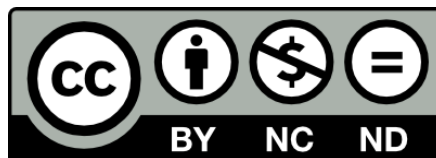




Android System Architecture Design

David Lau • China



本作品采用知识共享 署名-非商业性使用-禁止演绎 3.0 中国大陆 许可协议进行许可。
要查看该许可协议，可访问<http://creativecommons.org/licenses/by-nc-nd/3.0/cn/>

您可以自由：

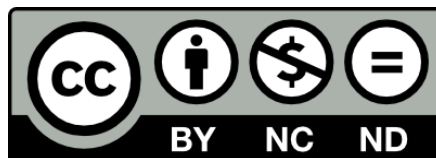
复制、发行、展览、表演、放映、广播或通过信息网络传播本作品

惟须遵守下列条件：

- 署名 — 您必须按照作者或者许可人指定的方式对作品进行署名。
- 非商业性使用 — 您不得将本作品用于商业目的。
- 禁止演绎 — 您不得修改、转换或者以本作品为基础进行创作。

© Copyright 2013 These slides created by :刘智勇(David Lau)

Email: zhiyong.liu@aliyun.com Latest Update: 2013-09-08



This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

You are free:

to Share — to copy, distribute and transmit the work

Under the following conditions:

Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Noncommercial — You may not use this work for commercial purposes.

No Derivative Works — You may not alter, transform, or build upon this work.

© Copyright 2013 These slides created by :刘智勇(David Lau)

