



THINK OPEN

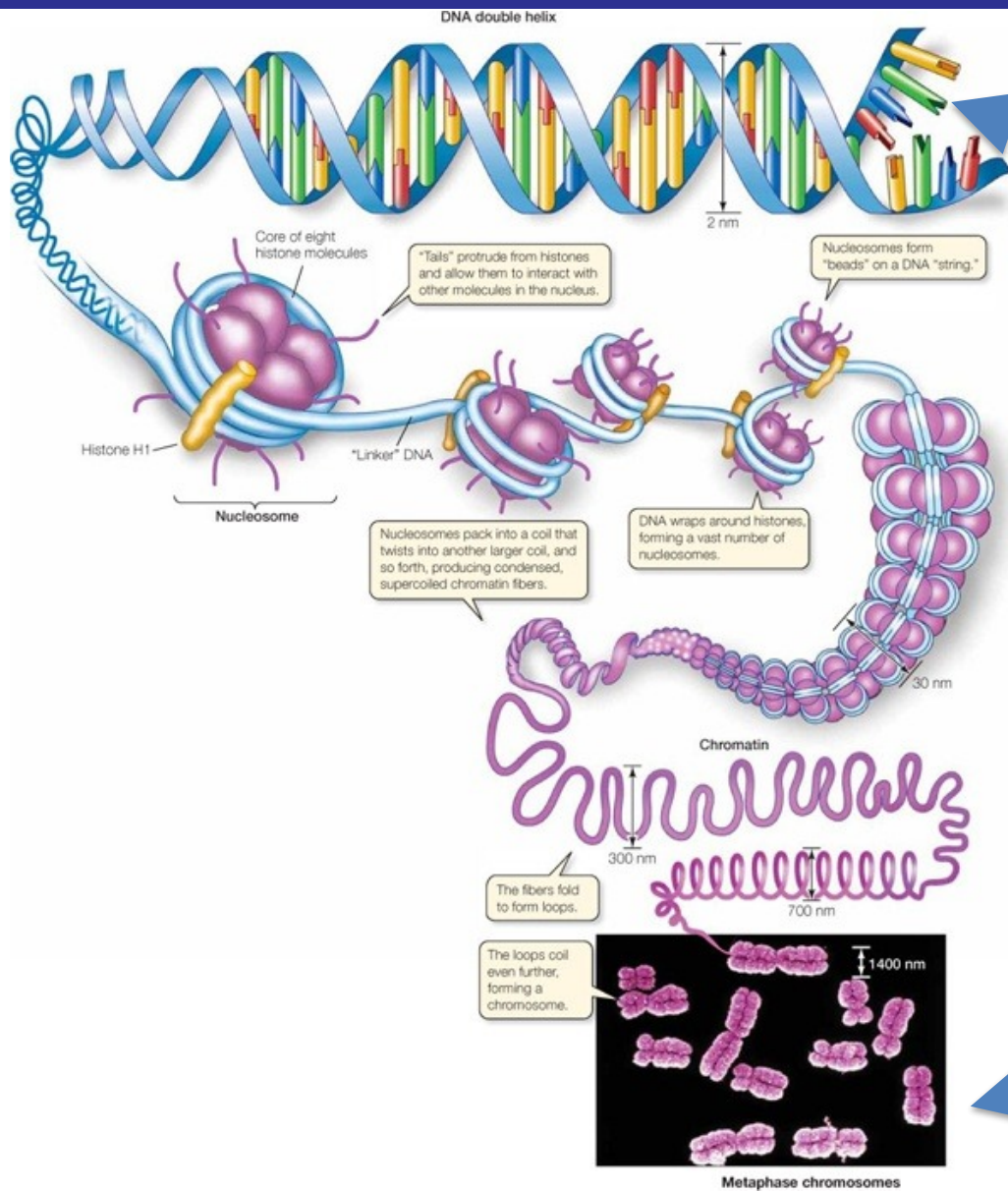
开放性思维

# *Best Practice Of Adopting Kubernetes On Genome Sequencing*

唐盛军 / 华为云SA  
[tsjsdbd@huawei.com](mailto:tsjsdbd@huawei.com)

- 为什么测序需要容器和K8S
  - **基因测序介绍**
  - 容器的优势
  - 任务(资源)调度
- K8S与测序的结合
  - Job执行单位
  - PVC共享存储
- 测序流程控制
  - Helm的不足
  - 云上编排服务

