

# ES完全入门



## 28. ES文本分析



# 文本分析

- ◆ 插入 / 索引文档时，文本字段会经过一个分析(Analyze)过程
- ◆ 只适用于文本(text)字段
- ◆ 目标：构建和存储能够进行高效查询的数据结构(反向索引)
  - ◆ \_source存的是原始文档，并不用于查询
- ◆ 搜索 / 查询时经过类似的过程

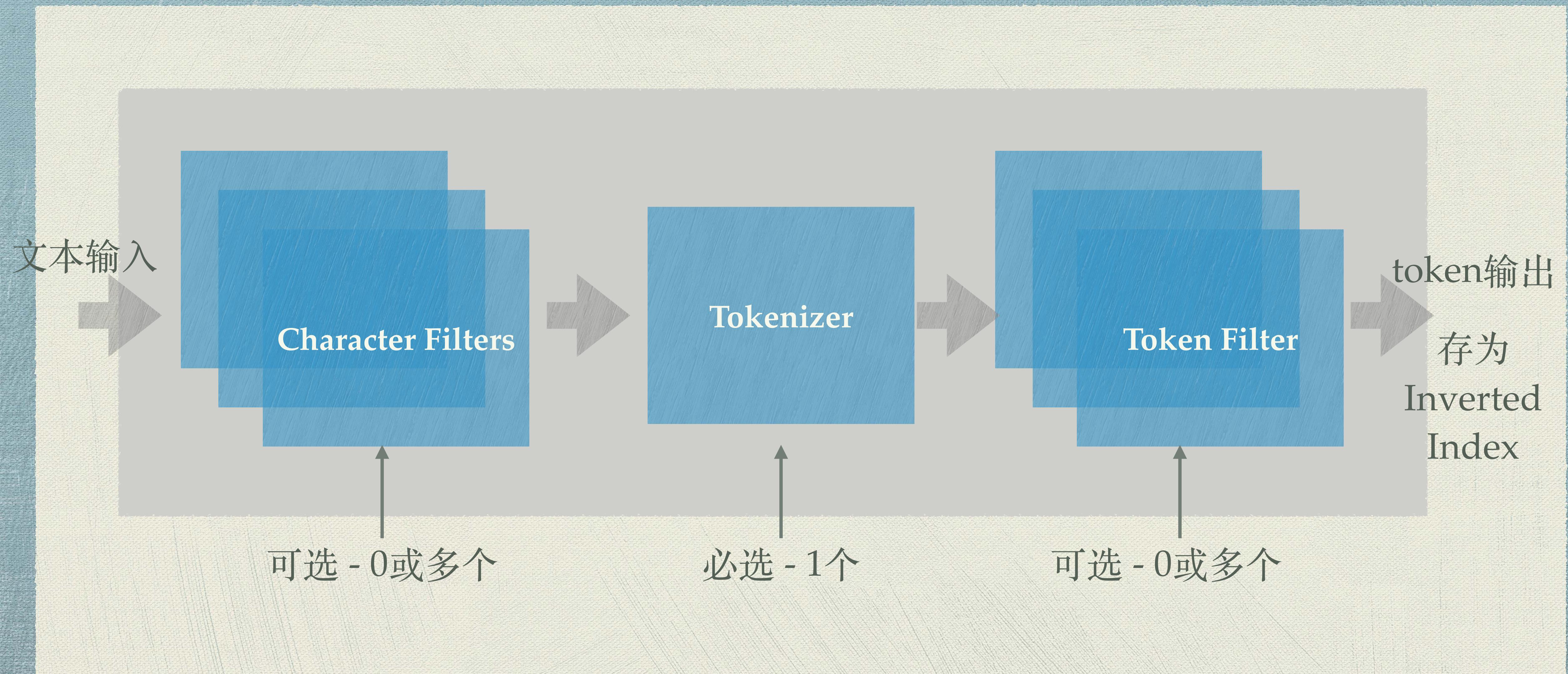
# 文本分析的两个主要阶段

- ◆ 分词(tokenization):
  - ◆ 将输入文本切分成单词或tokens
- ◆ 正规化(normalization)
  - ◆ 进一步完善/增强tokens(同义词/词根/移除)

Technique that computers use  
to extract worthwhile information  
to extract worthwhile  
information from the human  
language in a smart  
**text analysis** and efficient manner.



