Using Capture Groups

Victor Grazi
ORACLE JAVA CHAMPION, SPEAKER AND GEEK

@vgrazi

Introduction

Using Capture groups

Parenthesized patterns that match a character sequence

Easily extracted from your text

Can be followed by quantifiers

LDAP Labels for Company XYZ
Format: BusinessUnit-Team-*Subteam-*Project-Region

* Optional

Examples

- securities-development-equities-valuation-asia
- fixed_income-development-equities-emea
- fx-development-america

Goal

Extract business unit and region from each label

Capture Group

(pattern)quantifier

e.g.

(\d+\w+){2/4}

or

(\w)+

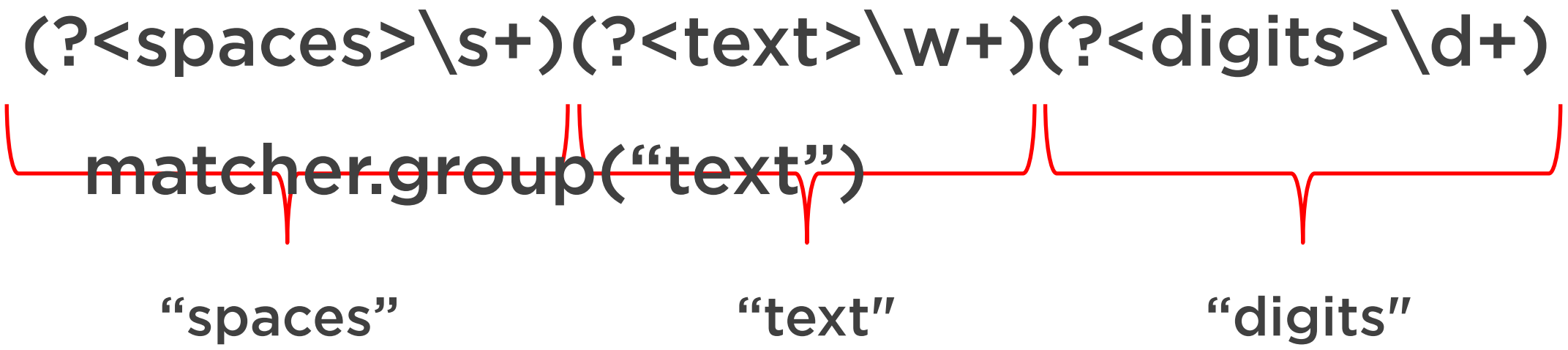
Capture Groups are Brittle

(\s+)(\w+)(\d+)

matcher.group(2)

?

Named Capture Groups



Case Insensitive Flag

```
(?i)
((?i)my-pattern)
((?i)my-(-?i)pattern)
((?id)my-pattern)  groupCount=1
(?i:my-pattern)    groupCount=0
```

Back references

To Be Or Not To Be, That Is The Question

(To Be).*\1

Replacing Text with Capture Groups

```
"Equities-Development-Asia".replaceAll(
  "(?<business>\\w+)(-(\\w+))+(?<region>\\w+)"
  "Region:${region} Business:${business}"));
```

Region:Asia Business:Equities

Duplicate Names are Ignored

(?<some-name>\w+) (?<some-name>\w+)

${some-name}

Summary

- Using Capture groups
- Parenthesized patterns that match a character sequence
- Easily extracted from your text
- Back references: \1 or \k<name>
- Replacement: $1 or ${name}