

01 | 架构到底是指什么？

2018-04-28 李运华



01 | 架构到底是指什么？

朗读人：黄洲君 09'19" | 4.27M

对于技术人员来说，“架构”是一个再常见不过的词了。我们会对新员工培训整个系统的架构，参加架构设计评审，学习业界开源系统（例如，MySQL、Hadoop）的架构，研究大公司的架构实现（例如，微信架构、淘宝架构）.....虽然“架构”这个词常见，但如果深究一下“**架构**”到底指什么，大部分人也许并不一定能够准确地回答。例如：

- 架构和框架是什么关系？有什么区别？
- Linux 有架构，MySQL 有架构，JVM 也有架构，使用 Java 开发、MySQL 存储、跑在 Linux 上的业务系统也有架构，应该关注哪个架构呢？
- 微信有架构，微信的登录系统也有架构，微信的支付系统也有架构，当我们谈微信架构时，到底是在谈什么架构？

要想准确地回答这几个问题，关键在于梳理几个有关系而又相似的概念，包括：系统与子系统、模块与组件、框架与架构。

系统与子系统

我们先来看维基百科定义的“系统”。

系统泛指由一群有关联的个体组成，根据某种规则运作，能完成个别元件不能单独完成的工作的群体。它的意思是“总体”“整体”或“联盟”。

我来提炼一下里面的关键内容：

1. 关联：系统是由一群有关联的个体组成的，没有关联的个体堆在一起不能成为一个系统。例如，把一个发动机和一台 PC 放在一起不能称之为一个系统，把发动机、底盘、轮胎、车架组合起来才能成为一台汽车。
2. 规则：系统内的个体需要按照指定的规则运作，而不是单个个体各自为政。规则规定了系统内个体分工和协作的方式。例如，汽车发动机负责产生动力，然后通过变速器和传动轴，将动力输出到车轮上，从而驱动汽车前进。
3. 能力：系统能力与个体能力有本质的差别，系统能力不是个体能力之和，而是产生了新的能力。例如，汽车能够载重前进，而发动机、变速器、传动轴、车轮本身都不具备这样的能力。

我们再来看子系统的定义。

子系统也是由一群有关联的个体所组成的系统，多半会是更大系统中的一部分。

其实子系统的定义和系统定义是一样的，只是观察的角度有差异，一个系统可能是另外一个更大系统的子系统。

按照这个定义，系统和子系统比较容易理解。我们以微信为例来做一个分析。

1. 微信本身是一个系统，包含聊天、登录、支付、朋友圈等子系统。
2. 朋友圈这个系统又包括动态、评论、点赞等子系统。
3. 评论这个系统可能又包括防刷子系统、审核子系统、发布子系统、存储子系统。
4. 评论审核子系统不再包含业务意义上的子系统，而是包括各个模块或者组件，这些模块或者组件本身也是另外一个维度上的系统。例如，MySQL、Redis 等是存储系统，但不是业务子系统。

模块与组件

模块和组件两个概念在实际工作中很容易混淆，我们经常能够听到类似这样的说法：

- MySQL 模块主要负责存储数据，而 Elasticsearch 模块主要负责数据搜索。

- 我们有安全加密组件、有审核组件。
- App 的下载模块使用了第三方的组件。

造成这种现象的主要原因是，模块与组件的定义并不好理解，也不能很好地进行区分。我们来看看这两者在维基百科上的定义。

软件模块 (Module) 是一套一致而互相有紧密关连的软件组织。它分别包含了程序和数据结构两部分。现代软件开发往往利用模块作为合成的单位。模块的接口表达了由该模块提供的功能和调用它时所需的元素。模块是可能分开被编写的单位。这使它们可再用和允许人员同时协作、编写及研究不同的模块。

