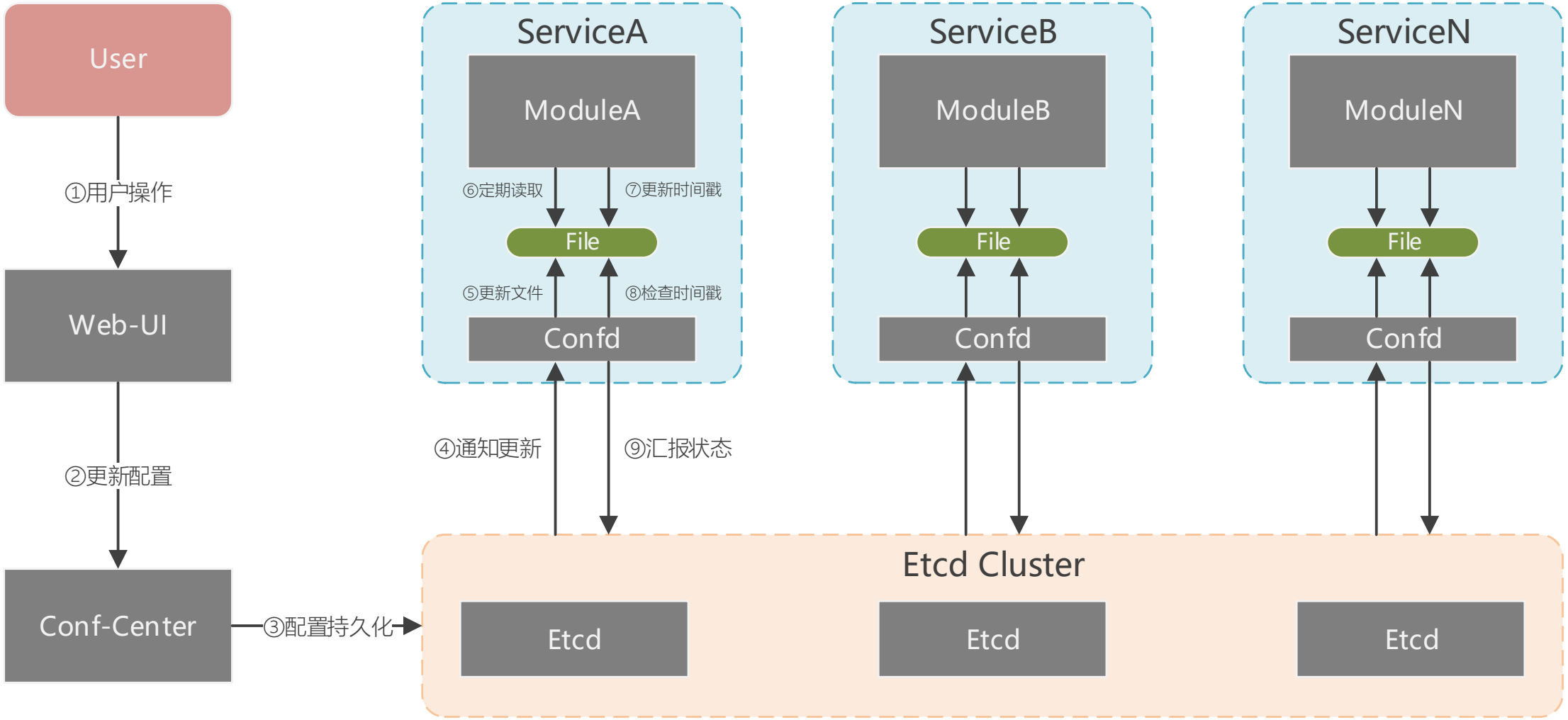
配置中心设计方案

整体架构


User
①用户操作
Web-UI
②更新配置
Conf-Center
③配置持久化
ServiceA
ModuleA
⑥定期读取
⑦更新时间戳
File
⑤更新文件
⑧检查时间戳
Confd
ServiceB
ModuleB
File
Confd
ServiceN
ModuleN
File
Confd
④通知更新
⑨汇报状态
Etcd Cluster
Etcd
Etcd
Etcd