Email: zhiyong.liu@aliyun.com Latest Update: 2013-09-08

纲要



- 软件设计与思维活动
- 架构设计的境界
- 架构设计？
- **API**及如何设计一个好的**API**

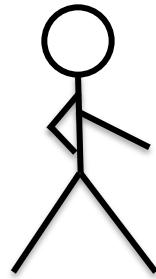


- 软件设计与思维活动

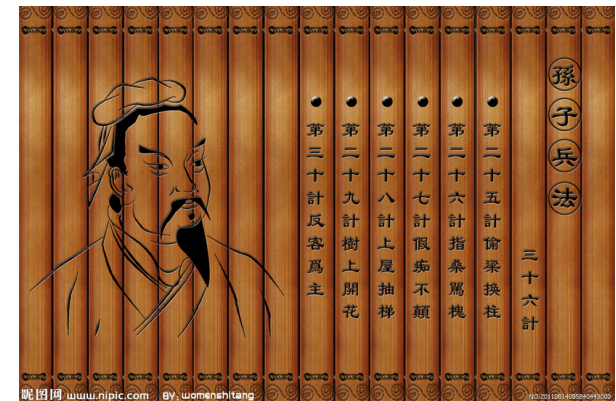


將者，智、信、仁、勇、嚴也

——《孫子兵法》



架构师，谋全局者也

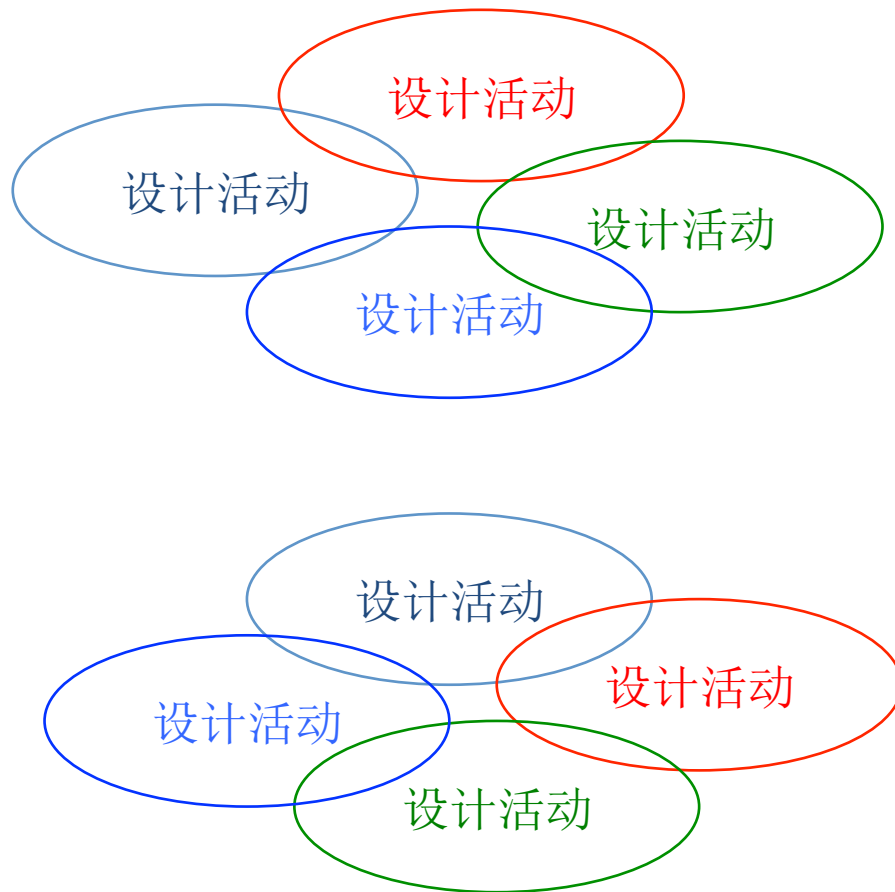


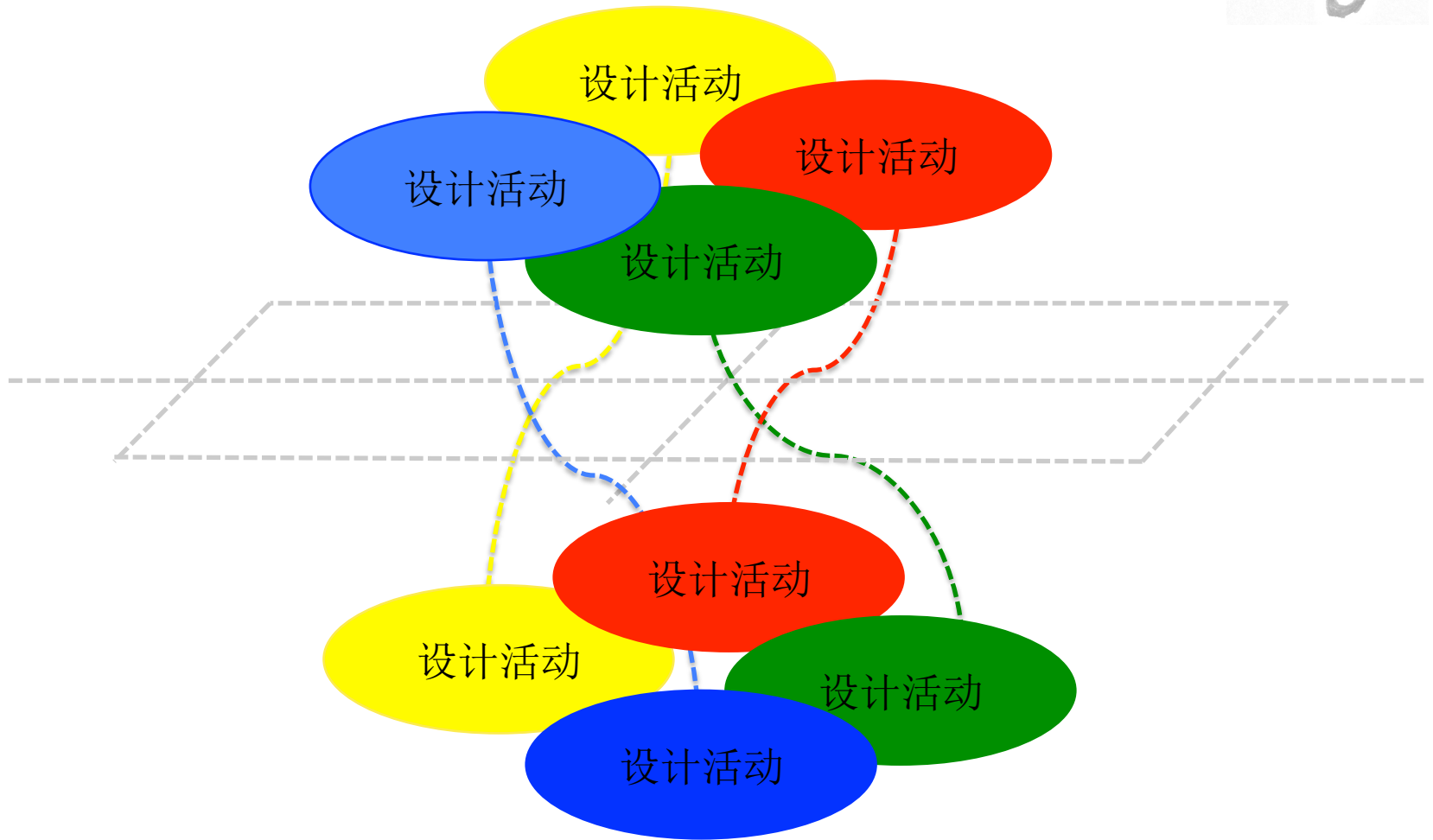
软件设计活动与思维活动



- 基于软件设计活动的分析和软件思维活动的研究，为了更准确地寻找到软件设计活动中影响设计结果的因子，笔者假设设计活动是线性条件下，基于思考力、创造力、设计力和执行力的组合，在四力的构成中，其对应的落地行为又分别对应着问题思考、方法描述、模型塑造以及代码编程，当然分析是基于线性的，然而设计活动是由多个设计小活动组合而成，因此形成在设计大范围内的，四力作用，在软件设计活动的空间和时间上，四力的作用，又不分时间先后，不分地点，原则上在设计活动中，侧重注意四力的相互作用力。

