- Changes to the content of an index are made **visible** only after the writer **commits** by writing a new segments file (**segments\_N**)
  - This point in time, when the action of writing of a new segments file to the directory is completed, is an **index commit**



# What's in a Lucene Segment?



● **Segment info:** metadata about a segment, such as the number of documents, what files it uses, and information about how the segment is sorted

● **Field names:** the set of field names used in the index

● Inverted index

- **Term dictionary:** a dictionary containing all of the terms used in all of the indexed fields of all of the documents. The dictionary also contains the number of documents which contain the term, and pointers to the term's frequency and proximity data.

- **Term Frequency data:** for each term in the dictionary, the numbers of all the documents that contain that term, and the frequency of the term in that document

- **Term Proximity data:** for each term in the dictionary, the positions that the term occurs in each document

- **Stored Field values:** for each document, a list of attribute-value pairs, where the attributes are field names. These are used to store auxiliary information about the document, such as its title

- **Normalization factors:** for each field in each document, a value is stored that is multiplied into the score for hits on that field

- **Per-document values:** like stored values, these are also keyed by document number, but are generally intended to be loaded into main memory for fast access. Whereas stored values are generally intended for summary results from searches, per-document values are useful for things like scoring factors, faceted browsing, etc.

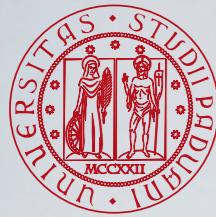
● Direct index

- **Term Vectors:** for each field in each document, the term vector (sometimes called document vector) may be stored. A term vector consists of term text and term frequency

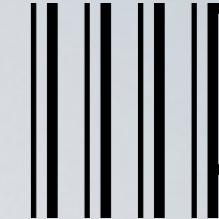
● Other

- **Point values:** Optional pair of files, recording dimensionally indexed fields, to enable fast numeric range filtering and large numeric values like BigInteger and BigDecimal (1D) and geographic shape intersection (2D, 3D)

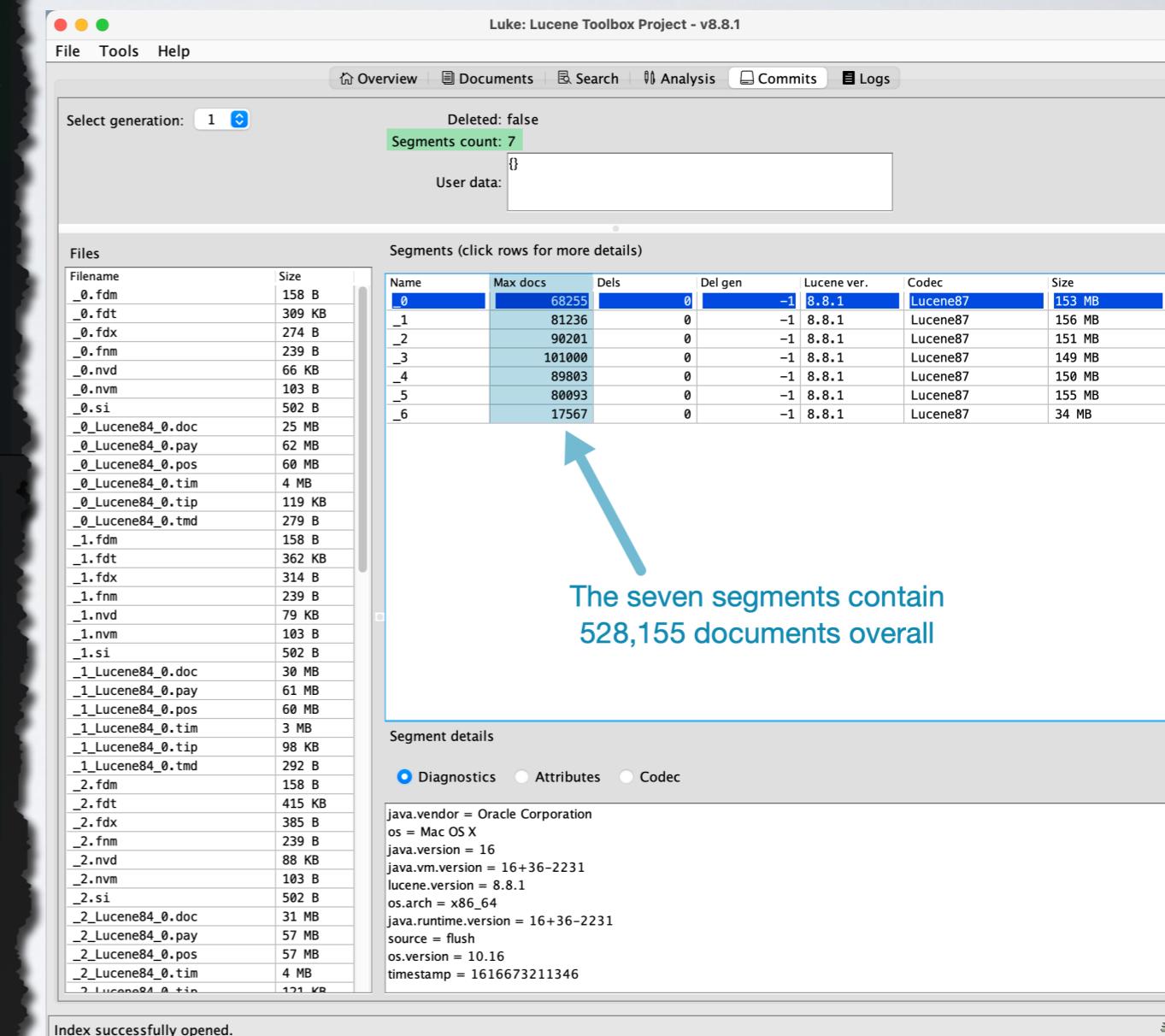
- **Live documents:** an optional file indicating which documents are live.