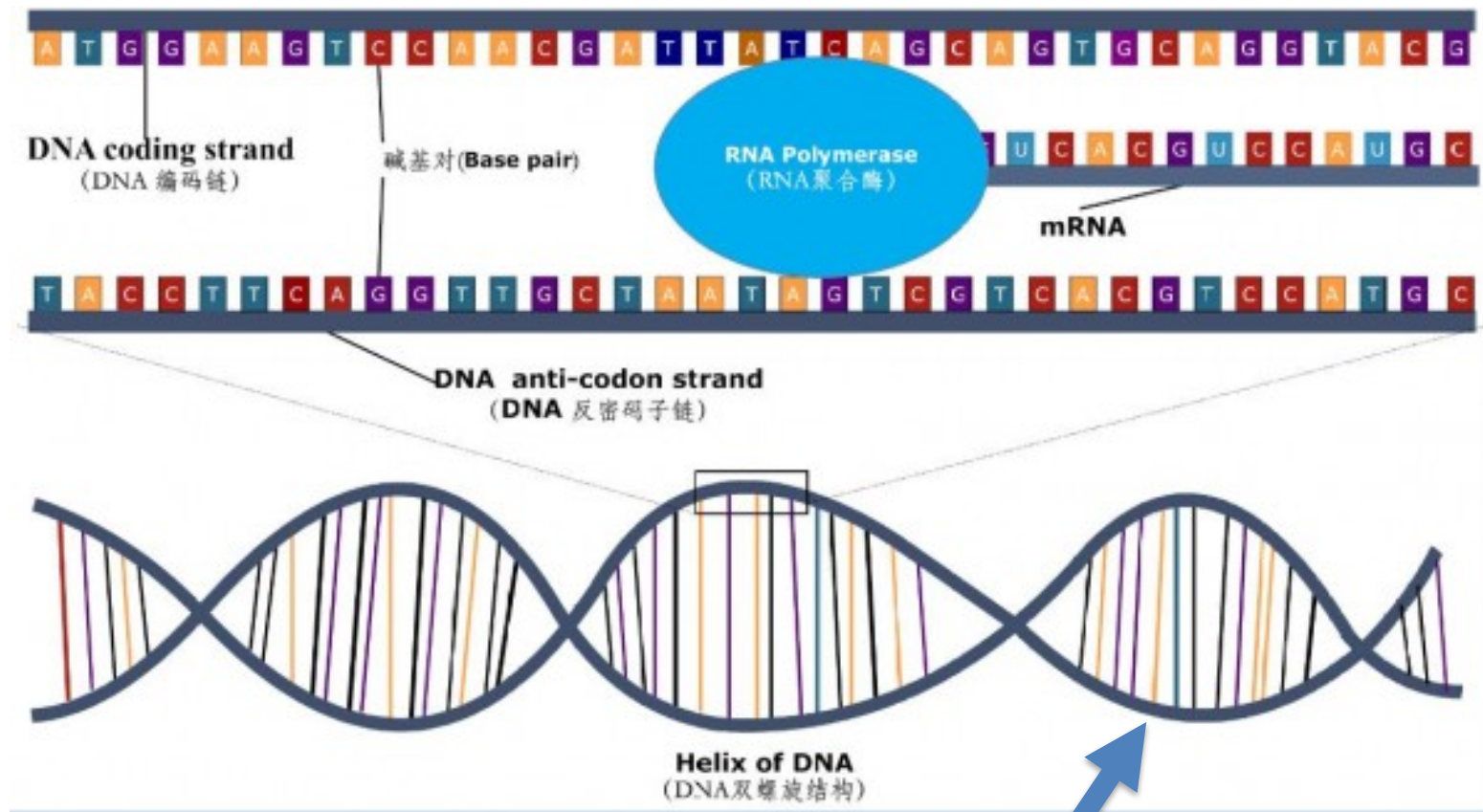
# 基因测序介绍 – 基因



碱基对：  
AT / GC

人类：  
23对（46个）染色体

# 基因测序介绍 – 测序

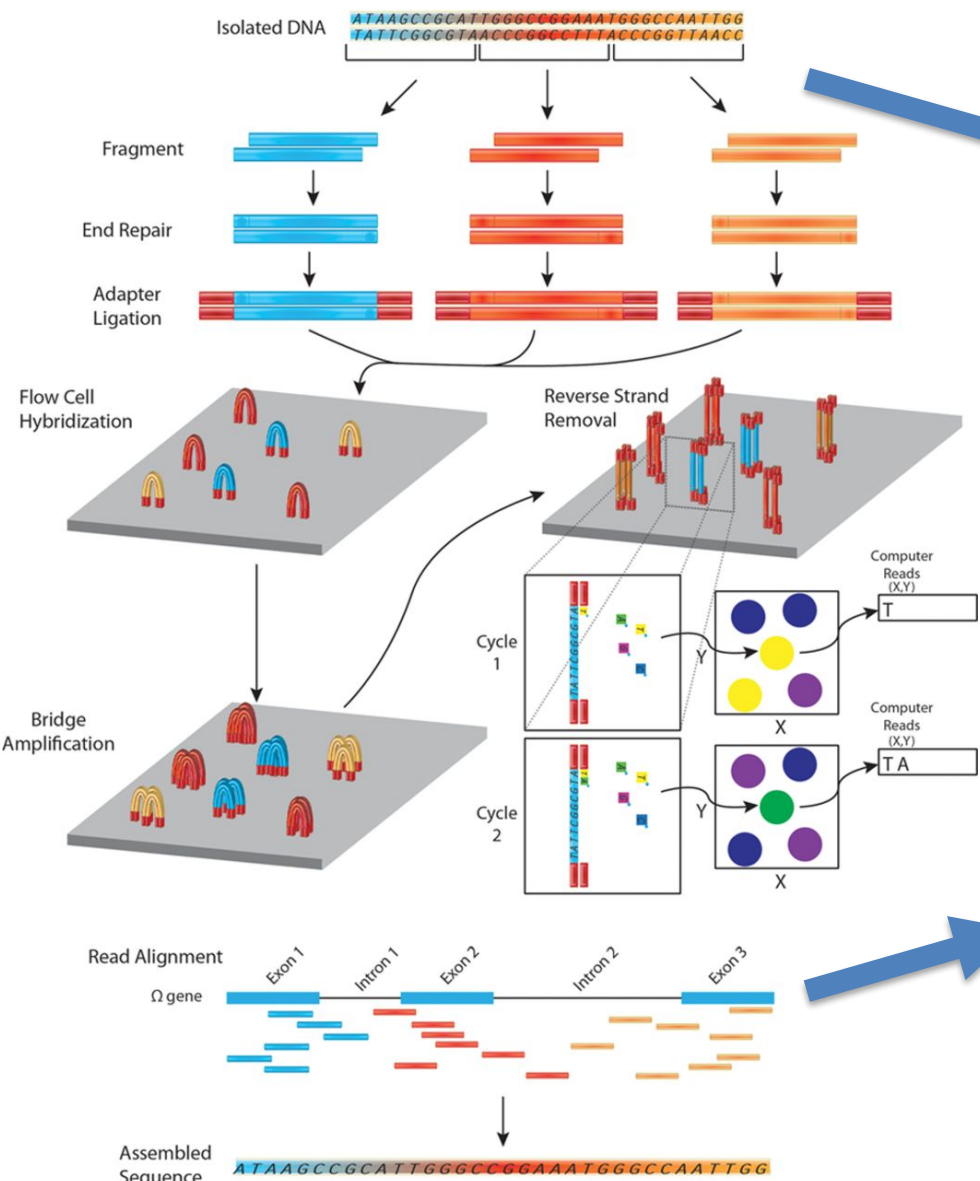


测序目标: 确定ATGC 的顺序  
长度: 30 亿对 (人类)





