1.loader 的配置

1.1 webpack config

```
module: {
    rules: [
        {
            test: /\.css/,
            use: [
                { loader: 'style-loader' },
                { loader: 'css-loader' }
            ]
        }
    ]
}
```

1.2 inline内联

```
import 'style-loader!css-loader!style.css';
```

2.RuleSet

- RuleSet 类主要作用于过滤加载 module 时符合匹配条件规则的 loader
- RuleSet 在内部会有一个默认的 module.defaultRules 配置
- 在真正加载 module 之前会和你在 webpack config 配置文件当中的自定义 module.rules 进行合并，然后转化成对应的匹配过滤器
- WebpackOptionsDefaulter (https://github.com/webpack/webpack/blob/v4.43.0/lib/WebpackOptionsDefaulter.js#L60)
- RuleSet (https://github.com/webpack/webpack/blob/v4.43.0/lib/RuleSet.js#L104)

```
[
    { type: 'javascript/auto', resolve: {} },
    { test: /\.mjs$/i,type: 'javascript/esm',resolve: { mainFields: [Array] } },
    { test: /\.json$/i, type: 'json' },
    { test: /\.wasm$/i, type: 'webassembly/experimental' }
]
```

```
class NormalModuleFactory {
    this.ruleSet = new RuleSet(options.defaultRules.concat(options.rules));
}
```

2.1 rule配置

- test 用以匹配满足条件的 loader
- include 用以匹配满足条件的 loader
- exclude 排除满足条件 loader
- resource condition (https://github.com/webpack/webpack/blob/v4.43.0/lib/RuleSet.js#L183)
- resourceQuery 在路径中带 query 参数的匹配规则
- normalizeRule (https://github.com/webpack/webpack/blob/v4.43.0/lib/RuleSet.js#L183)作用实际就是对传入的 rules 配置进行序列化(格式化)的处理为统一的格式
- normalizeCondition (https://github.com/webpack/webpack/blob/c9d4ff7b054fc581c96ce0e53432d44f9dd8ca72/lib/RuleSet.js#L413) 函数执行后始终返回的是一个函数，这个函数的用途就是接受模块的路径，然后使用你所定义的匹配使用去看是否满足对应的要求，如果满足那么会使用这个 loader，如果不满足那么便会过滤掉
- 在 RuleSet 构造函数内部使用静态方法normalizeUse (https://github.com/webpack/webpack/blob/c9d4ff7b054fc581c96ce0e53432d44f9dd8ca72/lib/RuleSet.js#L351) 方法来输出最终和 condition 对应的 rule 结果
- 经过 normalizeUse 函数的格式化处理，最终的 rule 结果为一个数组，内部的 object 元素都包含 loader/options 等字段

```
static normalizeRule(rule, refs, ident) {
    let condition = {
        test: rule.test,
        include: rule.include,
        exclude: rule.exclude
    };
    newRule.resource = RuleSet.normalizeCondition(condition);
}
```

```
static normalizeCondition(condition) {
    if (typeof condition === "string") {
        return str => str.indexOf(condition) === 0;
    }
    if (typeof condition === "function") {
        return condition;
    }
    if (condition instanceof RegExp) {
        return condition.test.bind(condition);
    }
}
```

```
static normalizeUse(use, ident) {
    if (typeof use === "function") {
        return data => RuleSet.normalizeUse(use(data), ident);
    }
}
```

normalizeUse结果

```
[
    {loader:'style-loader'},
    {loader:'css-loader'}
]
```