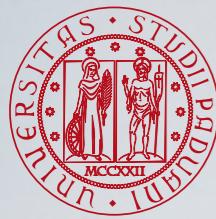
# Example of TIPSTER Index



```
[✓ index-nostop-nostem % ls
_0.fdm
_0.fdt
_0.fdx
_0.fnm
_0.nvd
_0.nvm
_0.si
_0_Lucene84_0.doc
_0_Lucene84_0.pay
_0_Lucene84_0.pos
_0_Lucene84_0.tim
_0_Lucene84_0.tip
_0_Lucene84_0.tmd
_1.fdm
_1.fdt
_1.fdx
_1.fnm
_1.nvd
_1.nvm
_1.si
_1_Lucene84_0.doc
_1_Lucene84_0.pay
_1_Lucene84_0.pos
_1_Lucene84_0.tim
_1_Lucene84_0.tip
_1_Lucene84_0.tmd
_2.fdm
_2.fdt
_2.fdx
_2.fnm
_2.nvd
_2.nvm
_2.si
_2_Lucene84_0.doc
_2_Lucene84_0.pay
_2_Lucene84_0.pos
_2_Lucene84_0.tim
_2_Lucene84_0.tip
_2_Lucene84_0.tmd
_3.fdm
_3.fdt
_3.fdx
_3.fnm
_3.nvd
_3.nvm
_3.si
_3_Lucene84_0.doc
_3_Lucene84_0.pay
_3_Lucene84_0.pos
_3_Lucene84_0.tim
_3_Lucene84_0.tip
_3_Lucene84_0.tmd
_4.fdm
_4.fdt
_4.fdx
_4.fnm
_4.nvd
_4.nvm
_4.si
_4_Lucene84_0.doc
_4_Lucene84_0.pay
_4_Lucene84_0.pos
_4_Lucene84_0.tim
_4_Lucene84_0.tip
_5.fdm
_5.fdt
_5.fdx
_5.fnm
_5.nvd
_5.nvm
_5.si
_5_Lucene84_0.doc
_5_Lucene84_0.pay
_5_Lucene84_0.pos
_5_Lucene84_0.tim
_5_Lucene84_0.tip
_6.fdm
_6.fdt
_6.fdx
_6.fnm
_6.nvd
_6.nvm
_6.si
_6_Lucene84_0.doc
_6_Lucene84_0.pay
_6_Lucene84_0.pos
_6_Lucene84_0.tim
_6_Lucene84_0.tip
_6_Lucene84_0.tmd
segments_1
write.lock
```



- No stop, no stemming
- Documents, frequencies, positions, and offsets



# Example of TIPSTER Index: Compound File Format

The image shows two screenshots. On the left is a file explorer window titled "index-nostop-nostem" showing a hierarchical list of index files. On the right is a screenshot of the "Luke: Lucene Toolbox Project - v8.8.1" interface showing index statistics.

**File Explorer Screenshot:**

- index-nostop-nostem
- index-nostop-stem
- index-stop-nostem
- index-stop-stem
- seupd2021-helloTipster-nostop-nostem.txt
- seupd2021-helloTipster-nostop-stem.txt
- seupd2021-helloTipster-stop-nostem.txt
- seupd2021-helloTipster-stop-stem.txt

**Luke Lucene Toolbox Project Screenshot:**

Select generation: 1 Deleted: false Segments count: 7

Segments (click rows for more details)

| Name | Max docs | Dels | Del gen | Lucene ver. | Codec    | Size   |
|------|----------|------|---------|-------------|----------|--------|
| _0   | 68255    | 0    | -1      | 8.8.1       | Lucene87 | 153 MB |
| _1   | 81236    | 0    | -1      | 8.8.1       | Lucene87 | 156 MB |
| _2   | 90201    | 0    | -1      | 8.8.1       | Lucene87 | 151 MB |
| _3   | 101000   | 0    | -1      | 8.8.1       | Lucene87 | 149 MB |
| _4   | 89803    | 0    | -1      | 8.8.1       | Lucene87 | 150 MB |
| _5   | 80093    | 0    | -1      | 8.8.1       | Lucene87 | 155 MB |
| _6   | 17567    | 0    | -1      | 8.8.1       | Lucene87 | 34 MB  |

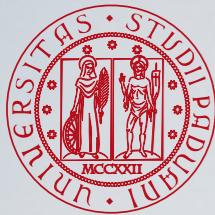
The seven segments contain 528,155 documents overall

Segment details

java.vendor = Oracle Corporation  
os = Mac OS X  
java.version = 16  
java.vm.version = 16+36-2231  
lucene.version = 8.8.1  
os.arch = x86\_64  
java.runtime.version = 16+36-2231  
source = flush  
os.version = 10.16  
timestamp = 1616672638872

Index successfully opened.

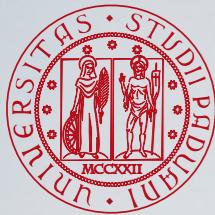
- For each segment
  - the **Compound File Format (.cfs)** merges all the index files into a single file
  - The **Compound Entry Table (.cfe)** holds pointers to all entries, i.e. the index files in the corresponding .cfs file
- The CFS is useful for systems which often run out of handles, due to the many files accessed during indexing
  - It is the default for small size segments
- The CFS slightly slows down index creation due to the usual creation of index files that are then packed together into the .cfs one



# Example of TIPSTER Index: Live Compound File Format

The screenshot shows a Java development environment with the following details:

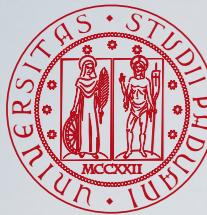
- File Explorer:** Shows a directory structure named "index-nostop-nostem" containing several temporary files (e.g., \_0\_Lucene8...oc\_ids\_0.tmp, \_0\_Lucene8...ointers\_1.tmp, \_0.fdm, \_0.fdt) and a "write.lock" file.
- Code Editor:** Displays the `DirectoryIndexer.java` file. The code configures an `IndexWriterConfig` object to use a RAM buffer size of 100 MB, open mode of `CREATE`, and commit on close. It also sets the use of compound files to `true`. A conditional block checks if an index path is provided.
- Output Window:** Shows the execution of the `DirectoryIndexer` class, displaying a log of indexing progress from 10,000 to 90,000 documents, each indexed in approximately 2 seconds.
- Bottom Status Bar:** Indicates "All files are up-to-date (moments ago)".



# Example of TIPSTER Index: Live Compound File Format

The screenshot shows a Java development environment with the following details:

- File Explorer:** Shows a directory structure named "index-nostop-nostem" containing several temporary files (e.g., \_0\_Lucene8...oc\_ids\_0.tmp, \_0\_Lucene8...ointers\_1.tmp, \_0.fdm, \_0.fdt) and a "write.lock" file.
- Code Editor:** Displays the `DirectoryIndexer.java` file. The code configures an `IndexWriterConfig` object to use a RAM buffer size of 100 MB, open mode of `CREATE`, and commit on close. It also sets the use of a compound file. A conditional block handles the `indexPath`.
- Output Window:** Shows the execution of the `DirectoryIndexer` class, displaying a log of indexing progress from 10000 to 90000 documents, each indexed in approximately 2 seconds.
- Bottom Status Bar:** Indicates "All files are up-to-date (moments ago)".