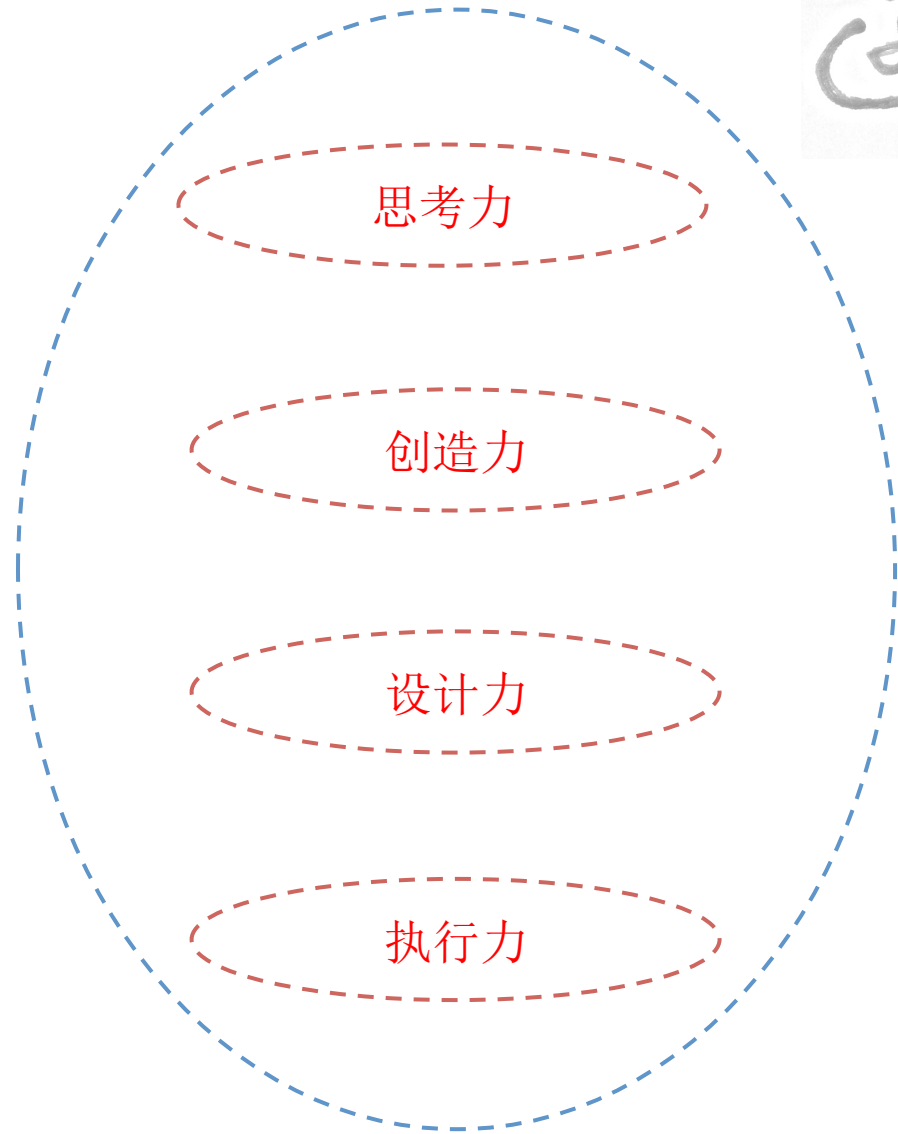
设计活动的组合形态

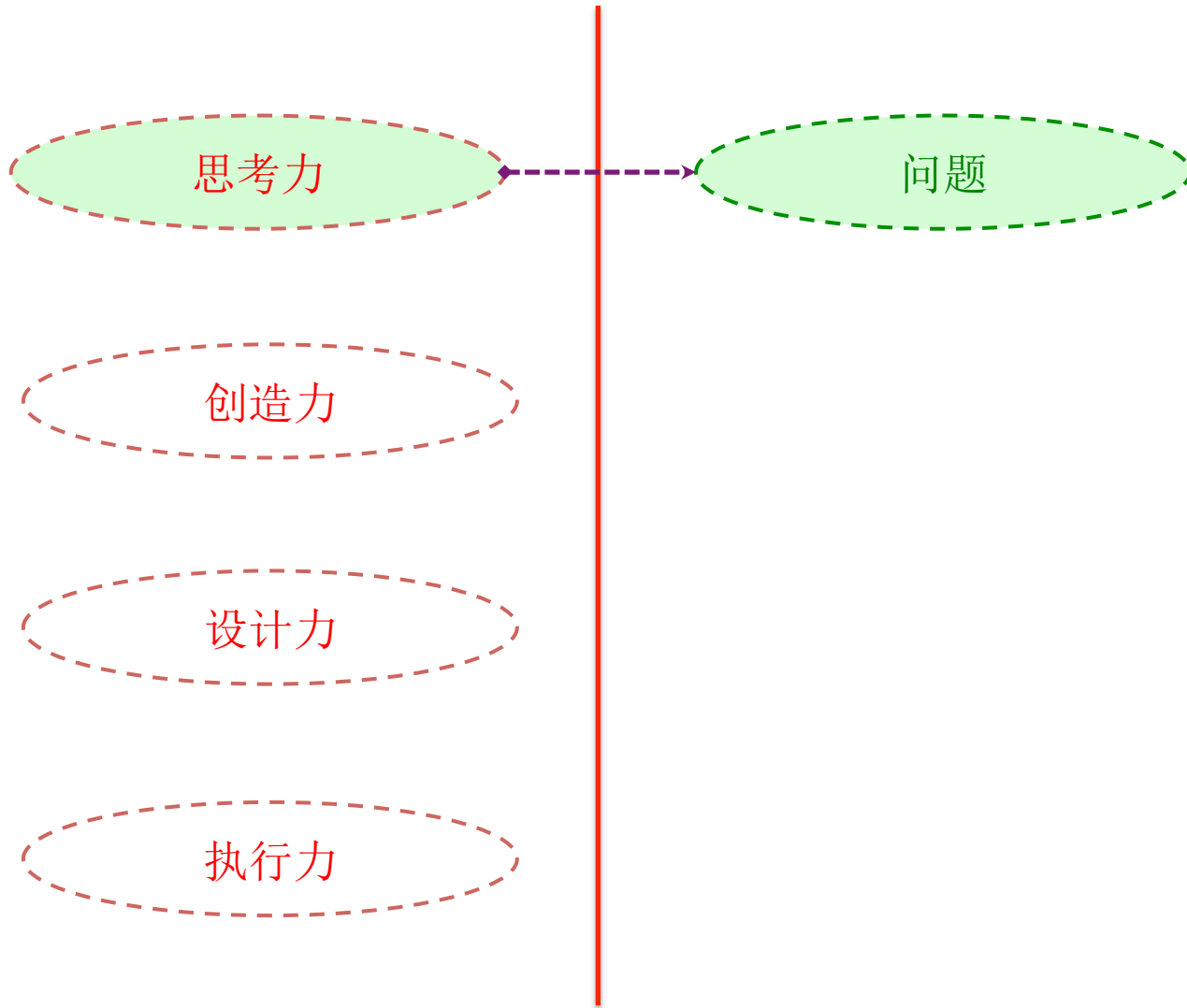


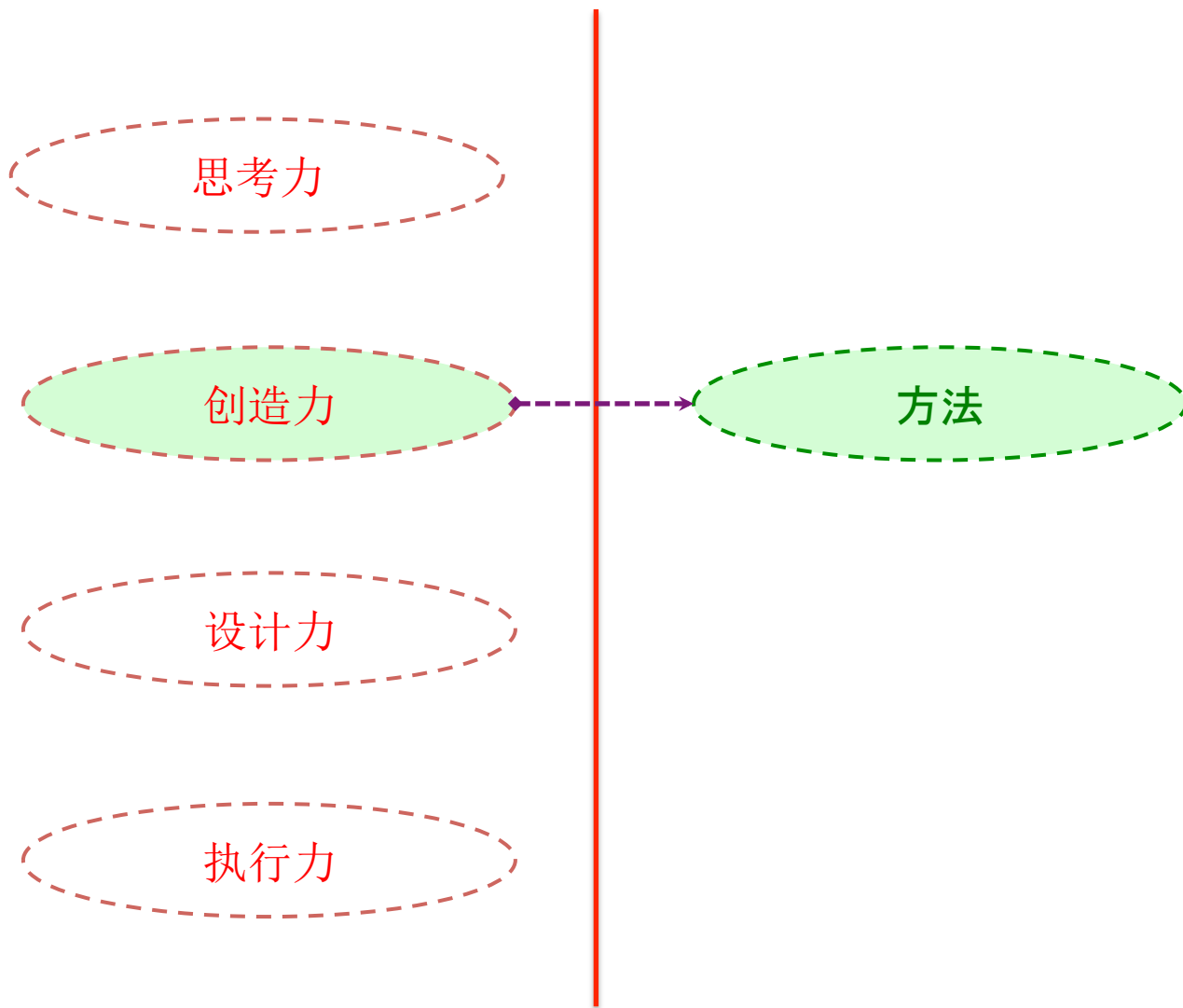




单一设计活动 与 单一思考活动









思考力

创造力