# 分析器(Analyzer)



# 字符过滤器(Character Filters)

- ◆ 添加, 移除或者改变字符
- ◆ 一个分析器可以包含0个或者多个字符过滤器
- ◆ 根据指定的顺序进行过滤
- ◆ 例子(html\_strip filer)
  - ◆ 输入: "<h2>Hello WORLD!</h2>"
  - ◆ 输出: "Hello WORLD!"
- ◆ 其它: mapping, pattern\_replace

# 分词器(Tokenizer)

- ◆ 一个分析器必定包含一个分词器
- ◆ 对文本进行分词(将句子拆分为单词(token)) , 比方说根据空格分词
- ◆ 分词时可能进一步移除字符(标点, 叹号)
- ◆ 例子
  - ◆ 输入: "Hello WORLD!"
  - ◆ 输出: ["Hello", "WORLD"]

# Token Filter

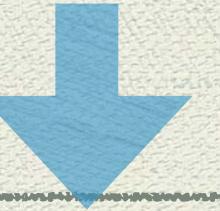
- ◆ 接收分词器的输出作为输入
- ◆ 添加 / 移除 / 修改其中的tokens
- ◆ 一个分析器可以包含0个或者多个token filters
- ◆ 根据指定的顺序进行过滤
- ◆ 例子(lowercase filter)
  - ◆ 输入: [Hello, WORLD]
  - ◆ 输出: ["hello", "world"]

# 标准分析器(Standard Analyzer)

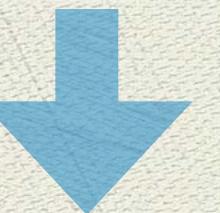


# 标准分析例子

“Hot cup of ☕ and a 🍿 is a Weird Combo :(!!”



“Hot”, “cup”, “of”, “☕”, “and”, “a”, “🍿”,  
“is”, “a”, “Weird”, “Combo”



“hot”, “cup”, “of”, “☕”, “and”, “a”, “🍿”,  
“is”, “a”, “weird”, “combo”

没有Character Filter,  
文本穿过无变化

标准分词器基于空格和标点  
进行分词

通过lowercase token filter  
转为小写

# 内置分析器

## Standard analyzer

缺省分析器。根据文法，标点和空格进行分词。输出的tokens转成小写

## Simple analyzer

根据非字母(空格，中划线，数字等)进行分词(Lowercase Tokenizer)

## Stop analyzer

Simple analyzer + 移除stop words(a, an, of, is etc)

## Whitespace analyzer

根据空格将文本切分成单词

## Keyword analyzer

不修改原文本，原样存储 (Noop Tokenizer)

## Language analyzer

处理不同国家的语言(英语/西班牙语/法语/俄语等等)

## Pattern analyzer

根据正则表达式分词(Pattern Tokenizer)

## Fingerprint analyzer

排序和移除重复token，拼接出单个token

# 小节1

- ◆ ES通过一个文本分析流程对文本字段进行分析。文本分析可使用内置或定制分析器。非文本字段并不经过分析。
- ◆ 文本分析主要经过两个阶段：
  - ◆ 分词(tokenization): 将输入文本拆分成单个的单词或tokens
  - ◆ 正规化(normalization): 对单词进行完善/增强(同义词/词根/移除)
- ◆ 文本分析由分析器(Analyzer)完成。分析器由character filters + tokenizer + token filter所组成
- ◆ ES缺省使用标准分析器(standard analyzer)，它不包含character filter，包含一个standard tokenizer和两个token filters(lowercase和stop， stop缺省禁用)

# 小节2

- ◆ 每个分析器必须有且仅有一个tokenizer，但是它可以有0个或者多个character和token filters
  - ◆ Character Filter用于去掉不需要的字符
  - ◆ Tokenizer对文本进行分词
  - ◆ Token Filters对Token再进行enhance/enrich/删除
- ◆ ES缺省内置了一些分析器，可以混用tokenizer + character / token filers以满足特定需求