配置文件同步和反馈

更新

1. Confd服务进程在收到Etcd的更新通知。
2. 更新本地配置文件的内容。
3. 文件内容时间戳（mtime）、状态时间戳（ctime）均设置为更新时间。

加载

1. 模块定时检查内容时间戳（ mtime ），是否相对于前次加载时有变化。
2. 当内容时间戳（ mtime ）变化时，确认内容有变化后重新加载。
3. 加载完成后，修改状态时间戳（ ctime ）为加载完成时的时间。（一定比mtime新）。

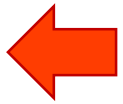
反馈

1. Confd定时检查配置文件状态时间戳（ctime），是否比内容时间戳新（mtime）。
2. 当状态时间戳（ctime）比内容时间戳（mtime）新时，说明模块已经完成了加载。
3. 上报加载状态。

Confd和etcd之间的接口、协议

配置文件规约 (schema)

```
{
    "cache_flag": true,
    "kvs_addr": "11.11.230.164:54300",
    "l5ranker_config": {
        "top3_stgy_threshold": 0
    },
    "model_conf": [
        {
            "feed_keys": [1],
            "fetch_keys": [3, 4, 7],
            "mid": 11
        }
    ],
    "redis_addr": [
        {
            "host": "11.11.230.165",
            "port": 26379
        },
        {
            "host": "11.11.230.164",
            "port": 26380
        }
    ]
}
```



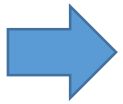
```
[
    {
        "name": "cache_flag",
        "type": "boolean",
        "description": "是否打开缓存",
        "defaultValue": false
    },
    {
        "name": "kvs_addr",
        "type": "string",
        "description": "KV存储服务器地址",
        "defaultValue": "localhost:54300"
    },
    {
        "name": "l5ranker_config.top3_stgy_threshold",
        "type": "number",
        "description": "L5直接回答比例阈值",
        "defaultValue": 0.2
    },
    {
        "name": "model_conf[].feed_keys[]",
        "type": "number",
        "description": "模型输入KEY",
        "defaultValue": null
    },
    {
        "name": "model_conf[].fetch_keys[]",
        "type": "number",
        "description": "模型获取KEY",
        "defaultValue": null
    },
    {
        "name": "model_conf[].mid",
        "type": "number",
        "description": "模型ID",
        "defaultValue": null
    },
    {
        "name": "redis_addr[].host",
        "type": "string",
        "description": "Redis服务地址",
        "defaultValue": null
    },
    {
        "name": "redis_addr[].port",
        "type": "number",
        "description": "Redis服务端口",
        "defaultValue": 23679
    }
]
```

规约项：

- **name:** 名称。
- **type:** 类型，(string、number、boolean)。
- **description:** 描述信息。
- **defaultValue:** 默认值。

配置项定位规则 (location)

```
{
    "cache_flag": true,
    "kvs_addr": "11.11.230.164:54300",
    "l5_model_version": 5,
    "l5ranker_config": {
        "top3_stgy_threshold": 0
    },
    "l5ranker_path": "../l5ranker/default",
    "model_conf": [
        {
            "feed_keys": [1],
            "fetch_keys": [3, 4, 7],
            "mid": 11
        }
    ],
    "redis_addr": [
        {
            "host": "11.11.230.165",
            "port": 26379
        },
        {
            "host": "11.11.230.164",
            "port": 26380
        },
        {
            "host": "11.11.230.166",
            "port": 26381
        }
    ]
}
```



```
cache_flag                              =  true
kvs_addr                                =  "11.11.230.164:54300"
l5_model_version                        =  5
l5ranker_config.top3_stgy_threshold     =  0
l5ranker_path                           =  "../l5ranker/default"
model_conf[0].feed_keys[0]              =  1
model_conf[0].fetch_keys[0]             =  3
model_conf[0].fetch_keys[1]             =  4
model_conf[0].fetch_keys[2]             =  7
model_conf[0].mid                       =  11
redis_addr[0].host                      =  "11.11.230.165"
redis_addr[0].port                      =  26379
redis_addr[1].host                      =  "11.11.230.164"
redis_addr[1].port                      =  26380
redis_addr[2].host                      =  "11.11.230.166"
redis_addr[2].port                      =  26381
```