设计力

执行力

模型





思考力

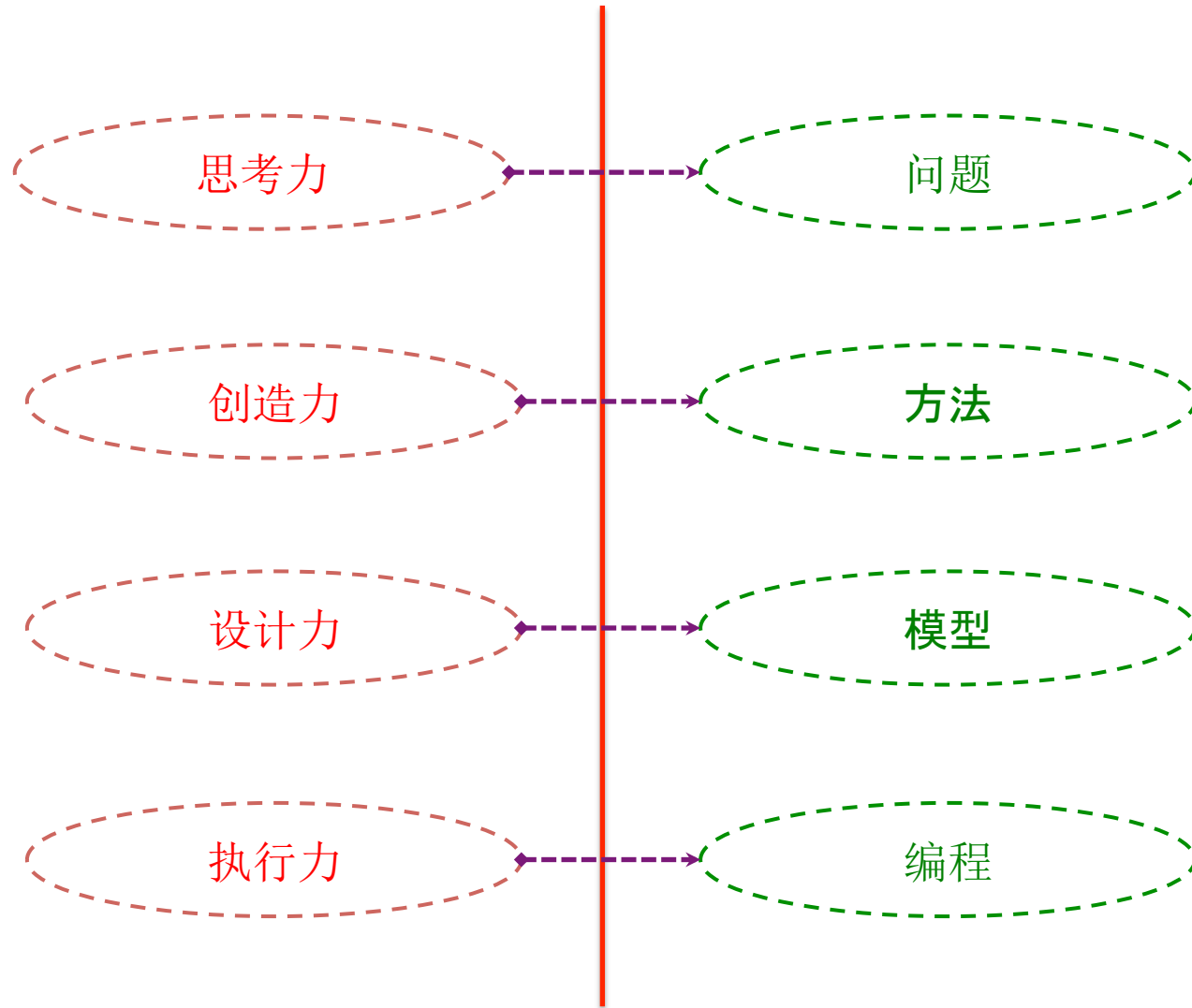
创造力

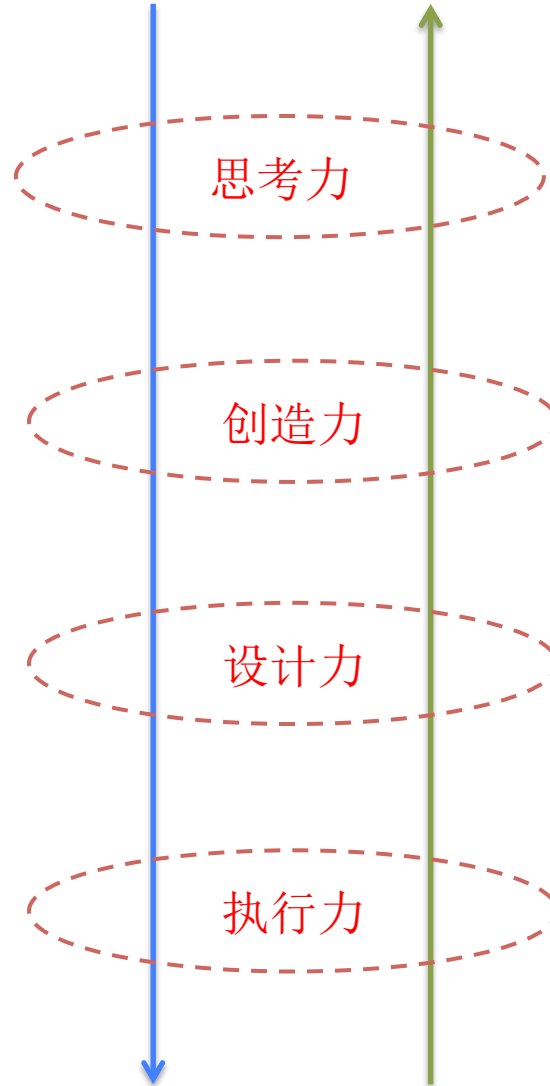
设计力

执行力



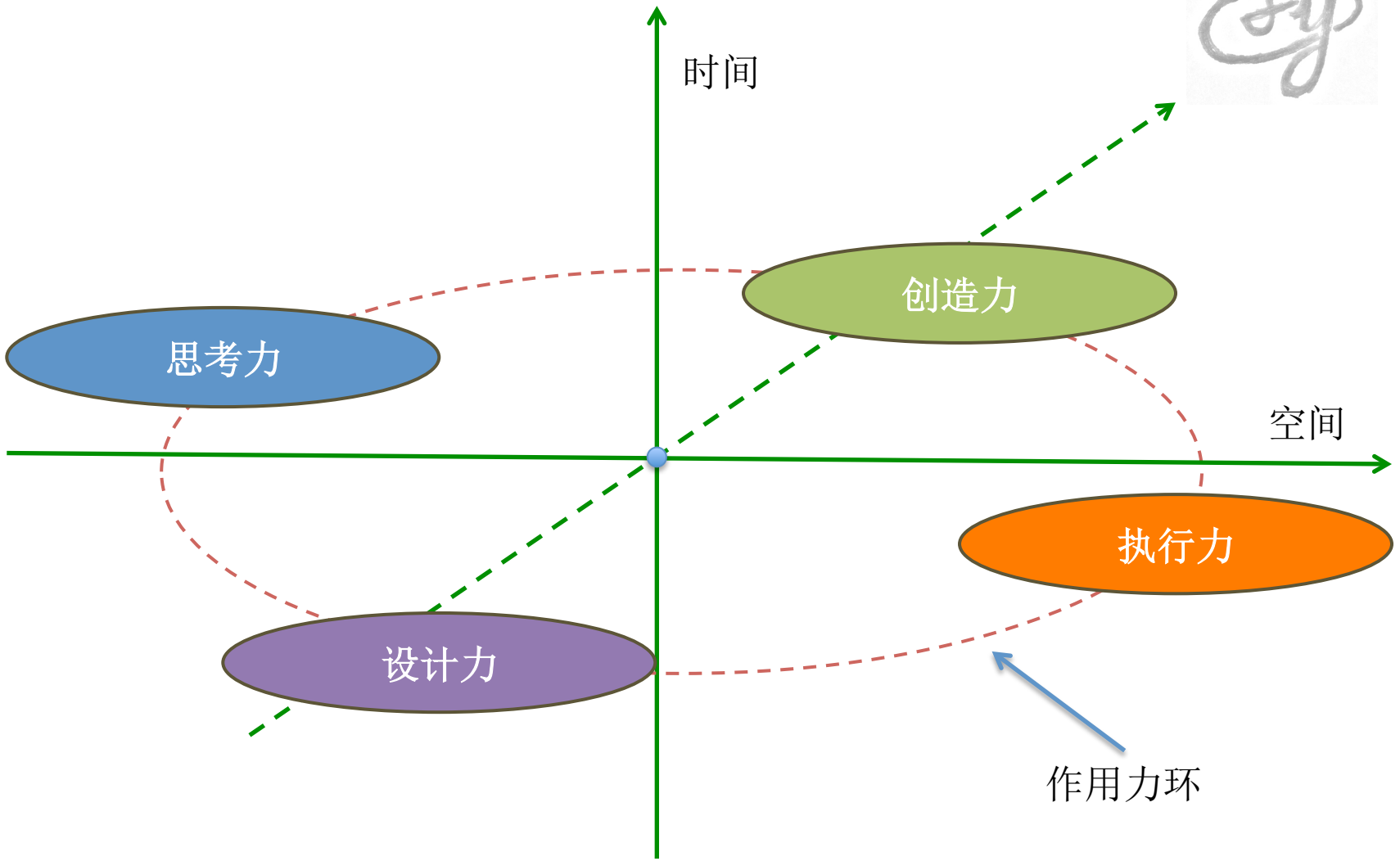
编程