# 基因测序介绍 – 二代测序(NGS)



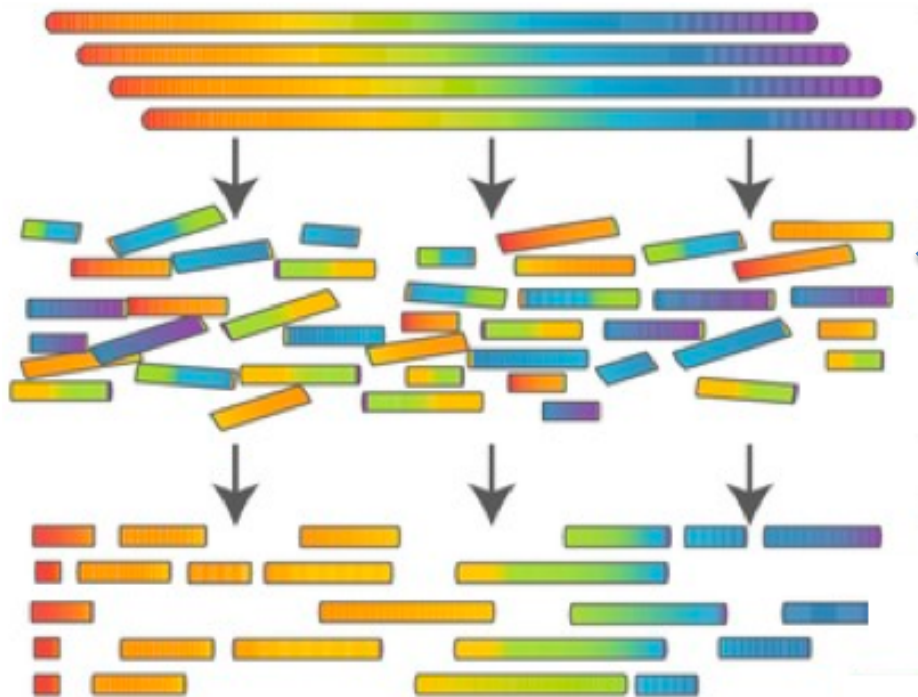
特点:

1. 打断成小片段，分别测序
2. 最后根据参考组，还原结果

一次测序量很大，成本优势明显  
目前成为主流测序方法

人类基因的顺序 =》 参考组

# 二代测序 → 拼图



拼图特点：

1. 重复的片很多
2. 缺少部分片
3. 部分片有形变
4. 部分片有干扰
5. 片的数量巨大

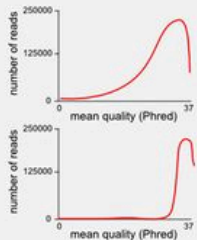
ATGTTCCGATTAGGAAACCTATCTGTAAGTTCATTTCAGTAAAAGGAGGAAA

拼图结果参考

# 拼图算法 → 软件处理

①

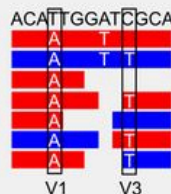
quality control



FASTQ  
↓ *FastQC*  
Quality plot  
↓ *sickle*  
FASTQ (trimmed)

⑤

filter variants

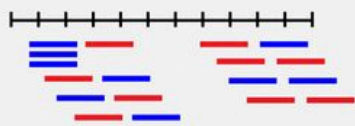


variant metrics  
V1: DP=7, AF=1.0  
V3: DP=6, AF=0.5

VCF1  
↓ *GATK suite*  
VCF1 (filtered)

②

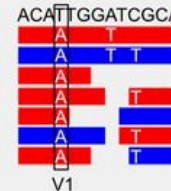
align to genome



FASTQ (trimmed)  
↓ *BWA*  
SAM  
↓ *Samtools*  
BAM

⑥

subtract variants

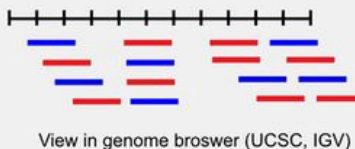


variant metrics  
V1: DP=7, AF=1.0

VCF1 (filtered) - VCF2  
↓ *GATK suite*  
VCF1 (subtracted)

③

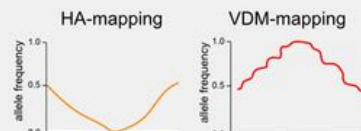
realign/remove dups



BAM  
↓ *GATK suite*  
BAM (realigned)  
↓ *Picard*  
BAM (dups removed)