软件组件定义为自包含的、可编程的、可重用的、与语言无关的软件单元，软件组件可以很容易被用于组装应用程序中。

可能你看完这两个定义后一头雾水，还是不知道这两者有什么区别。造成这种现象的根本原因是，模块和组件都是系统的组成部分，只是从不同的角度拆分系统而已。

从逻辑的角度来拆分系统后，得到的单元就是“模块”；从物理的角度来拆分系统后，得到的单元就是“组件”。划分模块的主要目的是职责分离；划分组件的主要目的是单元复用。其实，“组件”的英文 component 也可翻译成中文的“零件”一词，“零件”更容易理解一些，“零件”是一个物理的概念，并且具备“独立且可替换”的特点。

我以一个最简单的网站系统来为例。假设我们要做一个学生信息管理系统，这个系统从逻辑的角度来拆分，可以分为“登录注册模块”“个人信息模块”“个人成绩模块”；从物理的角度来拆分，可以拆分为 Nginx、Web 服务器、MySQL。

框架与架构

框架是和架构比较相似的概念，且两者有较强的关联关系，所以在实际工作中，这两个概念有时我们容易分不清楚。参考维基百科上框架与架构的定义，我来解释两者的区别。

软件框架 (Software framework) 通常指的是为了实现某个业界标准或完成特定基本任务的软件组件规范，也指为了实现某个软件组件规范时，提供规范所要求之基础功能的软件产品。

我来提炼一下其中关键部分：

1. 框架是组件规范：例如，MVC 就是一种最常见的开发规范，类似的还有 MVP、MVVM、J2EE 等框架。
2. 框架提供基础功能的产品：例如，Spring MVC 是 MVC 的开发框架，除了满足 MVC 的规范，Spring 提供了很多基础功能来帮助我们实现功能，包括注解 (@Controller 等)、

Spring Security、Spring JPA 等很多基础功能。

软件架构指软件系统的“基础结构”，创造这些基础结构的准则，以及对这些结构的描述。

单纯从定义的角度来看，框架和架构的区别还是比较明显的，框架关注的是“规范”，架构关注的是“结构”。框架的英文是 Framework，架构的英文是 Architecture。Spring MVC 的英文文档标题就是“Web MVC framework”。

虽然如此，在实际工作中我们却经常碰到一些似是而非的说法。例如，“我们的系统是 MVC 架构”“我们需要将 android app 重构为 MVP 架构”“我们的系统基于 SSH 框架开发”“我们是 SSH 的架构”“XX 系统是基于 Spring MVC 框架开发，标准的 MVC 架构”.....

究竟什么说法是对的，什么说法是错的呢？

其实这些说法都是对的，造成这种现象的根本原因隐藏于架构的定义中，关键就是“基础结构”这个概念并没有明确说是从什么角度来分解的。采用不同的角度或者维度，可以将系统划分为不同的结构，其实我在“模块与组件”中的“学生管理系统”示例已经包含了这点。