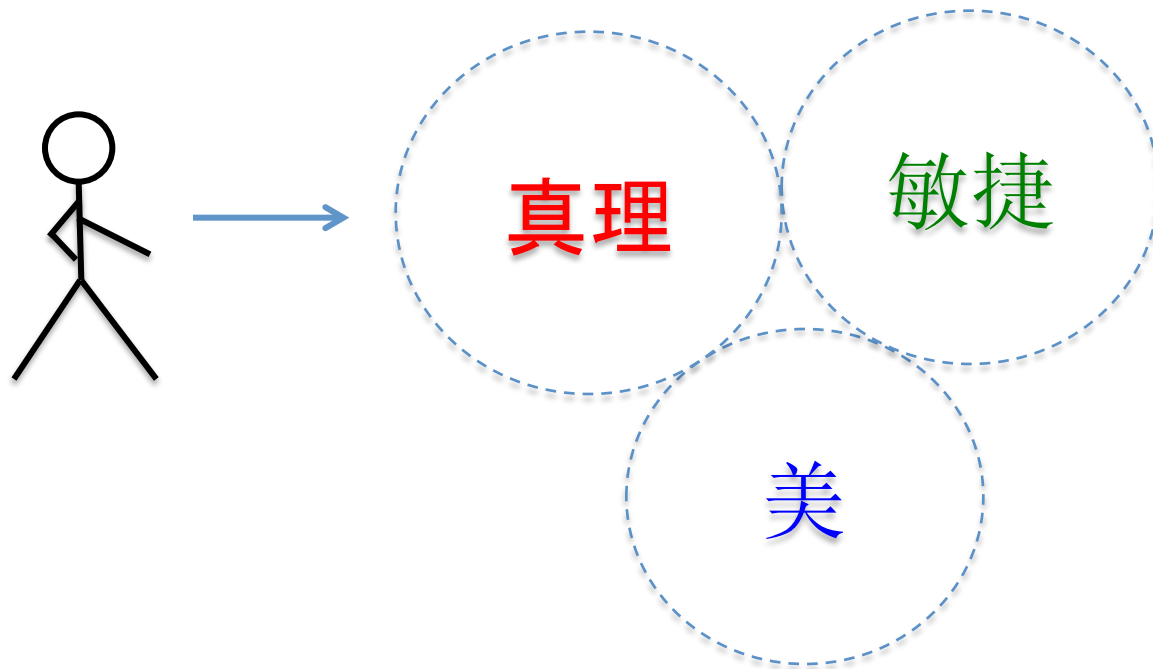
→ 思考引力

→ 执行反作用力





- 架构设计的境界



设计的境界：真、简、美



计算机的世界实际和我们真实的世界很接近，就好像我们宗教和人性中会歌颂真善美一样，计算机科学和软件工程中也同样存在着这样的追求，JAROSLAV TULACH认为是

