



# DevOpsDays

2017 DevOpsDays Beijing

时间：2017.03.18 地点：北京朝阳区万达索菲特大酒店

主办单位：



# 持续交付

聊聊那些概念，说说那些“坑”

讲讲我的历程，请当故事听

乔梁



2 0 0 1

2 0 0 4

2 0 0 7

2 0 1 0

2 0 1 1

2 0 1 3

2 0 1 6

# IT行业新词多

持续集成

敏捷软件开发

增长黑客

精益软件开发

精益创业

精益设计

看板方法

设计思维

持续部署

精益八折

持续交付

DevOps

# 持续交付

DevOps

敏捷开发

持续集成



# what is continuous delivery?

---

The ability to get changes—features, configuration changes, bug fixes, experiments—into production or into the hands of users safely and quickly in a sustainable way.





what

job title, keywords or company

where

Find Jobs

Tip: Enter your city or zip code in the "where" box to show results in your area.

## Devops Engineer jobs

Sort by: **relevance** - date

### Salary Estimate

\$90,000+ (7413)  
\$100,000+ (6254)  
\$110,000+ (4426)  
\$115,000+ (3355)  
\$125,000+ (1762)

### Job Type

Full-time (7616)  
Contract (875)  
Internship (68)  
Temporary (68)  
Part-time (37)  
Commission (11)

### Location

San Francisco, CA (731)  
New York, NY (425)

**New!** [Join Indeed Prime](#) - Get offers from great tech companies

Jobs 1 to 10 of 8,615

### DevOps Engineer

VitalSource Staffing - Chicago, IL

1-7+ years of DevOps / system administration and designing software solutions for high performant systems. The DevOps role is a critical part of their solutions...

1 day ago - [save job](#) - [more...](#)

### DevOps Engineer

Dynpro - [★★★★☆ 4 reviews](#) - Durham, NC +1 location

A DevOps Engineer will be responsible for identifying redundant processes and eliminating duplicate work, standardize processes, introducing automated processes...

7 days ago - [save job](#) - [more...](#)

### DevOps Engineer

Northrop Grumman - [★★★★☆ 2,248 reviews](#) - Woodlawn, MD

Minimum of 3 plus years as build and release engineer for software development. Are you interested in the opportunity to work for an industry-leading company...

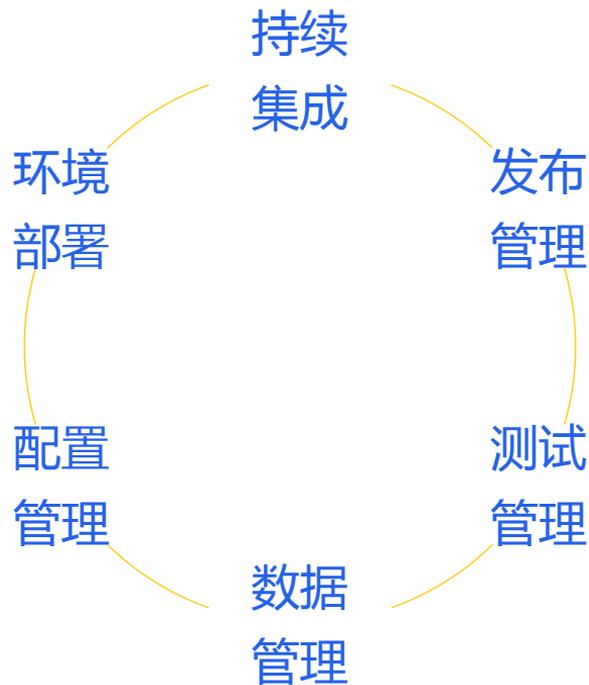
21 hours ago - [save job](#) - [more...](#)