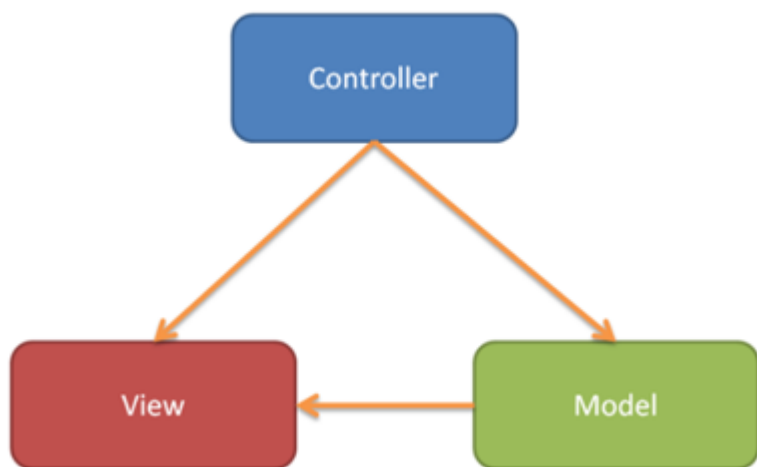
从业务逻辑的角度分解，“学生管理系统”的架构是：



从物理部署的角度分解，“学生管理系统”的架构是：



从开发规范的角度分解，“学生管理系统”可以采用标准的 MVC 框架来开发，因此架构又变成了 MVC 架构：



这些“架构”，都是“学生管理系统”正确的架构，只是从不同的角度来分解而已，这也是 IBM 的 RUP 将软件架构视图分为著名的“4+1 视图”的原因。

重新定义架构

参考维基百科的定义，我将架构重新定义为：软件架构指软件系统的顶层结构。

这个定义看似很简单，但包含的信息很丰富，基本上把系统、子系统、模块、组件、架构等概念都串起来了，我来详细解释一下。

首先，“系统是一群关联个体组成”，这些“个体”可以是“子系统”“模块”“组件”等；架构需要明确系统包含哪些“个体”。

其次，系统中的个体需要“根据某种规则”运作，架构需要明确个体运作和协作的规则。

第三，维基百科定义的架构用到了“基础结构”这个说法，我改为“顶层结构”，可以更好地区分系统和子系统，避免将系统架构和子系统架构混淆在一起导致架构层次混乱。

小结

今天我为你梳理了与架构有关的几个容易混淆的概念，包括系统与子系统、模块与组件、框架与架构，解释了架构的定义，希望对你有所帮助。

这就是今天的全部内容，留一道思考题给你吧。你原来理解的架构是如何定义的？对比我今天讲的架构定义，你觉得差异在哪里？

欢迎你把答案写到留言区，和我一起讨论。相信经过深度思考的回答，也会让你对知识的理解更加深刻。（编辑乱入：精彩的留言有机会获得丰厚福利哦！）



版权归极客邦科技所有，未经许可不得转载

精选留言



公号-Java大后端

696

架构是顶层设计；框架是面向编程或配置的半成品；组件是从技术维度上的复用；模块是从业务维度上职责的划分；系统是相互协同可运行的实体。

2018-04-28

作者回复

精炼！

2018-04-28



公号-Java大后端



搬砖的：“头，我们要造什么？”；（做什么系统？）

工程师：“龙之梦商城”；（XXX系统，比如微博系统）

搬砖的：“图纸画出来了嘛？”；（架构是怎么设计的？）

工程师：“一楼主要以女性消费为主体、二楼以大众娱乐为主体、三楼以美食为主体”；
（相当于微博系统中的各个子系统，比如评论子系统、动态子系统、消息子系统）

搬砖的：“头，说人话”；

工程师：“一楼有卖衣服、化妆品的，二楼有唱歌、看电影的，三楼有吃的”；（【模块】按照逻辑区分，比如存储数据模块、搜索模块、消息推送模块）

搬砖的：“有没有很知名的店啊？”；

工程师：“有的，一楼有香奈儿、优衣库...、二楼有好乐迪、万达影院...、三楼有海底捞、避风塘...”；（【组件】按照物理区分，存储数据模块对应Mysql、搜索模块对应ElasticSearch、消息推送模块对应Kafka）