“Beauty, Truth, and Elegance”，就中文和哲学而言，我认为这种追求是真、简、美，真为真理，简为简捷，美为美丽。

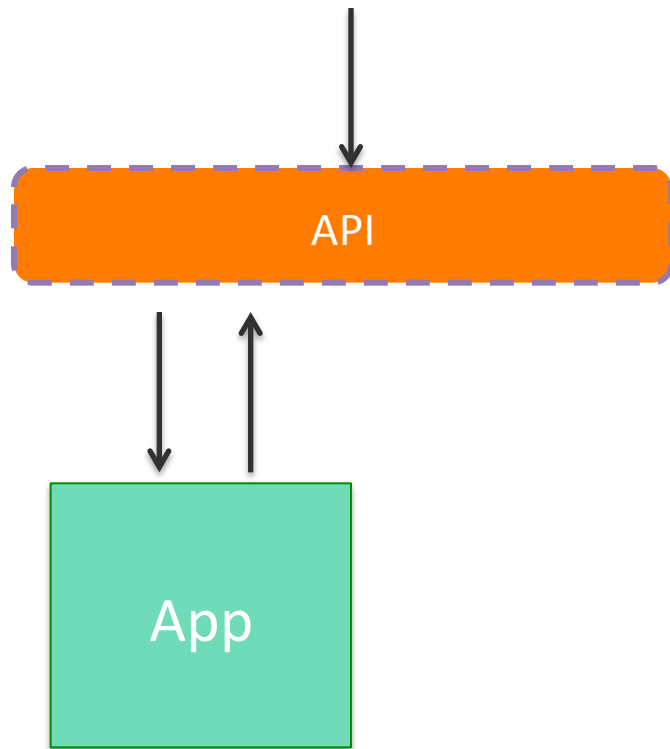
能做到？



- 对于工程师而言，所要做的事情不仅仅是信仰的追求，现实的对于他必须要求在产品的发布时间内，完成推出他的软件产品，对于美的追求，即使是在产品最后的阶段，工程师心理上不会放弃，因而工程师们会表现出非常想修复、修补重大漏洞，再会存在。但是，事实上，这种单纯的信仰追求在现实中并不是工程师们在现实中追求的目标。那不是说，在现实产品研发中，“真、简、美”需要离我们远一点，是不是工程师就不需要“真、简、美”，有没有某一个领域需要他们的存在，很显然，在我们恰当区分设计与执行之后，放在思考多维空间，这样的追求会时时刻刻，无处不在影响到工程师的设计产品。比如，在我们面对更多一无所知的Android模块代码，Android架构设计就是以“真简美”为设计灵魂的。