⑦

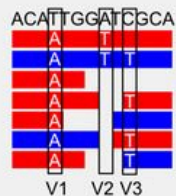
plot variants



VCF1 (subtracted)  
↓ *Cloudmap*  
Mapping plots

④

variant calling

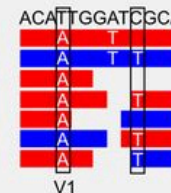


variant metrics  
V1: DP=7, AF=1.0  
V2: DP=2, AF=1.0  
V3: DP=6, AF=0.5

BAM (dups removed)  
↓ *GATK suite*  
VCF1

⑧

annotate variants

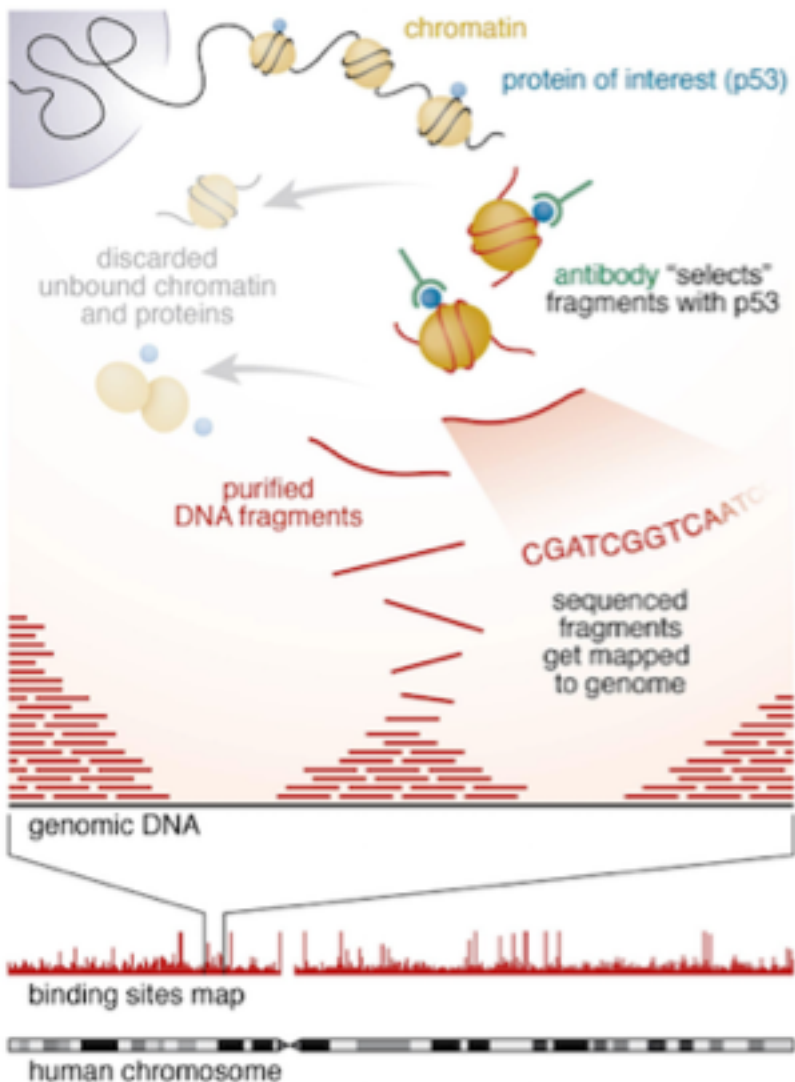


variant annotation  
V1: STOP

VCF1 (subtracted)  
↓ *SnpEff*  
Tabular (annotated)



# 二代测序总结



测序仪原理 =》 拼图算法

## 1. 数据量巨大（高通量数据源）：

- \* 以HiSeq X10测序仪为例,产生1600Gb /轮
- \* 一个人30x全基因组样本数据FASTQ文件大约是200GB
- \* 一个人30x全基因组中间数据sam文件大小超过600GB

## 2. 需要大量计算

- \* Mapping, Filter, Deduplication, Alignment, Index, Compress,
- \* Recalibration, Depth-coverage, Cleanup, Calling-variants...

## 3. 性能+成本要求高

- \* 巨大的数据需处理
- \* 时间需求 & 成本

各种软件各种算

- 为什么测序需要容器和K8S
  - 基因测序介绍
  - **容器的优势**
  - 任务(资源)调度
- K8S与测序的结合
  - Job执行单位
  - PVC共享存储
- 测序流程控制
  - Helm的不足
  - 云上编排服务

# 传统测序的不足

- 1. 需要大量软件 (计算)

步骤多: Mapping, Filter, Deduplication, Alignment, Index, Compress, Recalibration, Depth-coverage, Cleanup, Calling-variants...