← Right Way  
Wrong Way →



# 持续交付成熟度模型





湘  
油  
物  
步  
極  
價

綠  
其  
下  
著  
志  
烟

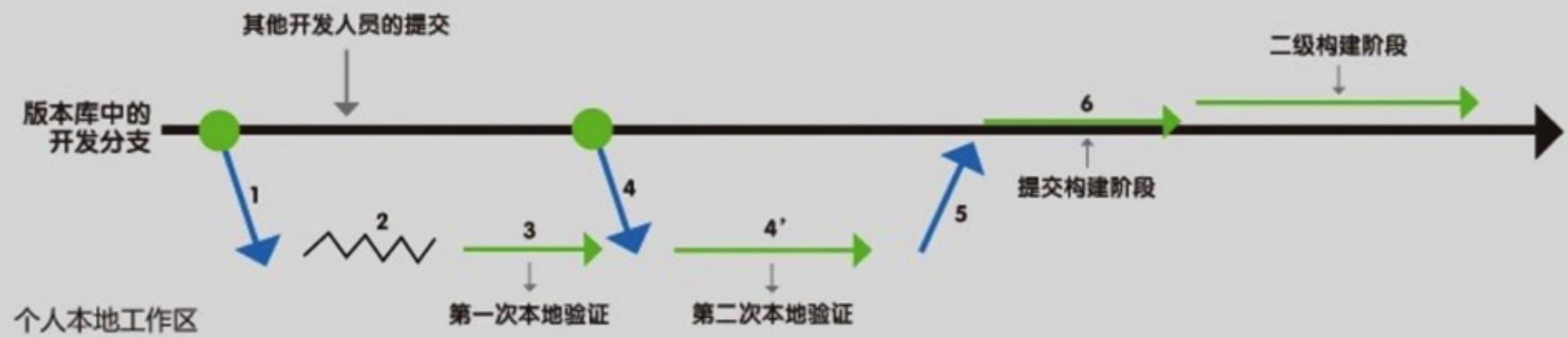


Themes 殘余 2/17

1. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.
2. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.
3. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.
4. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.

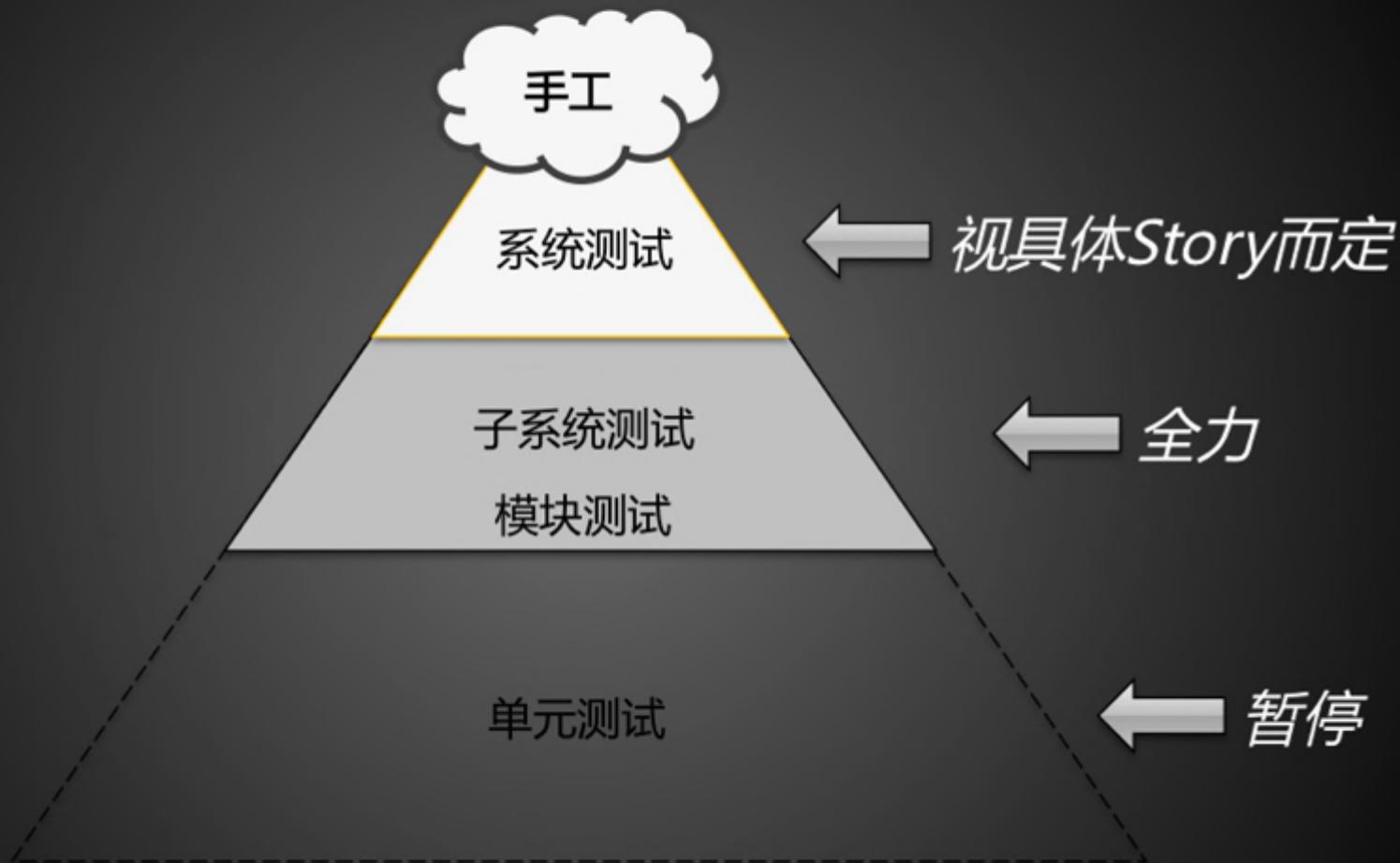
時間方 环境 负责人  
= 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.