● 架构设计

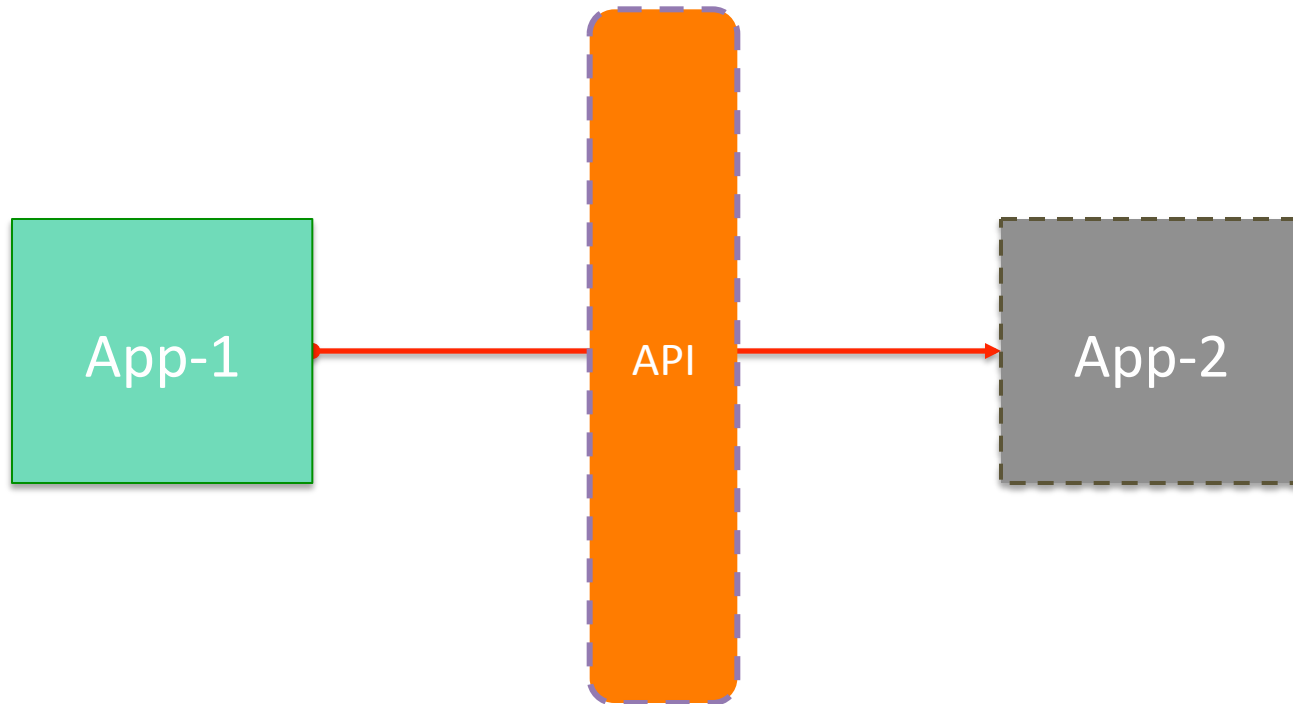


架构设计？



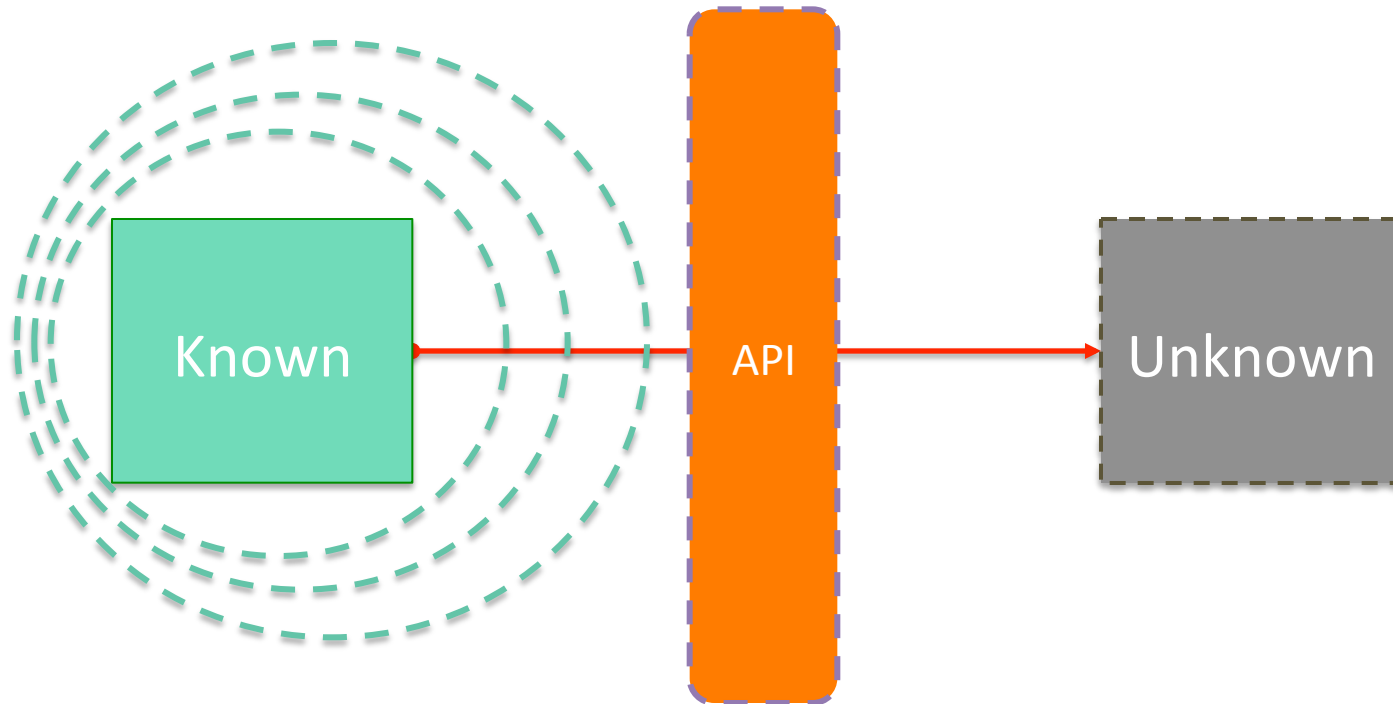


架构设计是：





这是合理的架构设计吗？



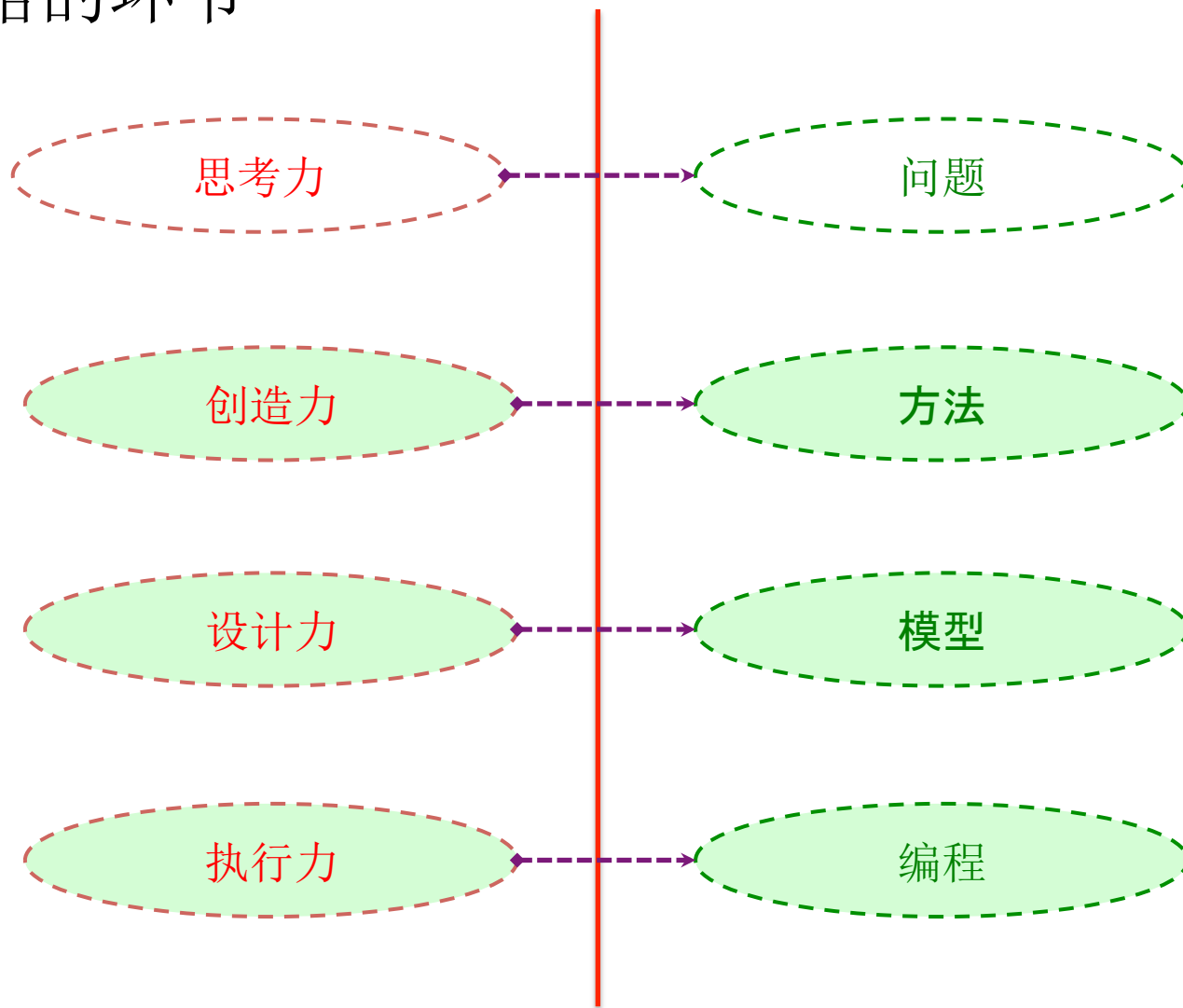
我们在开发微软视窗的应用程序时，不需要知道微软平台的框架代码，我们就可以开发应用程序。

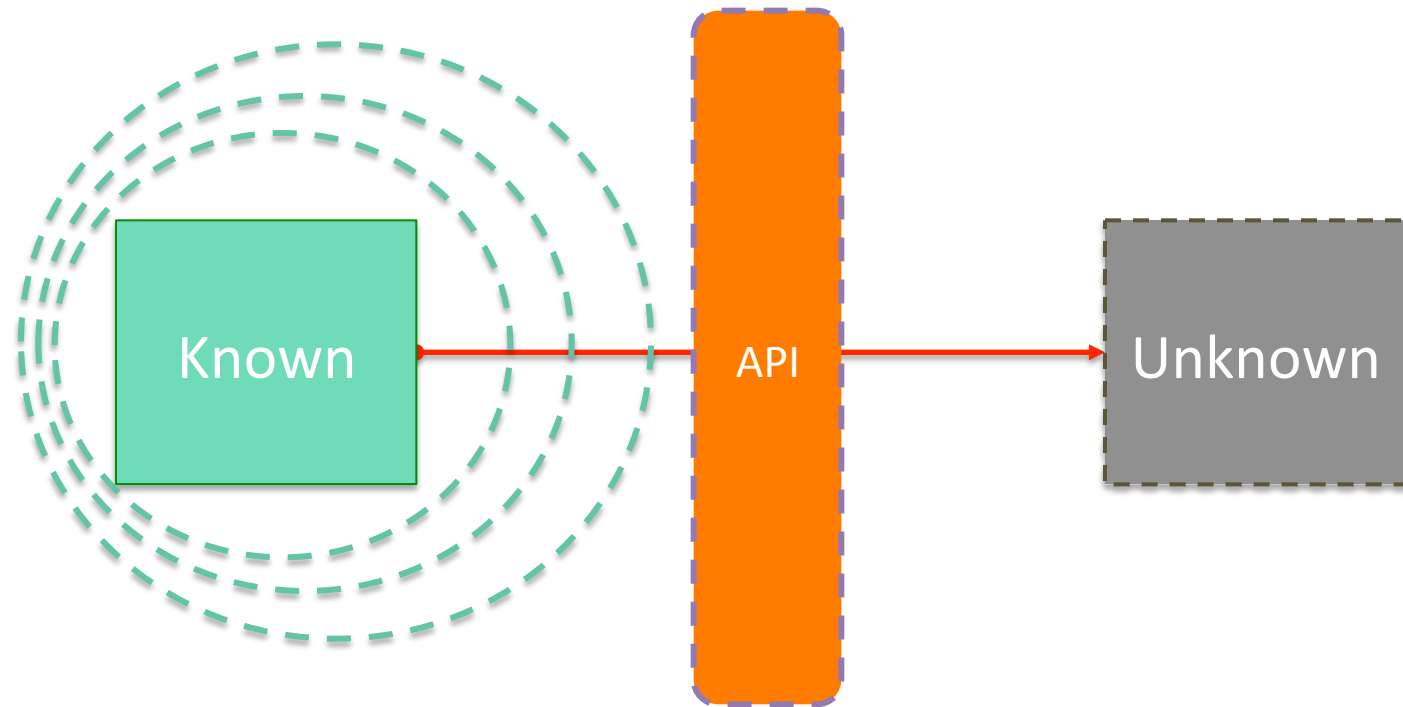
如果我们要成为一个优秀的程序员，我们只需要知道找到微软发布的API文档，通过努力学习，我们就可以灵活地应用API来开发应用程序。