值类型约定：

- **string:** 字符串型，使用引号表示
- **number:** 数字型，整型和浮点型。
- **boolean:** 布尔型，true或false。

配置文件格式规范

可以预期的名称

在JSON中配置项应该总是在Value部分，让Key可以预期。这样Schema容易定义，在静态语言中比较好处理。

好的列子：

```
[
    {
        "bid": "1001",
        "top3_stgy_threshold": 0.6
    },
    {
        "bid": "1002",
        "top3_stgy_threshold": 0.5
    }
]
```

坏的例子：

```
{
    "1001": {   //Key 是配置数据的一部分
        "top3_stgy_threshold": 0.6
    },
    "2001": {
        "top3_stgy_threshold": 0.5
    }
}
```

配置项目的命名规则

当前系统配置由于历史原因，命名方式混乱。为了方便使用，后续建议都使用小驼峰法命名。

例如：

```
{
    "cacheFlag": true,
    "kvsAddr": "11.11.230.164:54300",
    "l5rankerConfig": {
        "top3StgyThreshold": 0
    },
    "modelConf": [
        {
            "feedKeys": [1],
            "fetchKeys": [3, 4, 7],
            "mid": 11
        }
    ]
}
```

配置中心接口（配置部分）

配置文件接口

```
// ------------------------------------------------
// 获取服务列表
GET /api/v1/services/
[
    "serviceA", "serviceB", "serviceC"
]

// ------------------------------------------------
// 获取服务配置文件列表
GET /api/v1/services/{serviceName}/
[
    "fileNameA", "fileNameB"
]

// ------------------------------------------------
// 获取配置文件内容
GET /api/v1/services/{serviceName}/{fileName}

// ------------------------------------------------
// 更新配置文件内容（幂等覆盖）
PUT /api/v1/services/{serviceName}/{fileName}

// ------------------------------------------------
// 删除置文件
DELETE /api/v1/services/{serviceName}/{fileName}
```

配置项接口

```
// ------------------------------------------------
// 获取配置文中的部分配置项
GET /api/v1/schemas/services/{serviceName}/{fileName}/{name}
```

例如：

```
GET /api/v1/schemas/services/smu/application/redis_addr[0]
{
    "host": "11.11.230.164",
    "port": 26380
}

// ------------------------------------------------
// 更新配置文件中部分配置项
POST /api/v1/schemas/services/{serviceName}/{fileName}/{name}
```

例如：

```
POST /api/v1/schemas/services/smu/application/redis_addr[0].host

"11.11.230.164"

// ------------------------------------------------
// 删除配置文件中部分配置项
DELETE /api/v1/schemas/services/{serviceName}/{fileName}/{name}
```

配置中心接口（规约部分）

规约信息接口

```
// ------------------------------------------------
// 获取配置文件规约
GET /api/v1/schemas/services/{serviceName}/{fileName}
[
    {
        "name": "name",
        "type": "boolean",
        "description": "配置项说明",
        "defaultValue": null
    }
]

// ------------------------------------------------
// 更新配置文件规约（幂等覆盖）
PUT /api/v1/schemas/services/{serviceName}/{fileName}
[
    {
        "name": "name",
        "type": "boolean",
        "description": "配置项说明",
        "defaultValue": null
    }
]
```

```
// ------------------------------------------------
// 删除配置文件规约
DELETE /api/v1/schemas/services/{serviceName}/{fileName}

// ------------------------------------------------
// 获取配置文件规约列表
GET /api/v1/schemas/services/{serviceName}/
[
    "fileNameA", "fileNameB", "fileNameC"
]
```