

# JSF与SAF的兼容与比较

被章耿添加，被章耿最后更新于八月 13, 2015

- [JSF与SAF的兼容](#)
  - [JSF与SAF可以共存](#)
  - [JSF调用SAF（dubbo）](#)
  - [SAF（100及以上）调用JSF](#)
- [SAF如何升级到JSF](#)
  - [服务端先升级为JSF（推荐）](#)
  - [客户端先升级为JSF](#)
- [附：SAF和JSF比较](#)

## JSF与SAF的兼容

### JSF与SAF可以共存

JSF和SAF是独立的框架，可以同时存在，互不影响。

例如在pom.xml里一起引入。

```
<dependency>
  <groupId>com.jd</groupId>
  <artifactId>jsf</artifactId>
  <version>1.2.0</version>
</dependency>
<dependency>
  <groupId>com.jd</groupId>
  <artifactId>saf</artifactId>
  <version>1.1.0</version>
</dependency>
```

这样做的好处是，SAF调SAF，JSF调JSF，正常情况下不需要兼容调用。

当然，如果一定要兼容调用，我们也是支持的。

### JSF调用SAF（dubbo）

1.数据同步worker自动将注册到SAF的zookeeper注册中心的Provider同步到JSF注册中心。

2.连接JSF注册中心。

```
<jsf:registry id="jsfRegistry" index="i.jsf.jd.com" />
```

3.配置客户端。

- serialization配置为hessian或者java
- protocol配置为dubbo
- alias配置为 group:version （英文冒号隔开saf的group和version）  
如果服务端没有group，alias直接就是version

```
<jsf:consumer id="helloService2" interface="com.jd.testjsf.HelloService"
  serialization="hessian" alias="ZG110:0.0.1" protocol="dubbo" />
```

4.启动程序调用即可。

其它注意事项：

1. 调用SAF带token的服务

```
<jsf:consumer id="helloService2" interface="com.jd.testjsf.HelloService"
    serialization="hessian" alias="ZG111:0.0.1" protocol="dubbo" >
    <jsf:parameter key="jdtoken" value="test" />
</jsf:consumer>
```

- 2. JSF调用SAF1.0.8.1及以下，如果序列化方式不是hessian，服务端返回RpcException会无法反序列化。
- 3. JSF调用SAF的callback功能不支持。

## SAF（1.0.9及以上）调用JSF

- 1. 同样的，数据同步worker自动将注册到JSF注册中心的Provider同步到SAF的zookeeper注册中心。
- 2. 连接SAF注册中心。或者直连
- 3. 配置客户端。
  - version配置为alias
  - group不需要配置
  - protocol还是dubbo
- 4. 启动程序即可。

其它注意事项：

callback无法支持

原来服务端返回的RpcException将变成RuntimeException。

## SAF如何升级到JSF

原来使用SAF发布和调用服务，那么要升级为JSF怎么做呢？

### 服务端先升级为JSF（推荐）

推荐的做法是：

- 1. 首先同时发布SAF和JSF的服务端（就是引入JSF增加一份JSF的配置）。
- 2. 通知客户端从SAF升级为JSF，逐步迁移，直到没有SAF客户端
- 3. 服务端下线SAF接口（去掉SAF引入去掉SAF的配置）。

这样就实现了完整的升级切换。

### 客户端先升级为JSF

如果SAF客户端先升级，那么相当于JSF调SAF，参考[JSF调用SAF（dubbo）](#)

等服务端升级后，再修改配置普通的JSF配置。

## 附：SAF和JSF比较

功能	功能点	SAF（1.1.0）	JSF（1.0.0）
性能	默认序列化	hessian 字段顺序可随便	msgpack（不同类型tps提高10%-50%） 字段顺序不能变化

功能	功能点	SAF（1.1.0）	JSF（1.0.0）
feature	泛化调用	√	√
	异步调用	√	√
	异步回调		√
	跨语言-rest	√	√
	跨语言-http		√
	token认证	√	√
	方法限制		√
	连接监听		√
	调用压缩		√
	自定义过滤器		√
服务治理	服务上下线	√	√
	服务端动态分组		√
	客户端调用分组切换	√（配置规则）	√（页面操作更简便）
	配置下发		√
	黑白名单		√
	IP路由、参数路由	√	√
	方法调用	√（只测试环境开放）	√（线上也可以通过授权开放）
	服务降级		√
	并发控制		√
	调用次数控制	√	√
服务质量	监控配置	√	√
	服务端调用次数		√
	客户端收集耗时分布		√