# Lucene Codec File Names



- All files belonging to a segment have the same name with varying extensions

- File names are never re-used
- The generation is a sequential long integer represented in alpha-numeric (base 36) form

## ● Overall

- **Segments File (segments\_N)**: information about a commit point
- **Lock File (write.lock)**: prevents multiple IndexWriters from writing to the same file

## ● Segment

- **Segment Info (.si)**: metadata about a segment

## ● Fields

- **Fields (.fnm)**: information about the fields
- **Field Data (.fdt)**: the stored fields (if any) for documents in compressed (LZ4) blocks of 16KB or more  
`FieldType.setStored(true)`
- **Field Index (.fdx)**: pointers to field data, i.e. first document ID of each block and its offset on disk
- **Fields Metadata (.fdm)**: metadata about the pointers

## ● Terms (the inverted index)

- **Term Dictionary (.tim)**: the term dictionary, contains the list of terms in each field along with per-term statistics (such as docfreq) and pointers to the frequencies, positions, payload and skip data in the .doc, .pos, and .pay files
- **Term Index (.tip)**: the index into the Term Dictionary, so that it can be accessed randomly

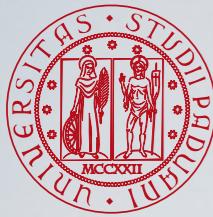
- **Term Metadata (.tdm)**: metadata about the pointers
- **Frequencies (.doc)**: the list of documents which contain each term along with frequency  
`FieldType.setIndexOptions(IndexOptions.DOCS_AND_FREQS)`
- **Positions (.pos)**: position information about where a term occurs in the index  
`FieldType.setIndexOptions(IndexOptions.DOCS_AND_FREQS_AND_POSITIONS)`
- **Payloads (.pay)**: additional per-position metadata information such as character offsets and user payloads  
`FieldType.setIndexOptions(IndexOptions.DOCS_AND_FREQS_AND_POSITIONS_AND_OFFSET)`
- **Norms (.nvd, .nvm)**: encodes length and boost factors for documents and fields  
`FieldType.setOmitNorms(false)`
- **Per-Document Values (.dvd, .dvm)**: encodes additional scoring factors or other per-document information  
`FieldType.setDocValuesType(...)`

## ● Direct Index

- **Term Vector Data (.tvd)**: term vector data  
`FieldType.setStoreTermVectors(true)`
- **Term Vector Index (.tvx)**: offset into the term vector data file

## ● Other

- **Point values (.dii, .dim)**: indexed points, if any  
`FieldType.setDimensions(...)`
- **Live Documents (.liv)**: information about what documents are live



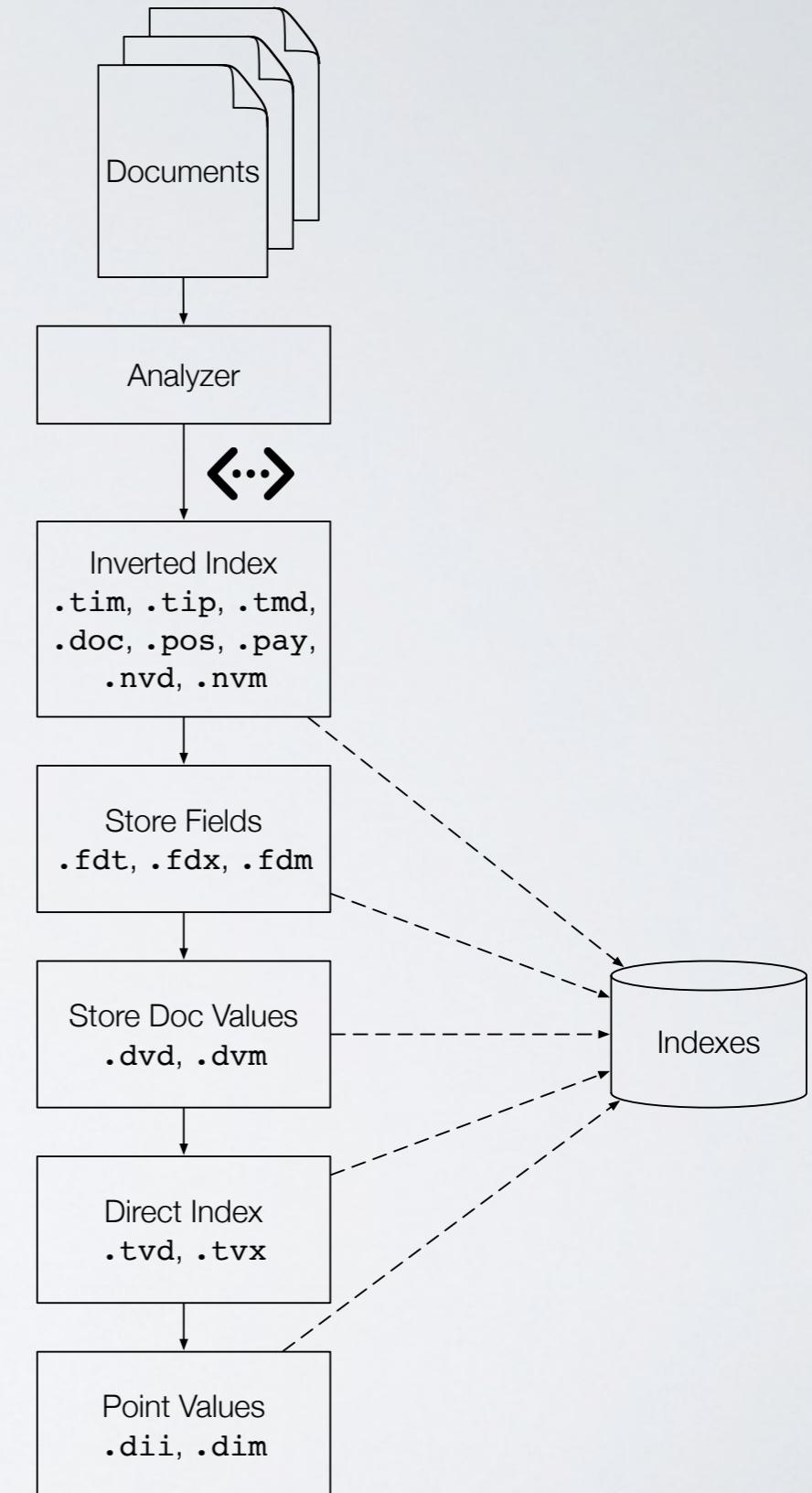
# Inside Lucene Indexing

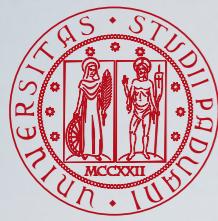
- Lucene provides different types of indexes