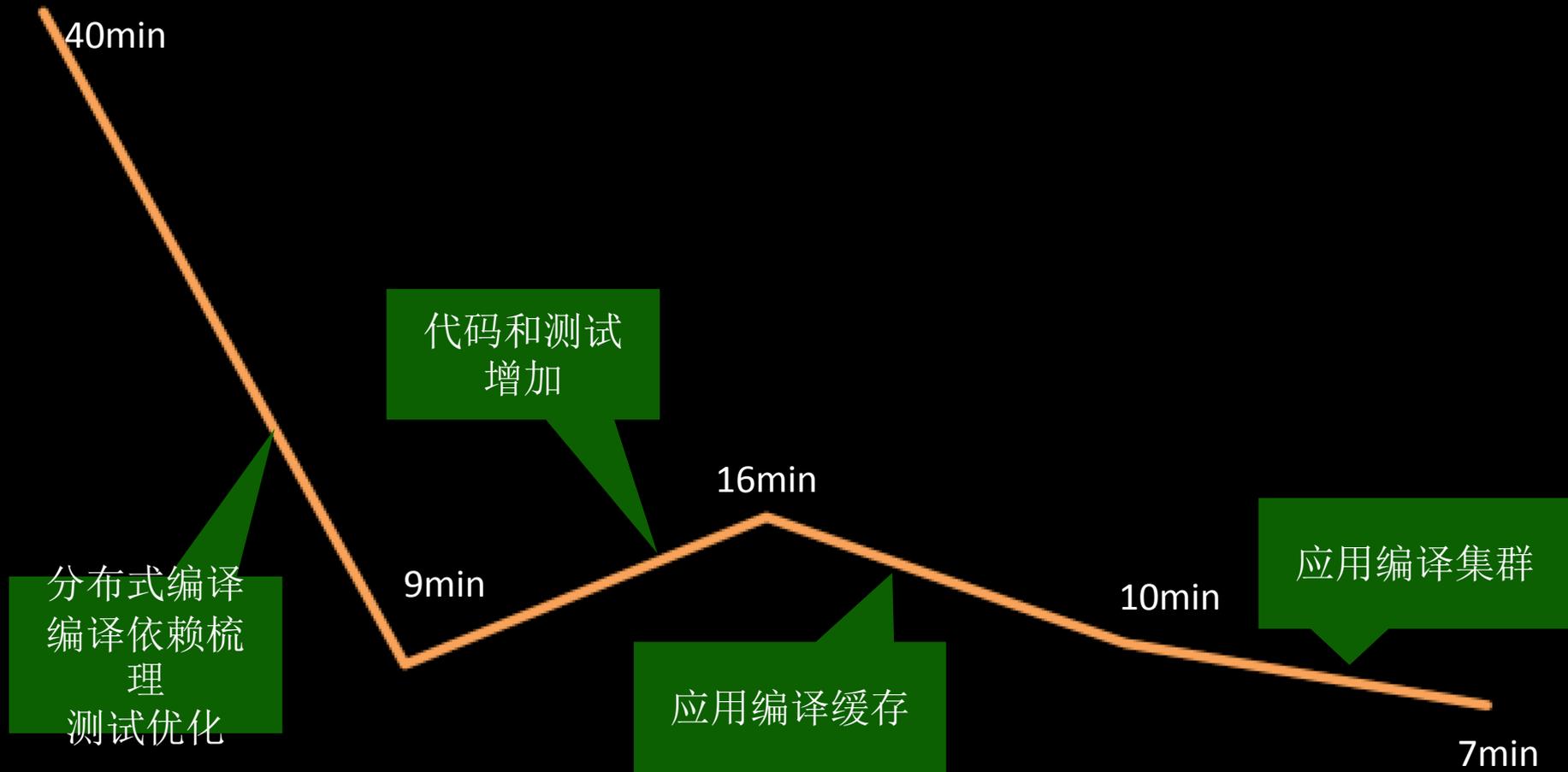
6. 15. 16. 17. 18. 19. 20. 21. 22. 23. 24. 25. 26. 27. 28. 29. 30. 31. 32. 33. 34. 35. 36. 37. 38. 39. 40. 41. 42. 43. 44. 45. 46. 47. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.