- 2. 能控制处理流程

\* 一个步骤2个小时, 需要能自动化控制执行顺序

- 3. 性能+成本

\* 一个人30x全基因组样本数据文件大约是200GB

\* 整个技术时间越快越好 + 费用越低越好

- 1. 提前安装

软件安装管理困难, 版本冲突, 环境修复等。如 GATK 软件需要 jdk 1.8, 而其他软件需要 jdk 1.7/1.6

- 2. PBS/SGE

安装管理复杂, 流控控制额外开发Python程序

- 3. HPC/Batch

多开机器, 跑得快。多人共用一套环境

# 容器的优势

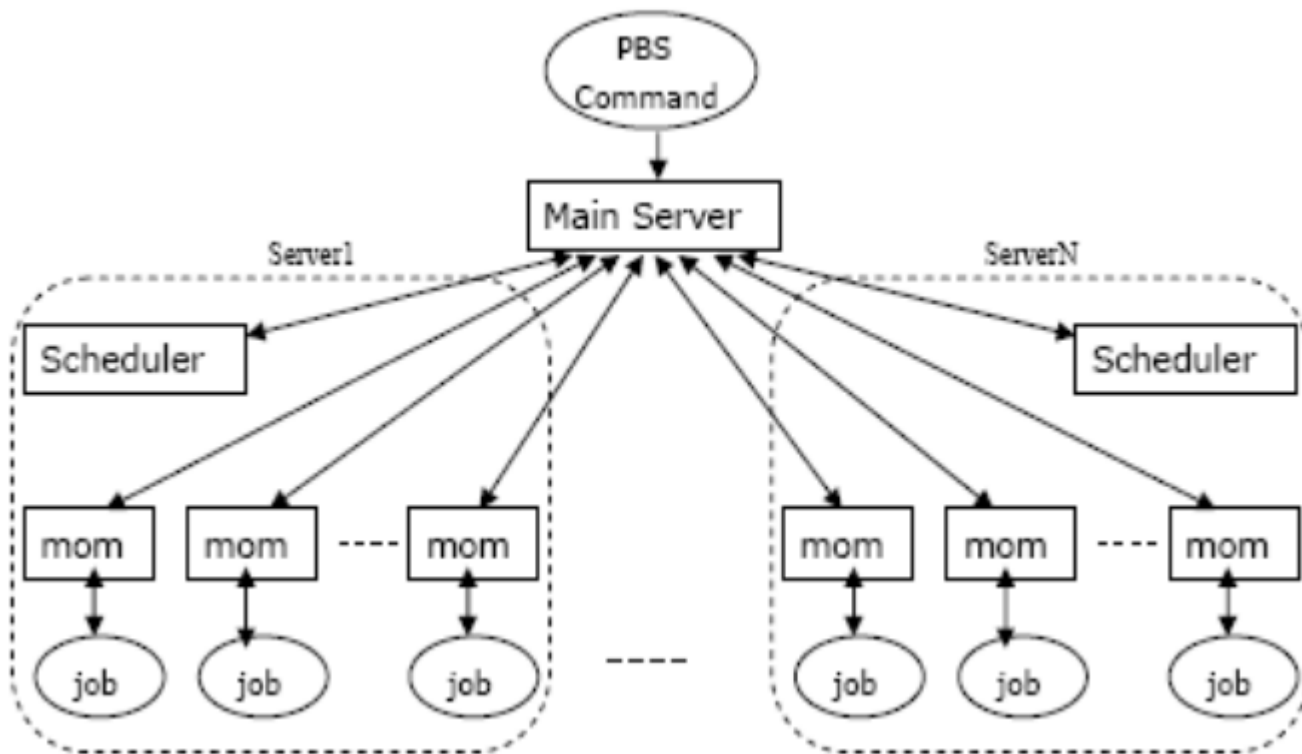
不足	原因	优势
生信软件繁多， 环境准备成本高	生信领域数据处理流程复杂，使得单步骤就存在多种软件可选，单软件也有 <b>多版本</b> 可选。而在传统虚拟机中运行软件，不仅安装多种软件痛苦，还需要考虑软件间 <b>版本兼容性</b> ，同一软件多实例间竞争影响。生信 <b>软件安装</b> 对于IT能力不是主战场的生信人来说，是一大痛苦。	Docker容器技术使得不同软件间运行无感知。同一个软件任意多实例运行也不感知。且Docker镜像以软件为单位制作，独立制作&更新非常便捷，且不影响其他软件。可以极大提高生信环境安装及环境恢复的效率。
调试流程不方便	由于前期调试流程中的错误，导致软件配置改变或者 <b>环境异常</b> ，就得重头搭建&恢复环境，单独 <b>软件的更新</b> 也非常不方便。软件便捷的分发&更新成本很高，开发环境的快速复制及 <b>重建</b> ，成为生信人迫切渴望。	Docker镜像便捷分发，Docker容器镜像的一次构建， <b>随处可运行</b> 机制，使得本地验证调试非常简便。环境的切换对Docker程序无影响，保证 <b>运行环境的一致性</b> ，以及数据处理 <b>结果的可重复性</b> 。
资源利用率低	单样本数据处理一般在单机上完成，所以目前普遍使用高规格机器来执行 <b>pipeline</b> 流程。而且多机并发能力不足，如单独开发任务调度框架，门槛成本和收益不明显。同时单机上 <b>多任务并发</b> 能力也受限。无法很好的满载利用计算资源。	单虚拟机上可以 <b>同时运行</b> 多个Docker容器。使得多个任务同时在一个VM上执行。同时利用Kubernetes资源调度能力，使得多Job按资源需求平均地在多VM上混合执行，最大化榨取资源。（充分利用满计算节点的计算能力）

- 为什么测序需要容器和K8S
  - 基因测序介绍
  - 容器的优势
  - **任务(资源)调度**
- K8S与测序的结合
  - Job执行单位
  - PVC共享存储
- 测序流程控制
  - Helm的不足
  - 云上编排服务