搬砖的：“对了，头，商城大门有啥需要叮嘱的施工规范不？或有啥简化施工工艺的新技术嘛？”；（有框架的可以用吗？）

工程师猛吸了一口烟，把烟头扔在地上，用皮鞋左右撵了两下，缓缓从嘴里崩出四个字。

“老样子吧”。（Spring全家桶甩起来）

2018-04-28

作者回复

极客时间卧虎藏龙，里面的用户个个都很有才，口才又好，长得又帅，我超喜欢😄

2018-04-28



Ivan

👍 53

框架是规矩，架构是按照规矩做规划。系统是学校，子系统是班级，模块是学生老师，组件是课桌椅。每一层级的作用意义和范围不一样，要求和可复用度也不一样

2018-04-28



杜晓东

👍 33

我们要做的东西都能抽象为一个系统，架构既可做动词也可做名词，作为动词就代表系统的设计，作为名词就代表系统的表现形式。

所以架构首先继承了系统的属性：

0、系统整体有价值

1、由多个有关系的个体组成

2、涌现，整体大于个体之和，也就是文中说的“流”出来的新的价值

系统的架构从无到有由人来执行，所以也具备人思考和交流的属性：

0、思维带宽较小，不能同时考虑很多事情，需要把系统做分解（模块和组件）

1、线性思想和交流，多维关系要降级到二维关系（4+1视图等多种方法论）

2、人作为个体的个性化-也即多样性，架构没用统一标准，适合自己/团队/公司最重要。

2018-04-28

作者回复

你已经开始解剖架构的本质了，后面章节会讨论这个话题。

通常“架构”还是用作名词，动词就用“架构设计”，有的观点用“构架”，有点拗口和容

易混淆，所以我一般宁愿用“架构设计”

2018-04-28



blue

👍 24

系统与子系统：系统是由一系列有关联，按特定规则组成的个体，并且产生新的能力，而系统与子系统则是观察的交角度不同

模块与组件：模块是从逻辑角度去看待，而组件是从物理角度去看待

框架与架构：框架是规范也是约束，可以理解为封闭性的话题，定义好，让别人如何去使用，而架构是一种结构，是一种开放性的话题，如何去设计组织架构，如何让架构更具有拓展性，减少沟通错误成本

2018-04-29

作者回复

提炼了精华，赞

2018-04-29



今夕是何年

👍 6

架构是顶层设计，框架是具体实现。

2018-05-01



tick

👍 6

微信是系统，微信的登录是子系统，学生管理是系统，学生管理的登录就是模块，这是为什么呢，模块和子系统啥区别？

2018-04-28

作者回复

子系统是独立运行的，模块是子系统的逻辑组成部分，如果学生管理系统规模很大（例如在线学校），需要支撑每秒上万的登录请求，那么学生管理的登录模块一样可以升级为子系统。

2018-04-28



feiyue

👍 6

大道至简。架构是宏观整体，框架是微观组成。

2018-04-28



余红松-北京

👍 5

阅读笔记

- 关于系统：

系统和子系统其实都是系统，只不过在本业务里边的分层不同，只有顶层为系统，其他都为子系统，就像目录树一样，看你目录最深有多少层，

- 模块向虚(逻辑)，组件向实(代码)。

- 框架向虚(规范)，架构向实(结构)。

2018-05-06



CHaNniNG

👍 5

关于华仔的思考题，本人刚刚接触架构，之前作为开发人员理解的架构不是很深入，基本是按照各个技术组件的组合方式来理解的，架构做得好不好，主要看技术组件能否组合得好，并满足业务特性；今天听了华仔的讲解，发现我之前忽略的业务模块的组成这一方面而单单考虑技术点了，架构其实是可以从多个角度来进行阐述的。华仔的讲解让我对架构的概念有了更加清晰和全面的了解，谢谢华仔

2018-04-28



patrick_momo

👍 4

我理解的架构，是通过不同分工合作和识别不同生命周期为了一个组织或特定的目的而形成的树状结构，这种结构有利于在有限的时间内使组织内的角色完成相关的工作，从而达到业务增长的目的。而作者的这个架构应该更多地理解为软件架构，但是局限在顶层是否合适，还要看后续的文章内容阐述。

2018-04-28



Ken

👍 3

系统架构主要描述一个系统有哪些子系统构成，每个子系统的职责，以及子系统之间如何交互。

从技术角度来说，系统架构也会描述系统的技术栈，技术选型，以及高层次的业务流程。

2018-04-28

作者回复

这种方式适合描述较复杂的后台架构，有的系统不一定有子系统的概念，但它们一样有架构，例如我们可以说mysql的架构，jvm的架构，linux的架构

2018-04-28



felix

👍 3

之前理解的架构就包含2部分，逻辑架构和物理架构，现在知道了还有开发架构

2018-04-28



Suclogger

👍 3

我觉得架构的定义也可以从逻辑和物理上拆分，从逻辑视角出发，可以认为是业务的各个模块的定义，从物理视角出发，可以认为是各个模块的实现之间是一种什么方式交互合作。感谢作者的精彩分享，就是演讲者对专有单词的读音还要改善，比如nginx。