图例： ● 构建成功的线上版本 | 代码修改 | 运行测试 | 代码流向

# 测试金字塔





从三个月到两周





# How to do it in the right way ?



理想：2周1个beta版本，每月发布1个稳定版本  
现实：各种延迟，2，3个月没有稳定版可发布



需求变更多，很难管理

各个里程碑有疏漏的需求

开发完成后，Bug比较多

插入事情多，项目计划与交付风险不可预期

并行版本的开发分支过多，分支间同步工作量大

分支合并后验证成本较高

支线测试后期才暴露出来较多问题

多个需求，多个版本同时修改代码，开发、测试和上线部署冲突问题多

在临近产品交付的最后阶段，开发为了业务KPI临时加需求，导致测试压力大

测试作为靠后的工序，经常被压缩时间，给质量保证带来难度

测试对技术方案了解比较滞后，用例的设计存在一定盲目性

测试缺乏对系统实现的理解，对影响范围难以界定

问题发现的阶段太晚，内部测试不能发现，外发后解决，成本几十倍的增加、效率太低

软件构建打包时间太长

自动化测试成熟度不高

每次集成联调时间比较长

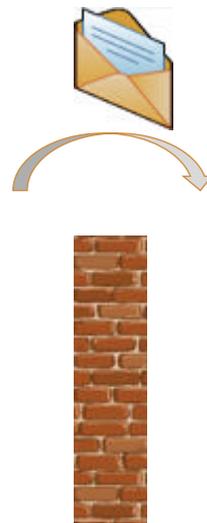
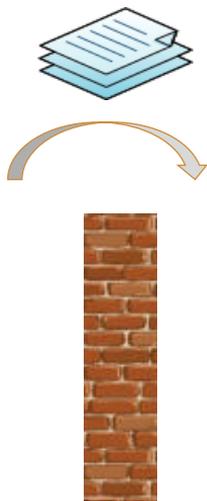
软件环境部署非常繁杂，耗时较长