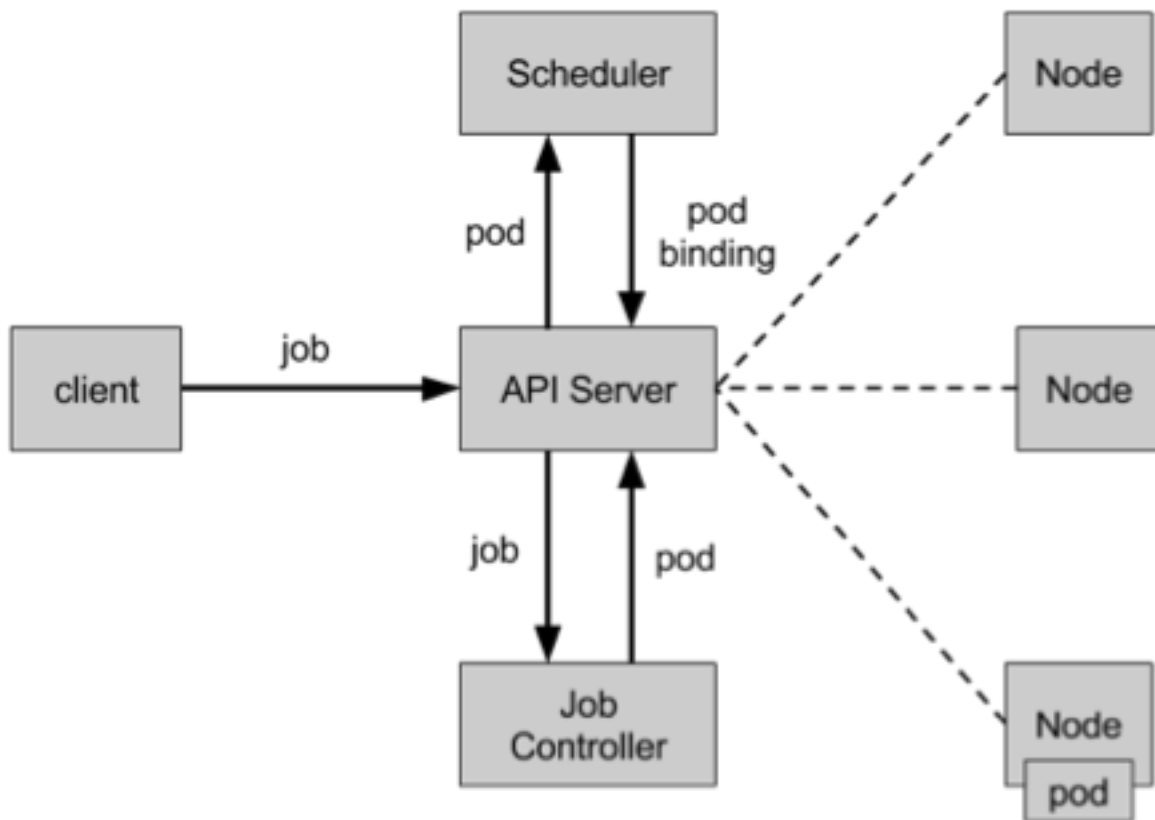


# 传统任务(资源)调度

使用HPC集群，通过PBS软件调度任务(资源)



# K8S任务(资源)调度



共同点：  
把任务调度到节点  
上去执行。

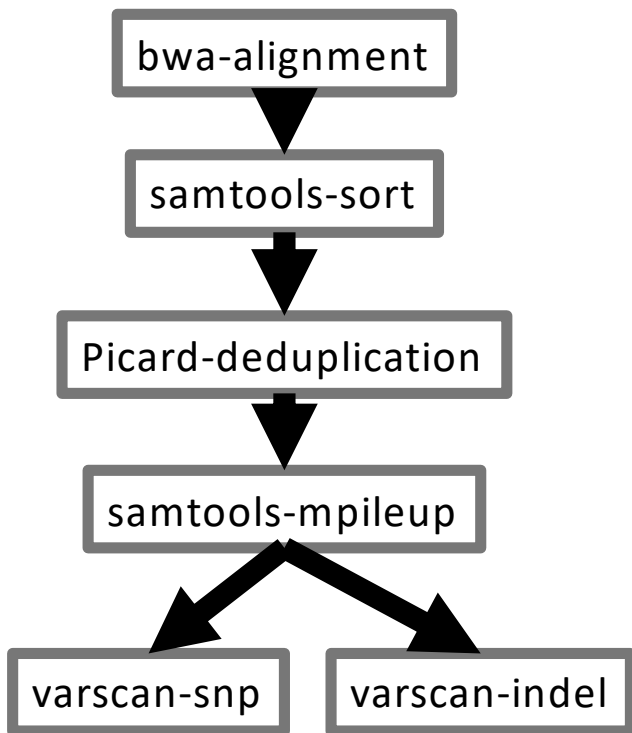


可以用K8S的Job跑  
基因测序任务

- 为什么测序需要容器和K8S
  - 基因测序介绍
  - 容器的优势
  - 任务(资源)调度
- **K8S与测序的结合**
  - Job执行单位
  - PVC共享存储
- 测序流程控制
  - Helm的不足
  - 云上编排服务