- inverted indexes
- store field
- doc values (not for textual fields)
- direct indexes

- Within **IndexWriter**, the order of indexing on the indexing chain is

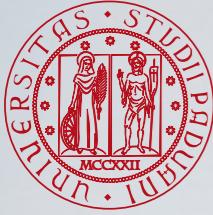
- inverted index
- store fields
- doc values
- direct index
- point values



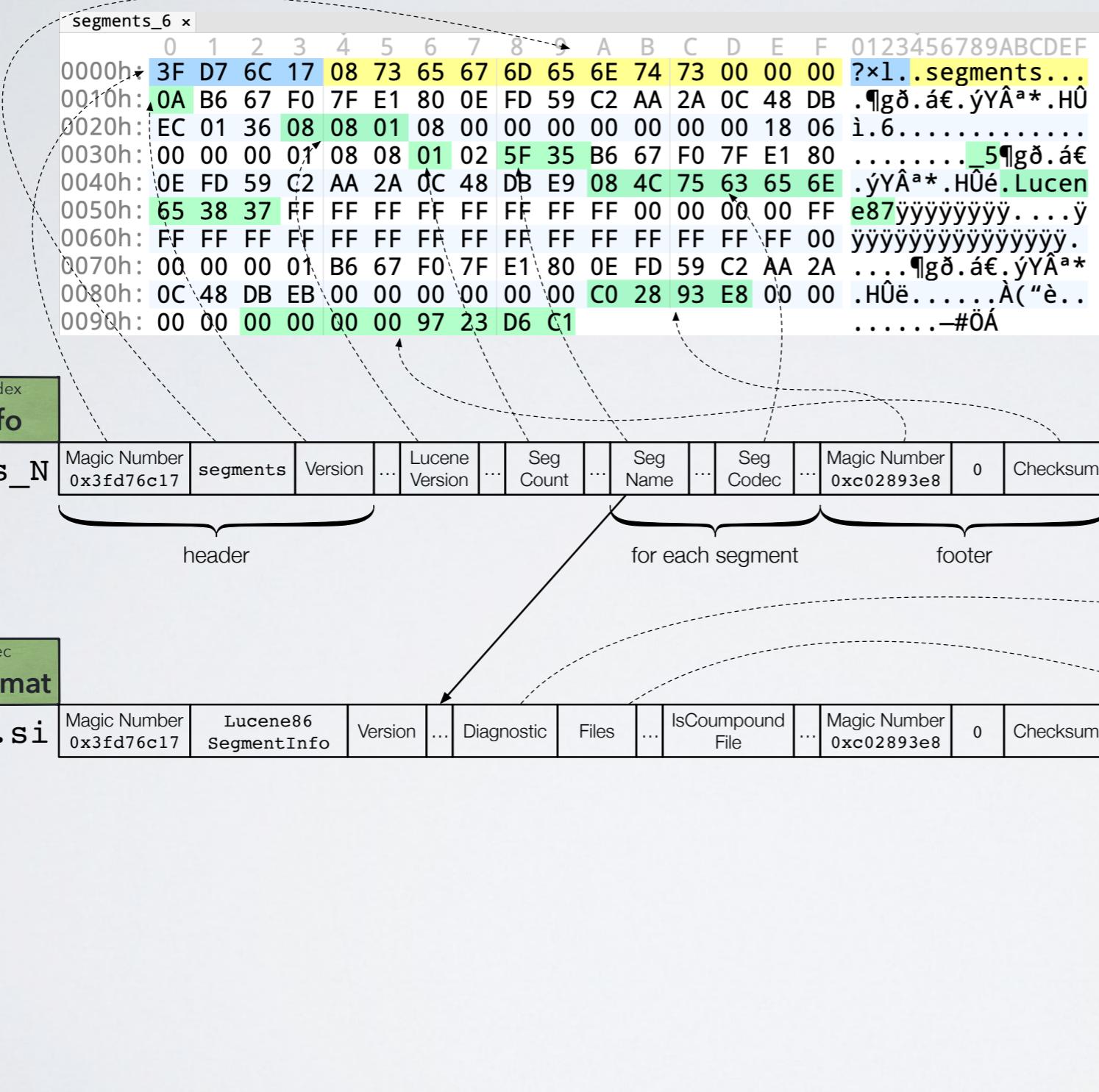


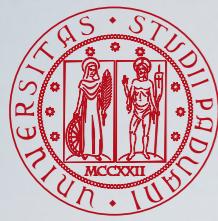
# The HelloIndex (toy) example

```
public class BodyField extends Field {  
  
    /**  
     * The type of the document body field  
     */  
    private static final FieldType BODY_TYPE = new FieldType();  
  
    static {  
        BODY_TYPE.setIndexOptions(IndexOptions.DOCS_AND_FREQS);  
        BODY_TYPE.setTokenized(true);  
        BODY_TYPE.setStored(true);  
        BODY_TYPE.setStoreTermVectors(true);  
    }  
  
    /**  
     * Create a new field for the body of a document.  
     *  
     * @param value the contents of the body of a document.  
     */  
    public BodyField(final Reader value) { super(HelloIndex.BODY,  
    }  
  
    /*  
     * Index the documents.  
     */  
    public void index(final List<Document> docs, final Analyzer analyzer) throws IOException {  
        System.out.printf("%n----- INDEXING DOCUMENTS -----%n");  
  
        // Open the directory in Lucene  
        final Directory directory = FSDirectory.open(indexPath);  
  
        // Utility class for holding all the required configuration for the indexer  
        final IndexWriterConfig config = new IndexWriterConfig(analyzer);  
  
        // force to re-create the index if it already exists  
        config.setOpenMode(IndexWriterConfig.OpenMode.CREATE);  
  
        // set the similarity. BM25 is already the default one  
        config.setSimilarity(new BM25Similarity());  
  
        // set the text-based codec  
        //config.setCodec(new Lucene87Codec(Lucene87Codec.Mode.BEST_SPEED));  
        config.setCodec(new SimpleTextCodec());  
  
        // prevent using the compound file format  
        config.setUseCompoundFile(false);  
  
        // The actual indexer  
        final IndexWriter writer = new IndexWriter(directory, config);  
    }  
}
```