有时候开发环境没有问题，部署到测试或生产环境就会有问题

# 1天2个beta版本，每周1个候选稳定版本

- 软件基本质量：crash下降了90%
- 品牌：腾讯精品第五 进步第三





能不能有点靠谱的设计

P0用例测试不通过，不许提测

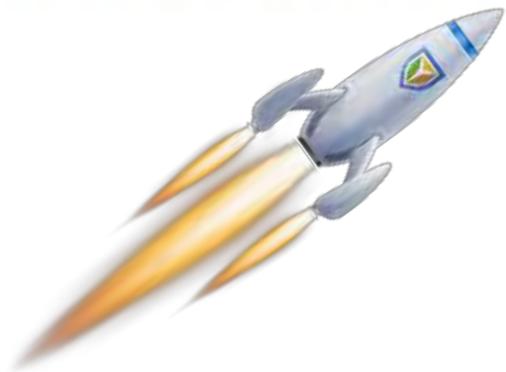
不出定稿不要给我们开发

测试阶段，不许再加需求了

需求变更要严格控制

这个问题是测试漏测。。。

# 1.组织架构解耦



这个设计这样做似乎更好？

我可以帮你多触发几种场景来验证

有版本可以验证不？

这个用户体验不够好，再改改

给我一个新版本体验一下

大家都好好验证一下。。。

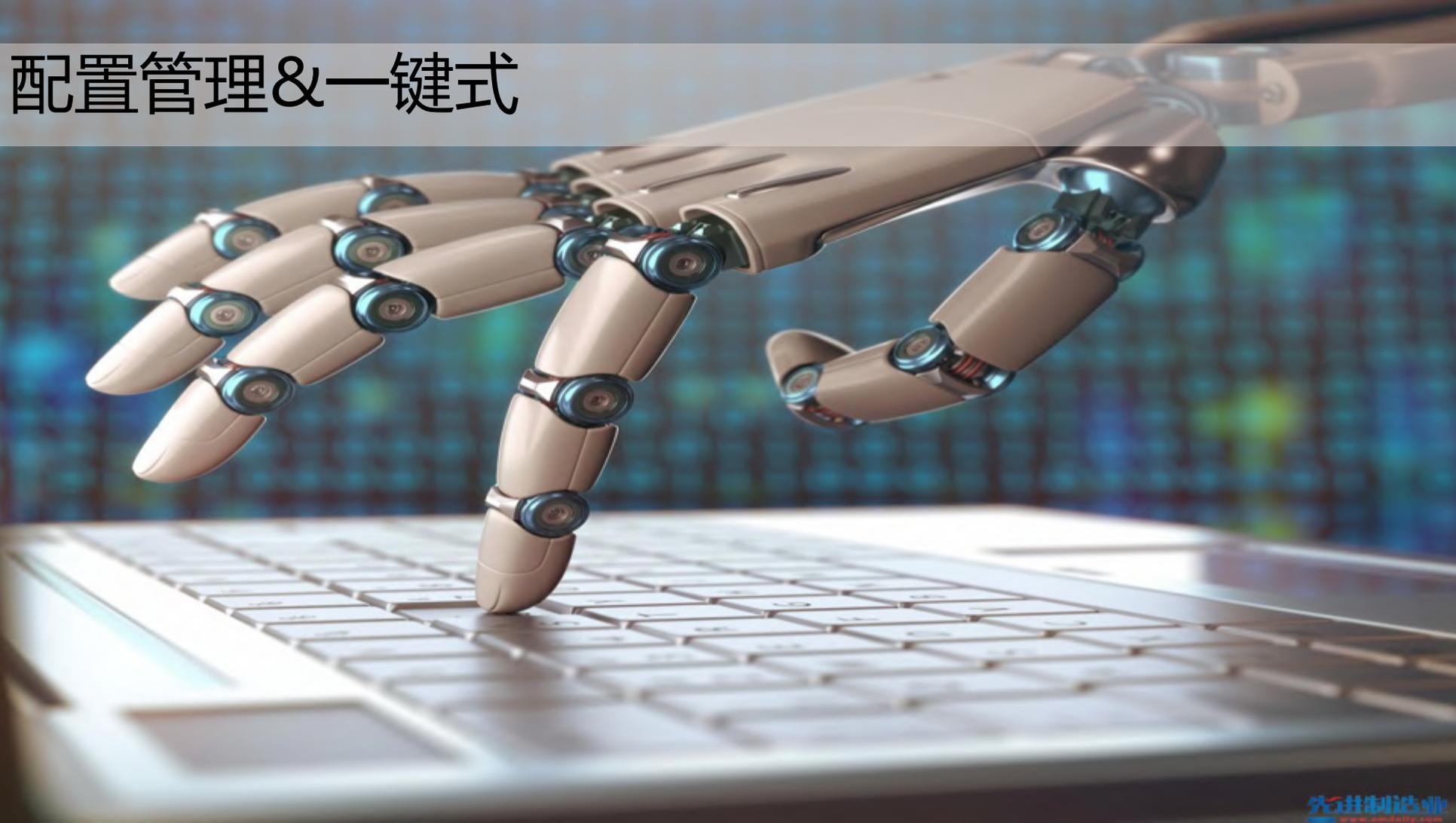


## 2. 软件架构解耦



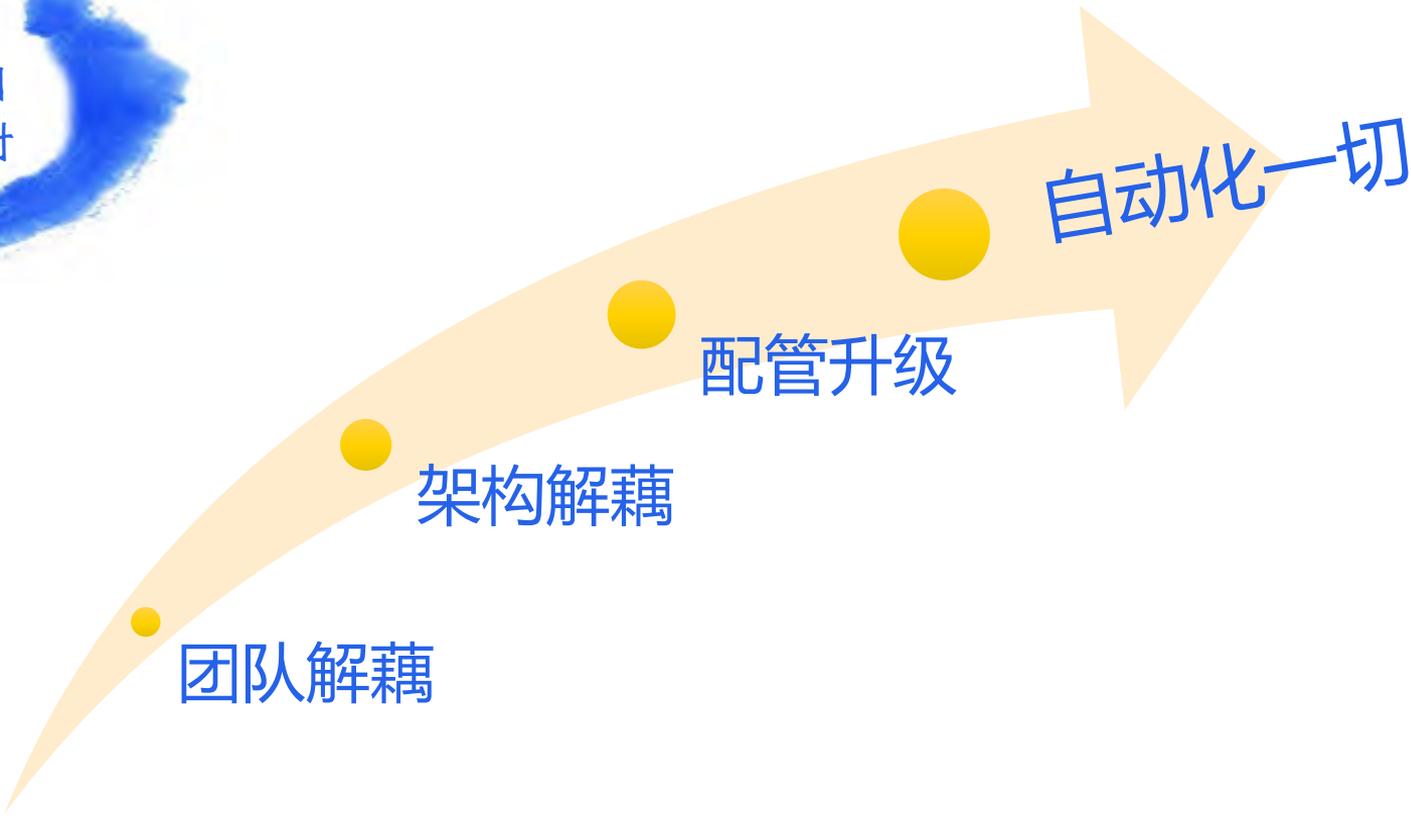


# 配置管理&一键式





从2个月  
到2小时



团队解藕

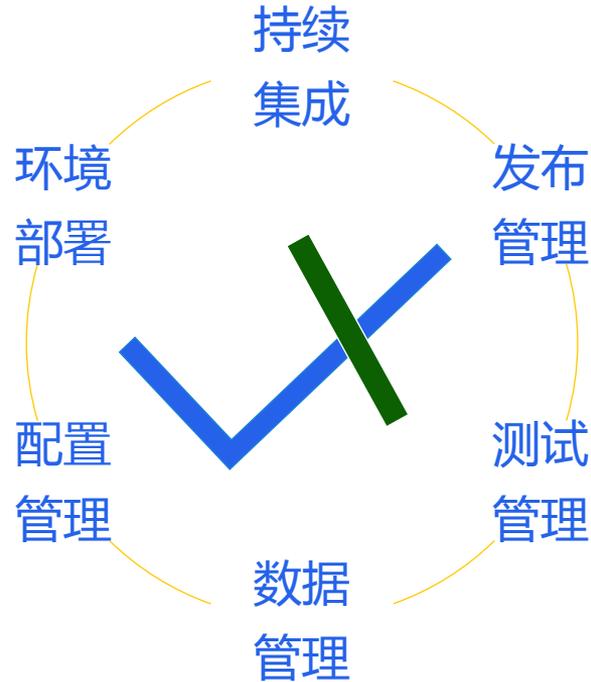
架构解藕

配管升级

自动化一切



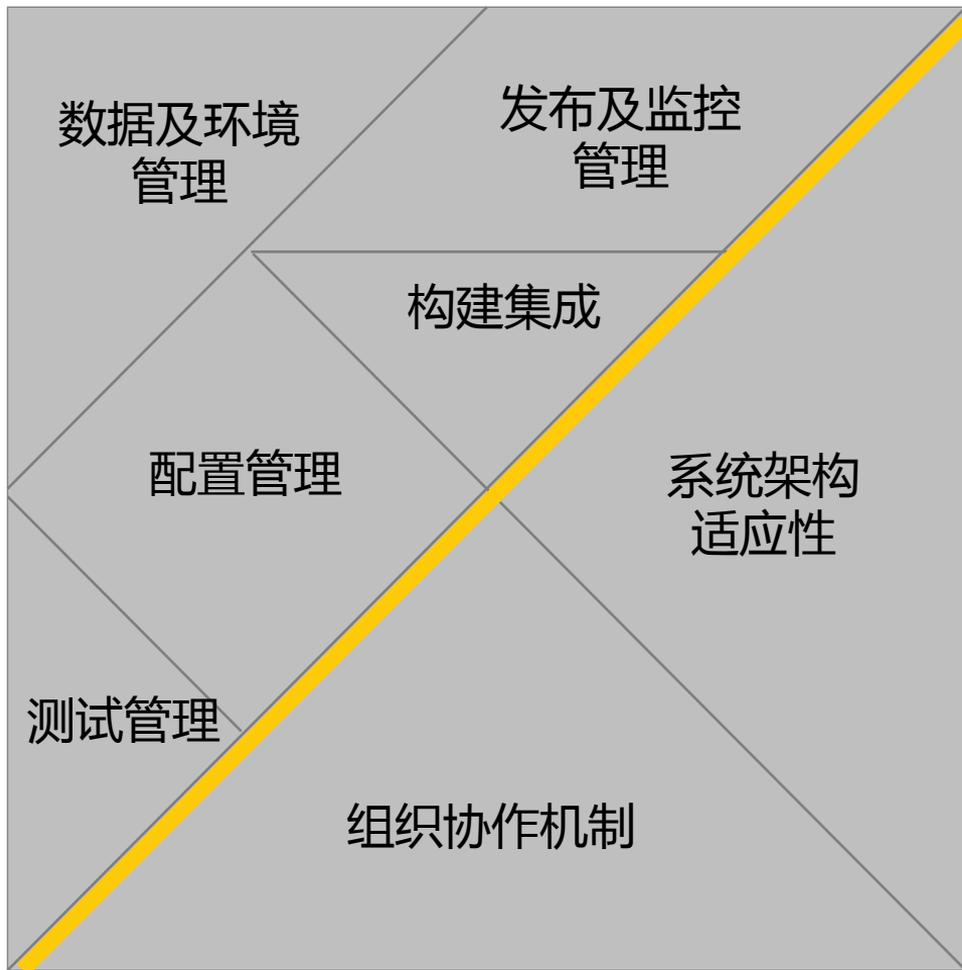
