

Yaml配置文件集成

联系QQ: [2816010068](#), 加入会员群

目录

- Yaml配置介绍
- Yaml配置文件解析
- Koala配置项梳理
- Koala配置文件生成器开发

Yaml配置文件介绍

- 基本语法
 - 大小写敏感
 - 使用缩进表示层级关系
 - 缩进时不允许使用Tab键，只允许使用空格。
 - 缩进的空格数目不重要，只要相同层级的元素左侧对齐即可
- YAML 支持的数据结构有三种。
 - 对象：键值对的集合，又称为映射（mapping） / 哈希（hashes） / 字典（dictionary）
 - 数组：一组按次序排列的值，又称为序列（sequence） / 列表（list）
 - 纯量（scalars）：单个的、不可再分的值

Yaml配置文件

- 对象

- animal: pets
- 对应的json格式:

```
{ animal: 'pets' }
```

- 数组

- Cat
- Dog
- Goldfish

对应的json格式:

```
[ 'Cat', 'Dog', 'Goldfish' ]
```

Yaml配置文件

- 对象和数组可以结合使用，形成复合结构。

```
languages:  
- Ruby  
- Perl  
- Python  
websites:  
  YAML: yaml.org  
  Ruby: ruby-lang.org  
  Python: python.org  
  Perl: use.perl.org
```

Yaml配置文件

- 常量

- 字符串
- 布尔值
- 整数
- 浮点数
- Null
- 时间
- 日期

number: 12.30

isSet: true

Yaml配置文件解析

- 安装第三方库: `go get gopkg.in/yaml.v2`
- 新建一个配置文件, 比如:

```
1  site:
2    port: 8080
3    https_on: false
4    domain: "pythonsite.com"
5    https_port: 443
6  nginx:
7    port: 80
8    log_path: "/opt/log/nginx.log"
9    path: "/opt/nginx/"
10   site_name: "this is my web site"
11   site_addr: "BeiJing"
12   upstream:
13     - 127.0.0.1:8080
14     - 127.0.0.2:8090
15     - 127.0.0.3:9090
```

Y a m l 配置文件解析

- 构造对应的Golang结构体

```
type Config struct {  
    Site SiteConfig `yaml:"site"`  
    Nginx NginxConfig `yaml:"nginx"`  
}  
  
type SiteConfig struct {  
    Port      int    `yaml:"port"`  
    HttpsOn   bool   `yaml:"https_on"`  
    Domain    string `yaml:"domain"`  
    HttpsPort int    `yaml:"https_port"`  
}  
  
type NginxConfig struct {  
    Port      int    `yaml:"port"`  
    LogPath   string `yaml:"log_path"`  
    SiteName  string `yaml:"site_name"`  
    SiteAddr  string `yaml:"site_addr"`  
    Upstream  []string `yaml:"upstream"`  
}
```


Yaml配置文件

- 进行解析

```
data, err := ioutil.ReadFile("./test.yaml")
if err != nil {
    fmt.Printf("read file failed, err:%v\n", err)
    return
}

var conf Config
err = yaml.Unmarshal(data, &conf)
if err != nil {
    fmt.Printf("unmarshal failed err:%v\n", err)
    return
}

fmt.Printf("conf is %#v\n", conf)
```

Koala rpc框架配置项梳理

- 服务监听端口
- Prometheus监听端口
- 服务的名字
- 注册中心配置
- 日志级别配置

Koala 环境划分

- 环境划分
 - product：生成环境
 - test：测试环境
- 环境区分
 - 通过环境变量进行区分
 - KOALA_ENV=product
 - KOALA_ENV=test
- env包
 - 获取当前的环境

koala配置文件目录划分

- conf
 - product
 - test

Koala env包开发

- 通过os包获取
 - `os.Getenv(“KOALA_ENV”)`

Koala服务部署规范

- xxx_service
 - bin
 - conf
 - product
 - test
 - script
 - Logs
- 名词解释
 - RootDir: /xxx/xxx/xxx/xxx_service
 - ConfigDir: /xxx/xxx/xxx/xxx_service/conf/product/或者
/xxx/xxx/xxx/xxx_service/conf/test/

配置文件生成器开发

- config_generator
 - 主要功能
 - 生成当前框架的配置文件
- Koala框架支持配置文件解析
 - 新增koala/server包
 - 通过Init函数, 完成配置文件的解析
- main_generator改造
 - 通过配置获取监听的端口
 - 通过配置获取普罗米修斯的端口

参考资料

- Yaml文件教程: <http://www.ruanyifeng.com/blog/2016/07/yaml.html>