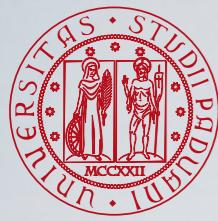
# Inside Lucene Segments





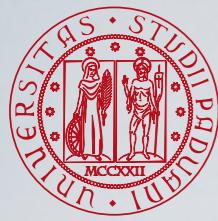
# Inside Lucene Fields



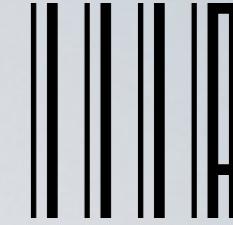


# Inside Lucene Inverted Index





# Lucene Index Summary



| Index Commit (segments_N) |           |     |
|---------------------------|-----------|-----|
| Seg Name                  | Seg Codec | ... |
| _5                        | Lucene87  | ... |
| Segment Info (_5.si)      |           |     |
| Files                     | ...       |     |
| _5.fnm                    | ...       |     |
| _5.fdt                    | ...       |     |
| _5.fdx                    | ...       |     |
| _5.fdm                    | ...       |     |
| _5_Lucene84_0.tim         | ...       |     |
| _5_Lucene84_0.doc         | ...       |     |
| _5_Lucene84_0.pos         | ...       |     |
| _5_Lucene84_0.tmd         | ...       |     |
| _5.tvd                    | ...       |     |
| _5.tvx                    | ...       |     |

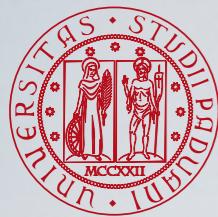
| Field Info (_5.fnm) |              |               |             |           |           |            |                |                        |             |     |
|---------------------|--------------|---------------|-------------|-----------|-----------|------------|----------------|------------------------|-------------|-----|
| Field Name          | Field Number | Term Vectors? | Omit Norms? | Payloads? | Docs Only | Docs Freqs | Docs Freqs Pos | Docs Freqs Pos Offsets | Docs Values | ... |
| id                  | 0            | false         | false       | false     | true      | false      | false          | false                  | NONE        | ... |
| body                | 1            | true          | false       | false     | false     | true       | false          | false                  | NONE        | ... |

| Postings (_5_Lucene84_0.tim) |            |         |               |     |            |
|------------------------------|------------|---------|---------------|-----|------------|
| Term                         | PosFPDelta | DocFreq | TotalTermFreq | ... | DocFPDelta |
| a                            |            | 2       | 3             | ... |            |
| about                        |            | 1       | 1             | ... |            |
| and                          |            | 1       | 1             | ... |            |
| approach                     |            | 1       | 1             | ... |            |
| are                          |            | 1       | 2             | ... |            |
| australia                    |            | 2       | 2             | ... |            |
| ...                          |            | ...     | ...           | ... |            |

Frequency (\_5\_Lucene84\_0.doc)

| DocDelta | Freq | ... |
|----------|------|-----|
| 0        | 2    | ... |
| 1        | 1    | ... |
| 0        | 1    | ... |
| 2        | 1    | ... |
| 2        | 1    | ... |
| 1        | 2    | ... |
| 1        | 1    | ... |
| 2        | 1    | ... |
| ...      | ...  | ... |

# Lucene Codecs



# What's in a Codec?

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

org.apache.lucene.codecs

**Class Codec**

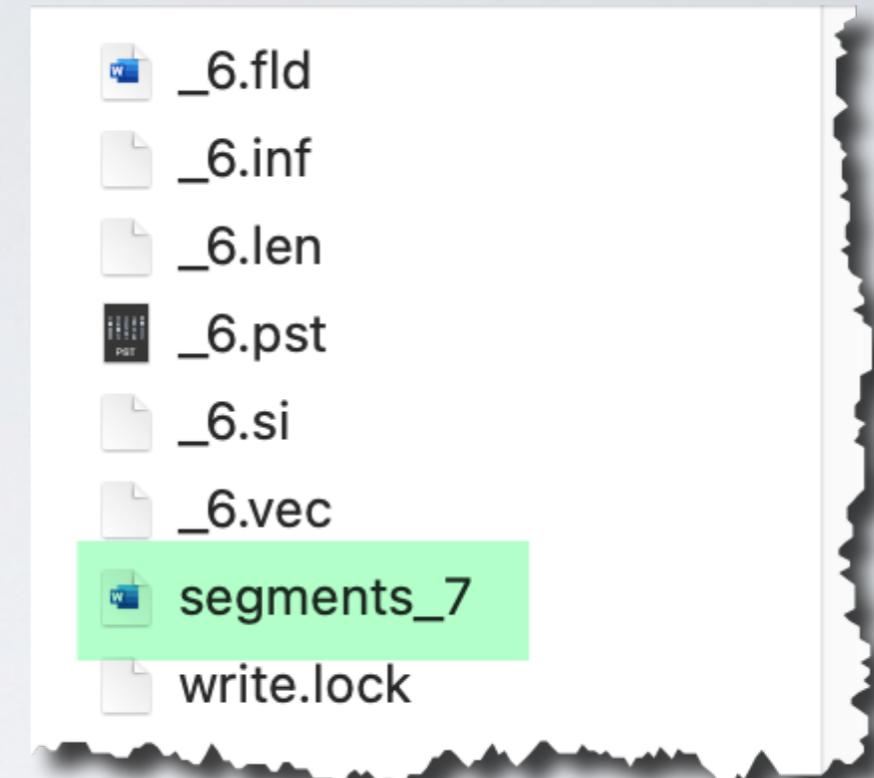
java.lang.Object  
org.apache.lucene.codecs.Codec

