Baas Backend as a service 后台即服务

当前后台大规模使用模板引擎或者Jsp在后端直接输出完备页面。

这很方便，后端开发自己可以直接在前台拿到任何想要的变量。

但这带来一个问题，前端开发不清楚后台的逻辑，也不掌握模板引擎的语法，这让前端页面中夹杂着各种额外的符号和变量。既不利于前端页面单机调试，也不利于前后端代码合并。最主要的问题是加大了前后端的耦合程度。修改任何地方都需要双方反复确认，提高了沟通成本，拖慢了开发效率。

尽管模板引擎和jsp在过去的网站建设中发挥了巨大的作用，但现在的趋势是放弃模板引擎，后端不再负责任何呈现的任务，仅仅返回前段需要的数据，即所有响应均为纯Json/XML格式字符串。

例如，展现一个贴吧的所有帖子

过去直接访问 localhost/forum/1/post ，后端会在查询完数据库后，将数据组合成html标签，使用模板引擎插入原本的html页面中，然后返回整个文档。

更好的做法是，访问 localhost/forum/1/ 后端返回未经处理的html页面，然后由前端js在页面加载过程中请求 localhost/api/forum/1/post 获得本贴吧帖子的信息，类似于：

[{“id”: 1, “title”: “test”, “content”: “Baas is trend”, “time”: ‘2016-06-13”},

{“id”: 2, “title”: “another”, “content”: “using json in response”, “time”: ‘2016-06-13”}

……

]

然后再由js将数据组装到html中呈现。

使用Baas架构十分有利于前后端解耦合，也并未增加前端工作量（使用Angular等前端框架时，这种方案几乎是0成本），后台逻辑也更加清晰，显然会提高开发效率。

缺点：不利于SEO，请求放在前端会增加请求次数，小幅降低页面呈现速度（会降低服务器负荷）。

在中小型网站，Restful API + Baas架构有很大的优势。