架构设计 与 程序设计的 区别

被忽略的环节





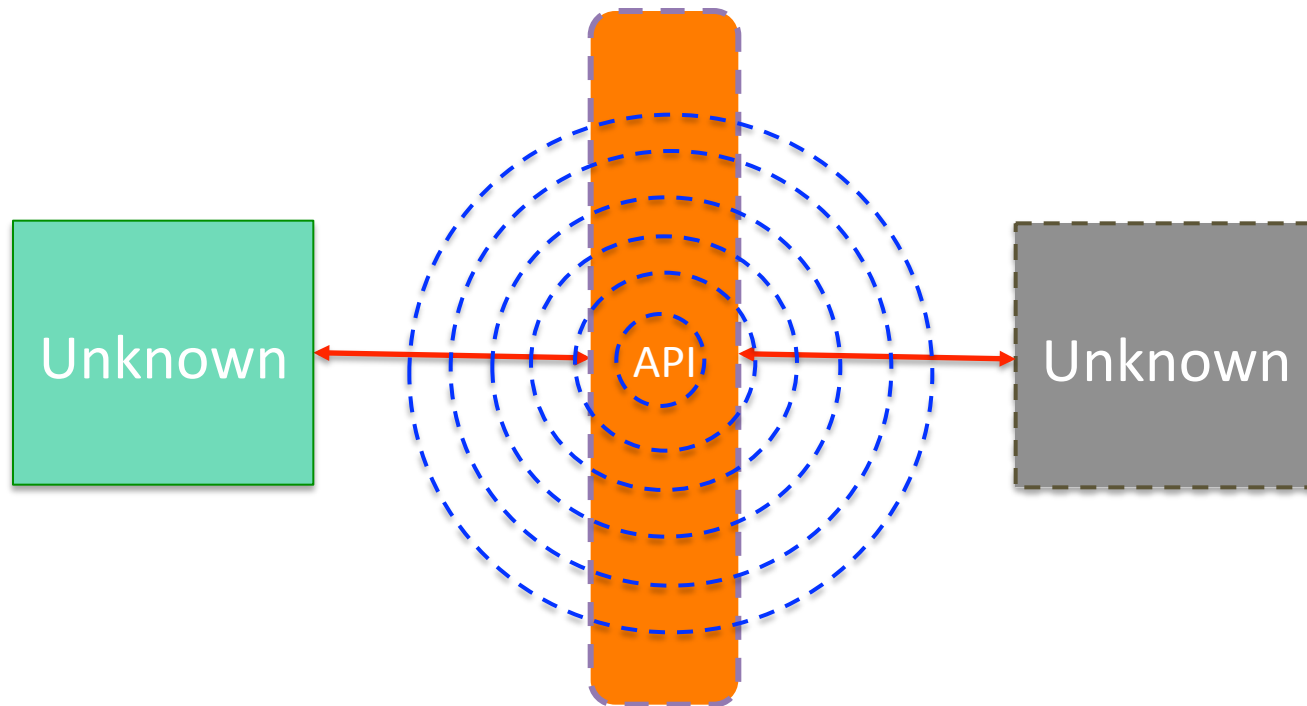
面对封闭的框架，我们开发程序时，每个人对框架的理解都会不同，对于我们所调用的基类、接口，其背后的机制始终都是一知半解，如此一来，我们的设计和求知，必然会停留在“用”的驱动层面，就好像我们在使用手机，但实际上我们并不能清楚完整解释智能手机是如何工作？这对于专业人士是不允许的，特别是对于谋划项目全局的架构师来说，更是如此。



然而，我们如果要将所有的Android代码都阅读和掌握，显然是不可能的，我们所能阅读和掌握的Android代码也只不过是其冰山下一角而已，如此一来我们就只有对Android中的框架或C/C++库进行抽象封装，用抽象的API隐藏所有的复杂性，请注意，架构师的焦点是需要做的是将原有的微软封闭式的开放方式，改进为一种隔离式的框架设计模式，这样的设计模式在Android随处可见，你可以不知道API两端的代码，但却在设计整个框架。



- **API**及如何设计一个好的**API**





-----> 软件工程管理的需要

-----> 模块化程序

-----> 项目团队沟通

-----> 经验编程

如何设计一个好的API



- Do Not Expose More Than You Want
- Code Against Interfaces, Not Implementations
- Use Modular Architecture
- Separate APIs for Clients and Providers
- Keep Testability in Mind
- Cooperating with Other APIs
- Runtime Aspects of APIs
- Declarative Programming



感谢JAROSLAV TULACH,
本章受惠于JAROSLAV TULACH的智慧, 谢谢!

参考资料:

《Practical API Design-Confessions of a Java Framework Architect》

About Me



- I have been working as a product-designer specializing in software/Web application design and development. I am passionate about mobile application development and became interested in Android programming when the platform was launched by Google. Thus I was not programming on Android projects, I spent spare time reading technical blogs, researching, analyzing, and testing mobile applications, as a software consultancy specialized in android technologies.
- In my product-design time, in the developing, I've encountered too many program manage troubles that suffer due to poor communication and code design, I know that help them to understand the system framework is very important. I am experienced in system and application layers, my goal is simple: help someone who wishes to better understand the **Android framework** in java、JNI and C/C++ libraries.
- Please also check my article and slides on this <http://blog.sina.com.cn/gswift>

Contact: Zhiyong.liu@aliyun.com



<http://weibo.com/gswift>