**Method Summary**

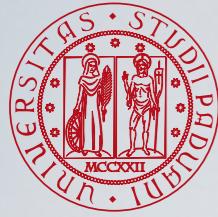
| All Methods                 | Static Methods  | Instance Methods | Abstract Methods | Concrete Methods |
|-----------------------------|---|------------------|------------------|------------------|
| Modifier and Type           | Method and Description  |                  |                  |                  |
| static Set<String>          | <b>availableCodecs()</b><br>returns a list of all available codec names   |                  |                  |                  |
| abstract CompoundFormat     | <b>compoundFormat()</b><br>Encodes/decodes compound files   |                  |                  |                  |
| abstract DocValuesFormat    | <b>docValuesFormat()</b><br>Encodes/decodes docvalues   |                  |                  |                  |
| abstract FieldInfosFormat   | <b>fieldInfosFormat()</b><br>Encodes/decodes field infos file   |                  |                  |                  |
| static Codec                | <b>forName(String name)</b><br>looks up a codec by name   |                  |                  |                  |
| static Codec                | <b>getDefault()</b><br>expert: returns the default codec used for newly created <code>IndexWriterConfigs</code> .         |                  |                  |                  |
| String                      | <b>getName()</b><br>Returns this codec's name   |                  |                  |                  |
| abstract LiveDocsFormat     | <b>liveDocsFormat()</b><br>Encodes/decodes live docs  |                  |                  |                  |
| abstract NormsFormat        | <b>normsFormat()</b><br>Encodes/decodes document normalization values   |                  |                  |                  |
| abstract PointsFormat       | <b>pointsFormat()</b><br>Encodes/decodes points index   |                  |                  |                  |
| abstract PostingsFormat     | <b>postingsFormat()</b><br>Encodes/decodes postings   |                  |                  |                  |
| static void                 | <b>reloadCodecs(ClassLoader classloader)</b><br>Reloads the codec list from the given <code>ClassLoader</code> .          |                  |                  |                  |
| abstract SegmentInfoFormat  | <b>segmentInfoFormat()</b><br>Encodes/decodes segment info file   |                  |                  |                  |
| static void                 | <b>setDefault(Codec codec)</b><br>expert: sets the default codec used for newly created <code>IndexWriterConfigs</code> . |                  |                  |                  |
| abstract StoredFieldsFormat | <b>storedFieldsFormat()</b><br>Encodes/decodes stored fields  |                  |                  |                  |
| abstract TermVectorsFormat  | <b>termVectorsFormat()</b><br>Encodes/decodes term vectors  |                  |                  |                  |

# The SimpleText Codec

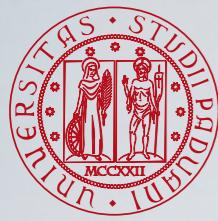
- It encodes an index in a textual format
  - Not to be used in production but useful for learning
- Note that file name and extensions are different from those of the default Lucene codec
  - The **segments\_N** file has always the same format because it is written by the `IndexWriter` and not by a `Codec`



| segments_7 | x           | 0           | 1           | 2           | 3        | 4        | 5        | 6        | 7        | 8        | 9          | A        | B                | C              | D                  | E        | F                  | 0123456789ABCDEF |
|------------|-------------|-------------|-------------|-------------|----------|----------|----------|----------|----------|----------|------------|----------|------------------|----------------|--------------------|----------|--------------------|------------------|
| 0000h:     | 3F D7 6C 17 | 08          | 73          | 65          | 67       | 6D       | 65       | 6E       | 74       | 73       | 00         | 00       | 00               | 00             | 00                 | 00       | ?×1..segments...   |                  |
| 0010h:     | 0A AA 71 85 | D6          | CC          | 5C          | 88       | 53       | E1       | D4       | C5       | CB       | F3         | F1       | F4               | .              | ªq...ÖÌ\ ^SáÔÅËóñô |          |                    |                  |
| 0020h:     | 18 01 37 08 | 08 01       | 08 00       | 00 00       | 00 00    | 00 00    | 00 00    | 00 00    | 00 00    | 00 00    | 1C 07      | .        | ..7.....         | .              | .                  | .        | .                  |                  |
| 0030h:     | 00 00 00 01 | 08 08       | 01 02       | 5F 36       | AA 71    | 85       | D6       | CC       | 5C       | .        | .          | .        | .                | .              | 6ªq...ÖÌ\          | .        | .                  |                  |
| 0040h:     | 88 53 E1 D4 | C5 CB       | F3 F1       | F4 15       | 0A 53    | 69       | 6D       | 70       | 6C       | .        | ^SáÔÅËóñô. | .        | Simpl            | .              | .                  | .        | .                  |                  |
| 0050h:     | 65 54 65 78 | 74 FF       | FF FF       | FF FF       | FF FF    | FF FF    | FF FF    | FF FF    | FF FF    | FF 00    | 00 00      | 00       | eTextÿÿÿÿÿÿÿÿÿÿ. | .              | .                  | .        | .                  |                  |
| 0060h:     | 00 FF FF FF | FF FF FF    | FF FF FF    | FF FF FF    | FF FF FF | FF FF FF | FF FF FF | FF FF FF | FF FF FF | FF FF FF | FF FF FF   | FF FF FF | .                | ÿÿÿÿÿÿÿÿÿÿÿÿÿÿ | .                  | .        | .                  |                  |
| 0070h:     | FF 00 00 00 | 00 01 AA    | 71 85       | D6 CC       | 5C 88    | 53 E1    | D4 28    | 93 E8    | 88 53    | 53 E1    | D4 28      | 93 E8    | 88 53            | 53 E1          | D4 28              | 93 E8    | ÿ....ªq...ÖÌ\ ^SáÔ |                  |
| 0080h:     | C5 CB F3 F1 | F4 17 00    | 00 00 00    | 00 00 00    | 00 00 00 | 00 00 00 | 00 00 00 | 00 00 00 | 00 00 00 | 00 00 00 | 00 00 00   | 00 00 00 | 00 00 00         | 00 00 00       | 00 00 00           | 00 00 00 | ÅËóñô....À("è      |                  |
| 0090h:     | 00 00 00 00 | 00 00 00 00 | 00 00 00 00 | 00 00 00 00 | BA AF    | F2 D9    | .        | .        | .        | .        | .          | .        | .                | .              | .                  | .        | ....°òÙ            |                  |