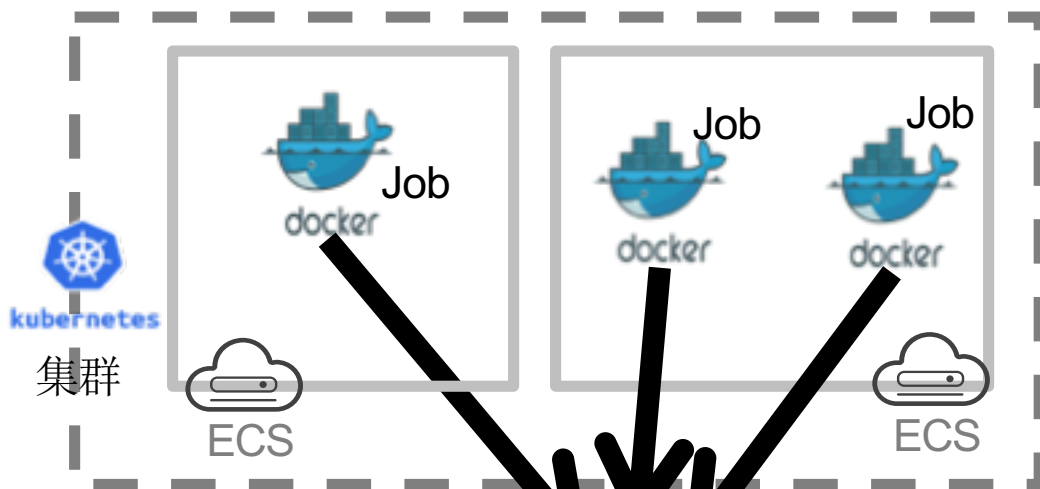
# 使用K8S执行测序流程

## 软件执行流程



基因样本片段（150M）测试结果：20分钟

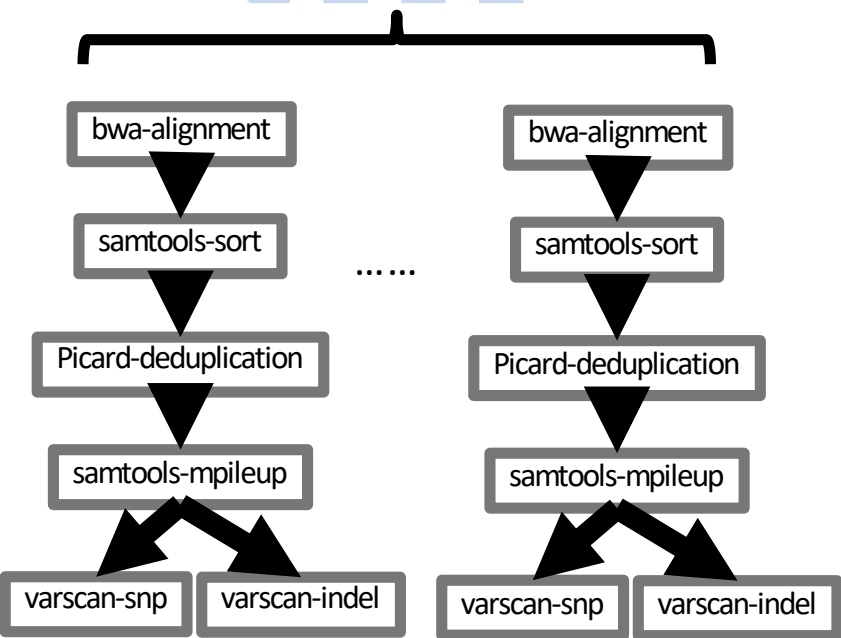
## 示意流程



样本放共享存储里：  
sample  
|--R1.fq.gz  
|--R2.fq.gz  
PVC  
Volume

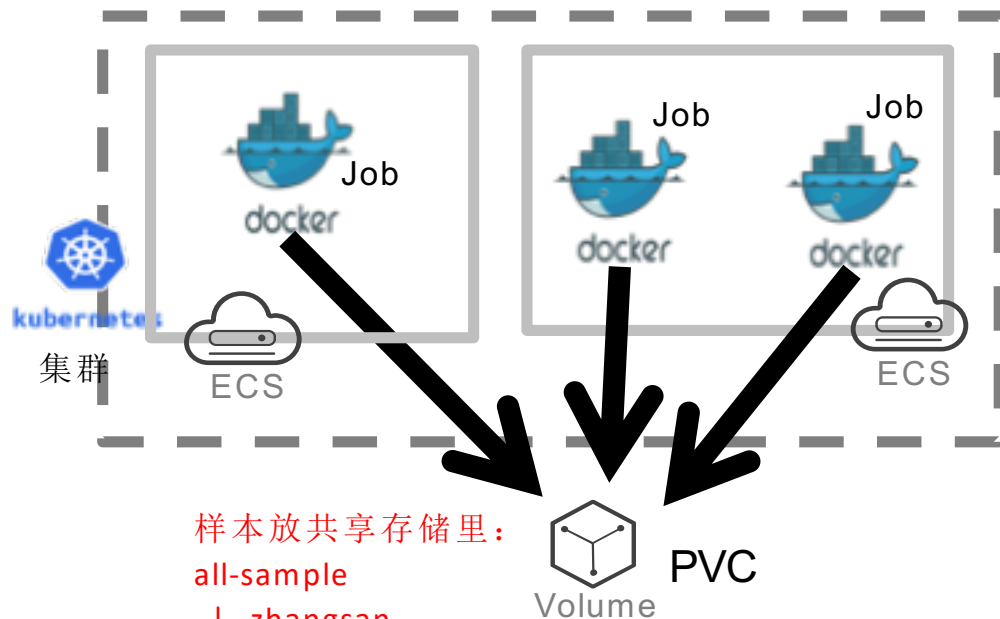
# 使用K8S执行复杂测序流程

多样本同时跑



一个Job执行完，然后执行下一个Job，谁来控制呢？

实现示意：更多的Job在跑



样本放共享存储里：

```

all-sample
|--zhangsan
|   |-- R1.fq.qz
|   |-- R2.fq.qz
|--lisi
|   |-- R1.fq.qz
|   |-- R2.fq.qz
|--wangwu
   |-- R1.fq.qz
   |-- R2.fq.qz
  
```



- 为什么测序需要容器和K8S
  - 基因测序介绍
  - 容器的优势
  - 任务(资源)调度
- K8S与测序的结合
  - Job执行单位
  - PVC共享存储
- **测序流程控制**
  - Helm的不足
  - 云上编排服务

# Helm中多Job顺序控制的不足

- Helm通过Hook来执行Job

- 预安装pre-install: : 在模板渲染后执行，但在Kubernetes中创建任何资源之前执行。
- 安装后post-install : 在所有资源加载到Kubernetes后执行
- 预删除pre-delete : 在从Kubernetes删除任何资源之前执行删除请求。
- 删除后post-delete : 删除所有release的资源后执行删除请求。

- 可以定义Dependency，但是不做流程控制的

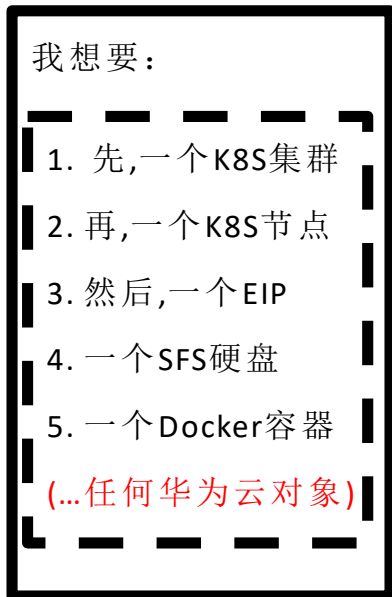
```

dependencies:
  - name: apache
    version: 1.2.3
    repository: http://example.com/charts
  - name: mysql
    version: 3.2.1
    repository: http://another.example.com/charts
  