rules:

```
[
  {
    resource: [Function],
    resourceQuery: [Function],
    use: [
      {loader:'style-loader'},
      {loader:'css-loader'}
    ]
  }
]
```
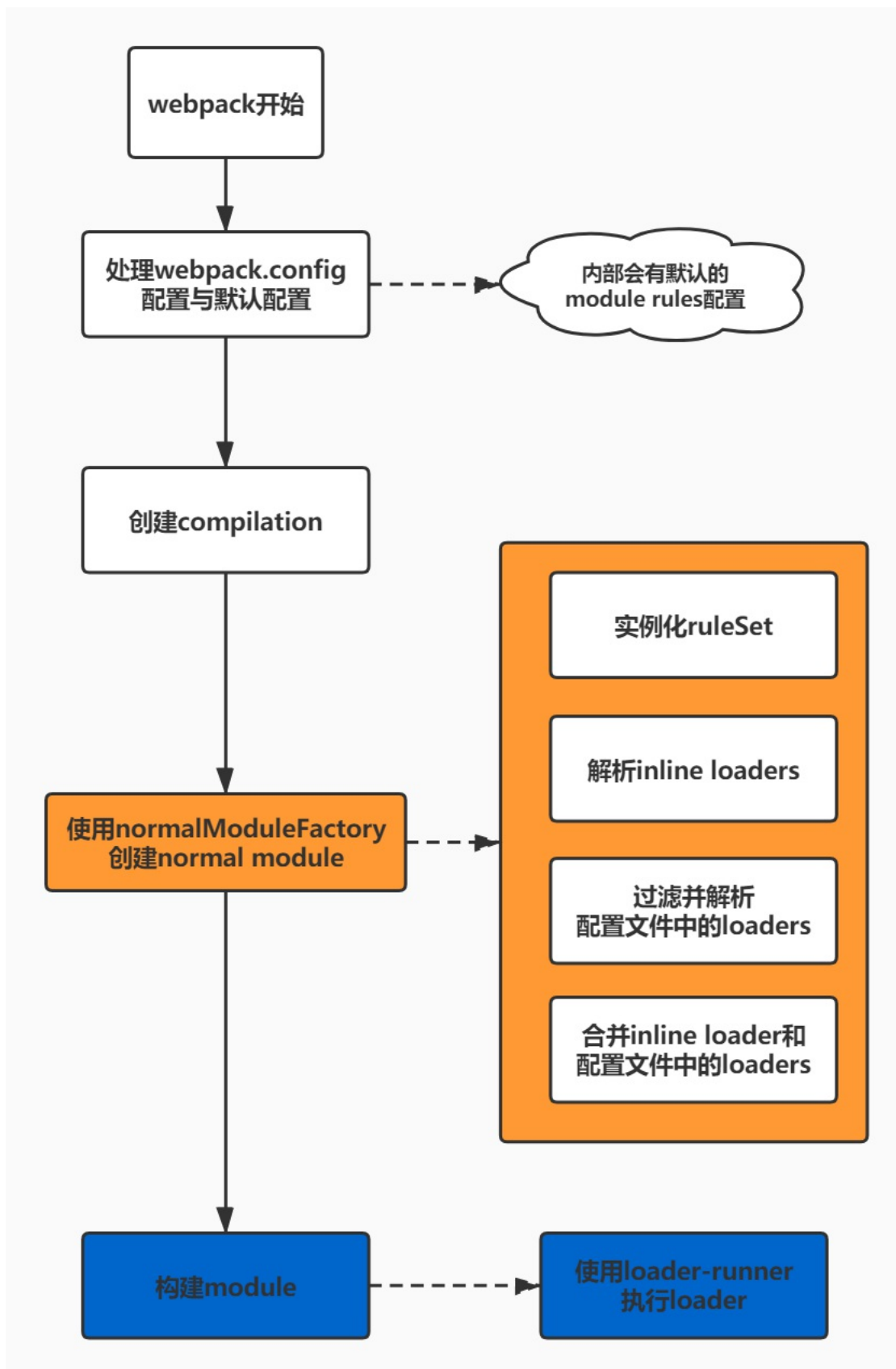
```
const result = this.ruleSet.exec({
    resource: resourcePath,
    resourceQuery
});
```

```
[
    {
        type: 'use',
        value: {
            loader: 'style-loader',
            options: {}
        },
        enforce: undefined
    },
    {
        type: 'use',
        value: {
            loader: 'css-loader',
            options: {}
        },
        enforce: undefined
    }
]
```

3.创建模块

- 实例化ruleSet (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModuleFactory.js#L115)
- RuleSet (https://github.com/webpack/webpack/blob/v4.43.0/lib/RuleSet.js)
- Compilation.moduleFactory.create (https://github.com/webpack/webpack/blob/v4.43.0/lib/Compilation.js)
- NormalModuleFactory.create (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModuleFactory.js#L373)
- resolver (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModuleFactory.js#L159-L371)

4.编译模块 #

- Compilation.buildModule (https://github.com/webpack/webpack/blob/v4.43.0/lib/Compilation.js#L1111)
- Compilation.module.build (https://github.com/webpack/webpack/blob/v4.43.0/lib/Compilation.js#L739)
- NormalModule.js.build (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModule.js#L427)
- NormalModule.doBuild (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModule.js#L287)
- NormalModule.runLoaders (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModule.js#L295)
- NormalModule.parser (https://github.com/webpack/webpack/blob/v4.43.0/lib/NormalModule.js#L482)

```
创建module
    │
    ▼
module.build
开始构建
    │
    ▼
创建
loaderContext对象
    │
    ▼
run-loader ──────────────────────────►
    │
    ▼
解析模块内容
```

loader-runner

```
loader-1.pitch                          loader-1
    │                                       ▲
    │ loaderIndex++              loaderIndex--│
    ▼                                       │
loader-2.pitch                          loader-2
    │                                       ▲
    │ loaderIndex++              loaderIndex--│
    ▼                                       │
loader-x.pitch                          loader-x
    │                                       ▲
    └──────► processResource ──────────────┘
```