# SimpleText Codec: Segment Info

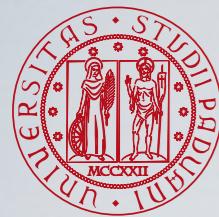


# SimpleText Codec: Fields

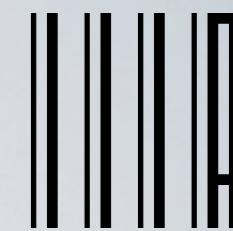
```
⚙ ~/Documents/progetti/software/search-engines/se-unipd/hello-index/experiment/index/_6.inf ◊
```

```
1 number of fields 2
2   name id
3   number 0
4   index options DOCS
5   term vectors false
6   payloads false
7   norms false
8   doc values NONE
9   doc values gen -1
10  attributes 0
11  data dimensional count 0
12  index dimensional count 0
13  dimensional num bytes 0
14  soft-deletes false
15  name body
16  number 1
17  index options DOCS_AND_FREQS
18  term vectors true
19  payloads false
20  norms true
21  doc values NONE
22  doc values gen -1
23  attributes 0
24  data dimensional count 0
25  index dimensional count 0
26  dimensional num bytes 0
27  soft-deletes false
28 checksum 0000000004031967692
29
```

```
⚙ ~/Documents/progetti/software/search-engines/se-unipd/hello-index/experiment/index/_6.fld ◊ master
1 |doc 0
2   field 0
3     name id
4     type string
5     value d1
6   field 1
7     name body
8     type string
9     value The quokka, the only member of the genus Setonix, is a small marsupial about the size of a domestic cat
10  doc 1
11   field 0
12     name id
13     type string
14     value d2
15   field 1
16     name body
17     type string
18     value wombats are small, short-legged, muscular quadrupedal marsupials that are native to Australia
19  doc 2
20   field 0
21     name id
22     type string
23     value d3
24   field 1
25     name body
26     type string
27     value Quokkas have little fear of humans and commonly approach people closely, particularly on Rottnest Island in Au
28 END
29 checksum 0000000000156156961
30
```



# SimpleText Codec: Inverted Index

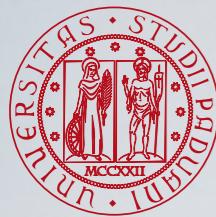


```
⚙ ~/Documents/progetti/software/search-engines/se-unipd/hello-index/experiment/index/_6.pst ✘ master
1 field body
2   term a
3     doc 0
4       freq 2
5     doc 2
6       freq 1
7   term about
8     doc 0
9       freq 1
10  term and
11    doc 2
12      freq 1
13  term approach
14    doc 2
15      freq 1
16  term are
17    doc 1
18      freq 2
19  term australia
20    doc 1
21      freq 1
22    doc 2
23      freq 1
24  term cat
25    doc 0
26      freq 1
27  term closely
28    doc 2
29      freq 1
30  term commonly
31    doc 2
```



# SimpleText Codec: Direct Index

```
1 doc 0
2 numfields 1
3 field 1
4   name body
5   positions false
6   offsets  false
7   payloads  false
8   numterms 15
9   term a
10    freq 2
11   term about
12    freq 1
13   term cat
14    freq 1
15   term domestic
16    freq 1
17   term genus
18    freq 1
19   term is
20    freq 1
21   term marsupial
22    freq 1
23   term member
24    freq 1
25   term of
26    freq 2
27   term only
28    freq 1
29   term quokka
30    freq 1
31   term setonix
32    freq 1
33   term size
34    freq 1
35   term small
36    freq 1
37   term the
38    freq 4
39 doc 1
40 numfields 1
41 field 1
42   name body
43   positions false
44   offsets  false
45   payloads  false
46   numterms 12
47   term are
48    freq 2
49   term australia
50    freq 1
```



# IndexWriterConfig: Full Indexing Configuration

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHOD    DETAIL: FIELD | CONSTR | METHOD

org.apache.lucene.index

**Class IndexWriterConfig**

java.lang.Object  
org.apache.lucene.index.LiveIndexWriterConfig  
org.apache.lucene.index.IndexWriterConfig

---

public final class **IndexWriterConfig**  
extends [LiveIndexWriterConfig](#)

Holds all the configuration that is used to create an [IndexWriter](#). Once [IndexWriter](#) has been created with this object, changes to this object will not affect the [IndexWriter](#) instance. For that, use [LiveIndexWriterConfig](#) that is returned from [IndexWriter.getConfig\(\)](#).

All setter methods return [IndexWriterConfig](#) to allow chaining settings conveniently, for example:

```
IndexWriterConfig conf = new IndexWriterConfig(analyzer);
conf.setter1().setter2();
```

Since:  
3.1

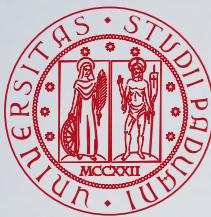
See Also:  
[IndexWriter.getConfig\(\)](#)

**Nested Class Summary**

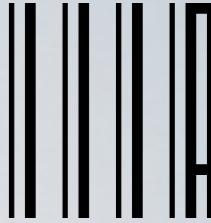
| Nested Classes | Modifier and Type | Class and Description   |
|----------------|-------------------|---|
|                | static class      | <a href="#">IndexWriterConfig.OpenMode</a><br>Specifies the open mode for <a href="#">IndexWriter</a> . |

**Field Summary**

| Fields | Modifier and Type | Field and Description   |
|--------|-------------------|---|
|        | static boolean    | <a href="#">DEFAULT_COMMIT_ON_CLOSE</a><br>Default value for whether calls to <a href="#">IndexWriter.close()</a> include a commit.   |
|        | static int        | <a href="#">DEFAULT_MAX_BUFFERED_DELETE_TERMS</a><br>Disabled by default (because IndexWriter flushes by RAM usage by default).   |
|        | static int        | <a href="#">DEFAULT_MAX_BUFFERED_DOCS</a><br>Disabled by default (because IndexWriter flushes by RAM usage by default).   |
|        | static long       | <a href="#">DEFAULT_MAX_FULL_FLUSH_MERGE_WAIT_MILLIS</a><br>Default value for time to wait for merges on commit or getReader (when using a <a href="#">MergePolicy</a> that implements <a href="#">MergePolicy.findFullFlushMerges</a> ( <a href="#">org.apache.lucene.index.MergeTrigger</a> , <a href="#">org.apache.lucene.index.SegmentInfos</a> , <a href="#">org.apache.lucene.index.MergePolicy.MergeContext</a> )). |
|        | static double     | <a href="#">DEFAULT_RAM_BUFFER_SIZE_MB</a><br>Default value is 16 MB (which means flush when buffered docs consume approximately 16 MB RAM).  |