```

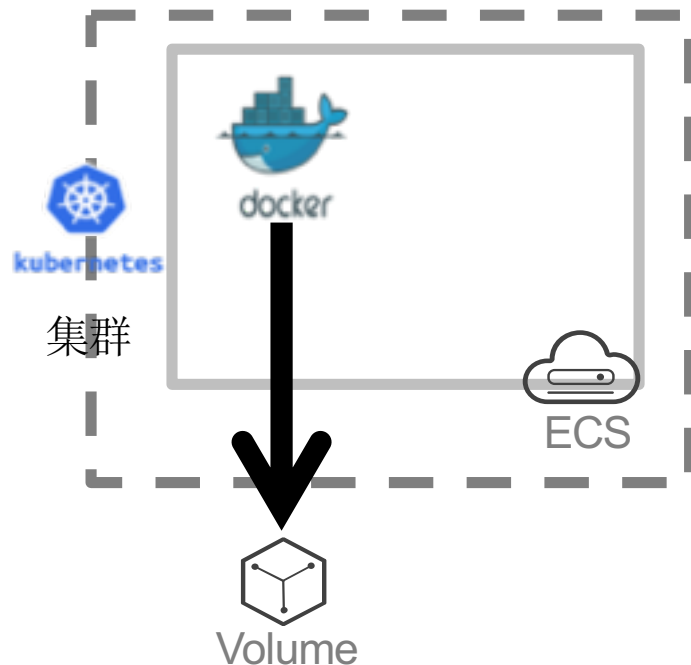
Once you have a dependencies file, you can run `helm dependency update` and it will use your dependency file to download all the specified charts into your `charts/` directory for you.

# 云上的流程控制框架

## 描述控制流程(文件)



## 执行结果



几乎每个云平台都有的服务：

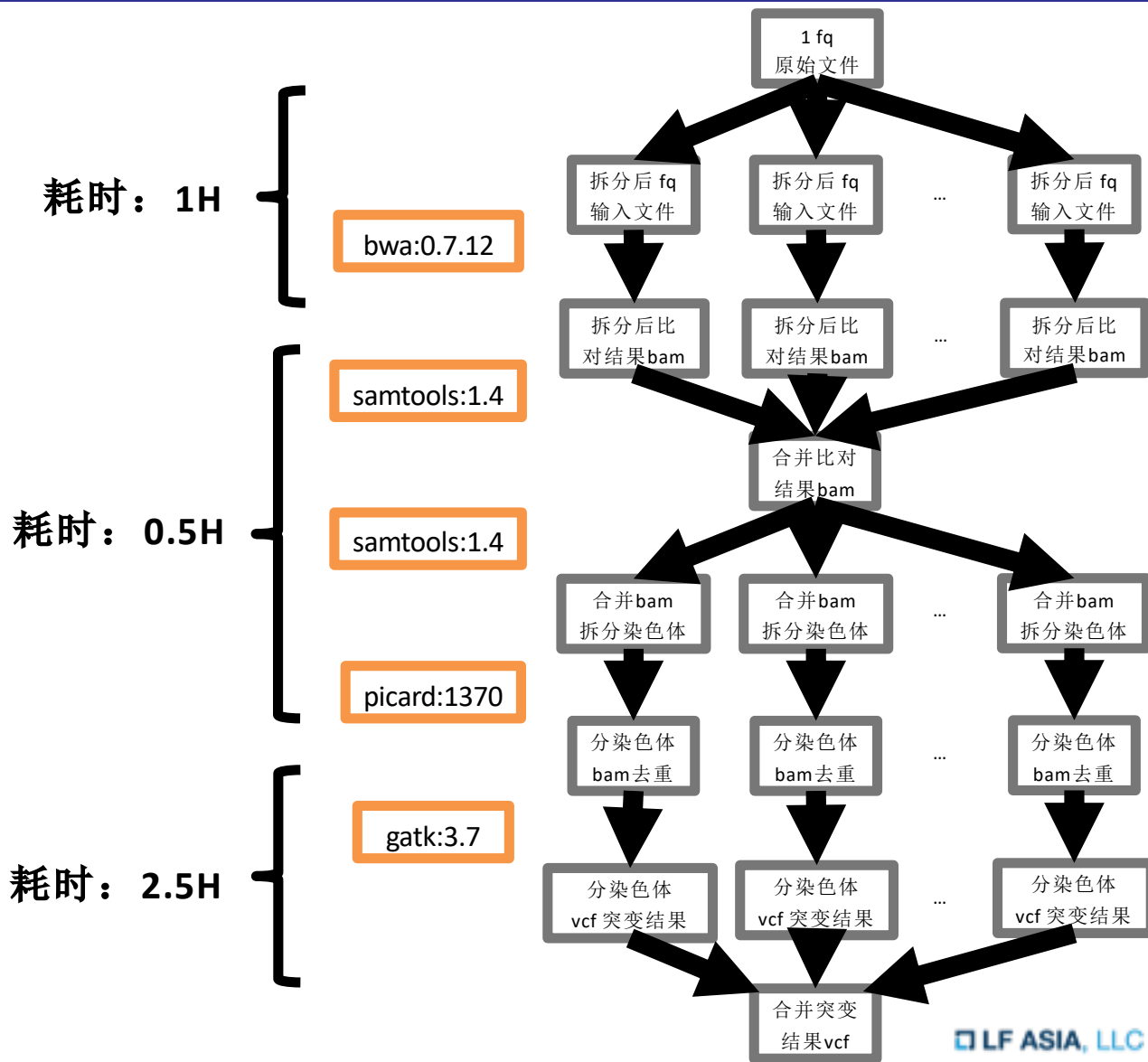
AWS的Cloudformation, 阿里的ROS, OpenStack的Heat, 微软的RM, 谷歌的CDM, 青云的RO

详见：[《全球公有云编排服务大比拼》](#)

# 使用K8S执行并发测序

大豆基因 (6G) 测试结果:

1. 总耗时: 4小时左右
2. (环境+流程) 可重复执行



# 总结：K8S & 基因测序

- 生信软件安装
  - 手动安装 => Docker容器
- 任务(资源)调度：
  - PBS => Kubernetes-Job
- 流程管理：
  - Python编程 => AOS自动化





LINUXCON

containercon



CLOUDOPEN

CHINA 中国

THINK OPEN

开放性思维