2018-04-28



弄花香满衣

👍 3

我理解的架构包括网络、机房、存储、容器、系统实现；其中系统实现约等于框架，比如常用的MVC框架、redis、RPC等等

2018-04-28



海格

👍 3

老师您好，我是一名运维，我讲讲对运维架构理解吧，运维架构指，业务系统所用到的中间件，在运维视角合理的规划，实现可靠性，高性能的为业务系统提供支撑功能。还有一个就

是把运维的日常工作柔和起来做一个运维平台，这个平台实现简单，高效的完成运维的日常工作听完老师今天的分享，我的理解就是把一个系统更加细化具体的分析出来，然后逐个实现，再整合，完成我们想让系统完成的功能。

2018-04-28



joedong

👍 2

个人觉得架构是复杂事物构建过程中产生的需求，而且不仅是效率上的，还是必要性上，都需要架构的支撑。以此引申出的架构设计，架构搭建，架构维护升级等架构相关工作。类比建筑行业里面的架构，如果没有好的架构支撑，就无法搭建稳固的建筑"产品"。而且架构将不可变的要素跟可变的要素进行了分离，使到建筑的例如外观，装饰，内部空间等的变化(短小时内)成为可能。

所以回到软件产品(系统)，架构的引入，确实也起到了类似的作用。而且越通用的架构越低层，但是同时也未必产品构建效率和产品整体性能较佳，所以需要根据不同类型的产品进行架构设计。比如网页显示型产品(web)跟功能调用型(输出类似json等数据)的，就会有不同的架构设计。

2018-04-29

作者回复

软件架构和建筑架构貌似类似，实质上有本质区别，后面会阐述

2018-04-30



Tunan

👍 2

我理解的架构就是功能划分，最后让整个软件成为架子，模块是乐高积木。在遵循已定接口规范下能按照产品需求自由组合模块之间的IO以达到需求效果。

2018-04-28

作者回复

这种理解适合业务系统架构设计，对于一些高性能高可用的架构不是很合适，单纯的功能划分并不能够保证高性能高可用，只能保证功能完成。

2018-04-28



常江舟

👍 2

我原来理解的架构

1.如果是单体应用这种，拿MVC来说，不管是前端还是后台，我说是MVC风格，各种前后端框架比如spring mvc,angular是各种风格的实现。这个时候别人问我软件的架构是什么，我说就是单体应用架构，顶多再描述一下，系统是动静分离，还有用到了哪些组件，原来我对组件的定义就是nginx,mysql,tomcat这些。这一点和今天老师讲的差不多

2.如果是分布式系统这种复杂的软件系统，别人问及架构。我会比较多地描述系统该系统整理分为哪几块，对应老师今天说的子系统的定义，而没有像老师一样，从业务逻辑和物理视图，开发角度等不同的面去描述。如果问及具体的某一块，我会比较多地描述其上下游依赖关系，在高性能，高可用方面怎么做的。总结一下，感觉自己的定义比较含糊，除非写文档的时候才会做几张业务架构图，和中间件相关的技术架构图。老师给的定义很清楚，而工程上也确实需要这样的定义，真是学到了。特别是联想到RUP那句话，让我恍然大悟！

2018-04-28



撒哈拉的海马

👍 2

我理解的架构，针对特定的需求，选择某种技术或某些技术簇，并设计出一套解决此问题的串联方法和规则，用来定义各系统和子系统的能力以及它们之间的交互串联规则说明和方法。

2018-04-28