# 持续交付成熟度模型



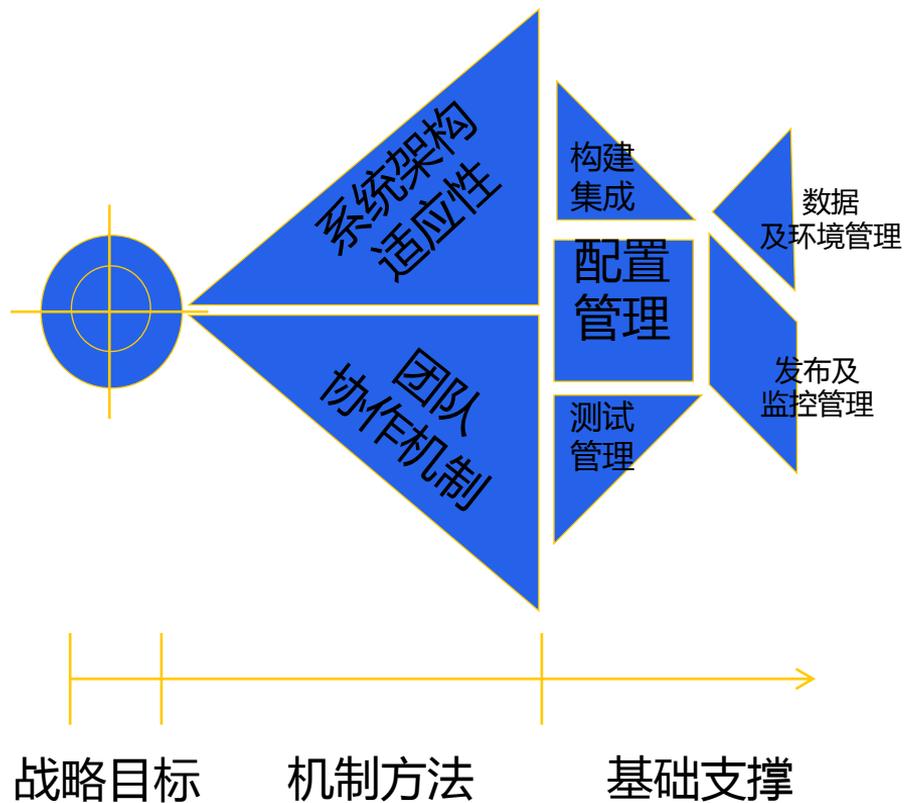
# 持续交付七巧板



# 持续交付七巧板



# 《持续交付》三步走



# 别提概念，只解决问题！

乔梁

持续交付

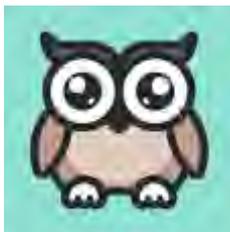
组织转型

移动互联网



畅游  
changyou

MEIZU



墨迹天气

卷皮  
JUANPI.COM

别提概念，只解决问题！



持续交付

敏捷



Q & A





# 高效运维社区

GreatOPS Community

GOPS 4月深圳 / 7月北京 / 11月上海

EXIN DevOps Master 认证研修

DevOpsDays 3月北京 / 8月上海

DevOps 企业内训 / 咨询服务

DevOps China 全国巡回技术沙龙

其他量身定制服务项目



商务经理：刘静女士

电话 / 微信：13021082989

邮箱：liujing@greatops.com



# Thanks

荣誉出品

高效运维社区

国际最佳实践管理联盟