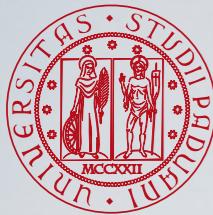


# IndexWriterConfig: Further Indexing Options

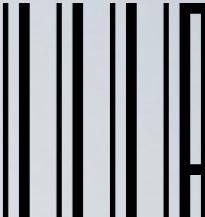


- **OpenMode:** defines how the index is opened (create, append, create or append)
- **Analyzer:** defines how the document pre-processing (tokenization, lexical analysis, ...) is performed
- **Similarity:** defines the adopted scoring algorithm. By default, Lucene uses the BM25 algorithm
- **Codec:** defines Lucene's internal encoders and decoders for all types of indexes
- **UseCompoundFile:** whether newly written segments should be packed in a compound file
- **IndexDeletionPolicy:** enables the management of commit points to implement such functionality as snapshots. By default DeletionPolicy only retains the last commit point
- **CommitOnClose:** determines whether a commit is performed before closing an index
- **MergePolicy:** determines how segments are merged.
- **MergeScheduler:** determines when segments are merged.
- **FlushPolicy:** determines when in-memory buffers are flushed. By default the timing will depend on the size of RAM and number of documents.
- **MaxBufferedDoc:** the minimal number of documents required before the buffered in-memory documents are flushed
- **RAMBufferSizeMB:** the amount of RAM that may be used for buffering added documents and deletions before they are flushed. Generally for faster indexing performance it's best to flush by RAM usage instead of document count and use as large a RAM buffer as you can

# Extracting Information from a Lucene Index



# Printing the Inverted Index



```
/*
public void printInvertedIndex() throws IOException {

    System.out.printf("%n----- PRINTING THE INVERTED INDEX -----%n");

    // Open the directory in Lucene
    final Directory dir = FSDirectory.open(indexPath);

    // Open the index
    final IndexReader index = DirectoryReader.open(dir);

    // The inverted index. Keys are terms; values are lists of (docID, term frequency) pairs
    final TreeMap<String, List<Map.Entry<String, Long>>> invertedIndex = new TreeMap<>();

    // Iterate over each document in the index
    for (int i = 0, docs = index.numDocs(); i < docs; i++) {

        // Read the document and get its identifier
        String docID = index.document(i).get(ID);

        // Get the vector of terms for that document
        Terms terms = index.getTermVector(i, BODY);

        // Get an iterator over the vector of terms
        TermsEnum termsEnum = terms.iterator();

        // Iterate until there are terms
        for (BytesRef term = termsEnum.next(); term != null; term = termsEnum.next()) {

            // Get the text string of the term
            String termstr = term.utf8ToString();

            // Get the total frequency of the term
            long freq = termsEnum.totalTermFreq();

            // Create a new (docID, term frequency) pair
            Map.Entry<String, Long> entry = new AbstractMap.SimpleEntry<>(docID, freq);
        }
    }
}
```

The inverted index is re-created in-memory as a map of terms -> postings list

Iterate over each document in the index and get the terms of vector for that document

Iterate over each term in the document and retrieve its frequency, to create an entry for the map



# Printing the Inverted Index

```
// update the inverted index with the new entry
invertedIndex.compute(termstr, (k, v) -> {

    // if the term is not already in the index, create a new list for its pairs
    if (v==null) {
        v = new ArrayList<>();
    }

    // add the pair to the list
    v.add(entry);

    // return the updated list
    return v;
});

// close the index and the directory
index.close();
dir.close();

// Print the inverted index to the console
invertedIndex.forEach( (k, v) -> {

    System.out.printf("+ %s ->", k);

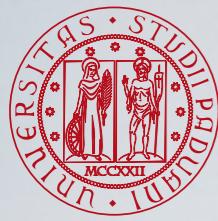
    v.forEach((e) -> {
        System.out.printf(" %s:%d", e.getKey(), e.getValue());
    });

    System.out.printf("%n");
});

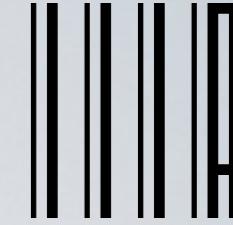
System.out.printf("-----%n");
}
```

Add the entry to the map, creating a new postings list if the term is not already present

Print each element of the map to the console



# Printing Vocabulary Statistics



```
public void printVocabularyStatistics() throws IOException {  
  
    System.out.printf("%n----- PRINTING VOCABULARY STATISTICS -----%n");  
  
    // Open the directory in Lucene  
    final Directory dir = FSDirectory.open(indexPath);  
  
    // Open the index - we need to get a LeafReader to be able to directly access terms  
    final LeafReader index = DirectoryReader.open(dir).leaves().get(0).reader();  
  
    // Total number of documents in the collection  
    System.out.printf("+ Total number of documents: %d%n", index.numDocs());  
  
    // Get the vocabulary of the index.  
    final Terms voc = index.terms(BODY);  
  
    // Total number of unique terms in the collection  
    System.out.printf("+ Total number of unique terms: %d%n", voc.size());  
  
    // Total number of terms in the collection  
    System.out.printf("+ Total number of terms: %d%n", voc.getSumTotalTermFreq());  
  
    // Get an iterator over the vector of terms in the vocabulary  
    final TermsEnum termsEnum = voc.iterator();  
  
    // Iterate until there are terms  
    System.out.printf("+ Vocabulary:%n");  
    System.out.printf(" - %-20s%-5s%-5s%n", "TERM", "DF", "FREQ");  
    for (BytesRef term = termsEnum.next(); term != null; term = termsEnum.next()) {  
  
        // Get the text string of the term  
        String termstr = term.utf8ToString();  
  
        // Get the document frequency (DF) of the term  
        int df = termsEnum.docFreq();  
  
        // Get the total frequency of the term  
        long freq = termsEnum.totalTermFreq();  
        System.out.printf(" - %-20s%-5d%-5d%n", termstr, df, freq);  
    }  
}
```

We need a **LeafReader** to be able to directly read a vector of all the terms from the index

Print overall statistics about the vocabulary

Iterate over terms in the index and retrieve their statistics to print them

# questions?

MY SPACE DEFENSE SHIELD DETECTS AN INCOMING THREAT. I AM LAUNCHING INTERCEPT ROCKETS.



scottadams@aol.com

www.dilbert.com

APPARENTLY THE BOYS AT GOOGLE USED A DEATH RAY TO BLAST THE INTERNATIONAL SPACE STATION OUT OF ORBIT AND TOWARD OUR HOUSE.



WHY DO YOU HAVE A SPACE DEFENSE SHIELD?



DOESN'T THAT SEEM LIKE A STUPID QUESTION NOW?



5-18-06 © 2006 Scott Adams, Inc./Dist. by UFS, Inc.