云际环境下虚拟专云的资源管理

北京大学
曹东刚

虚拟专云：面向领域的云上云

痛点

技术途径

应用案例

云平台趋向 标准化和同质化

大型通用云

定制困难 → 管理粗放 → 使用复杂 → 平台锁定

用户对云服务的要求 多样化和个性化

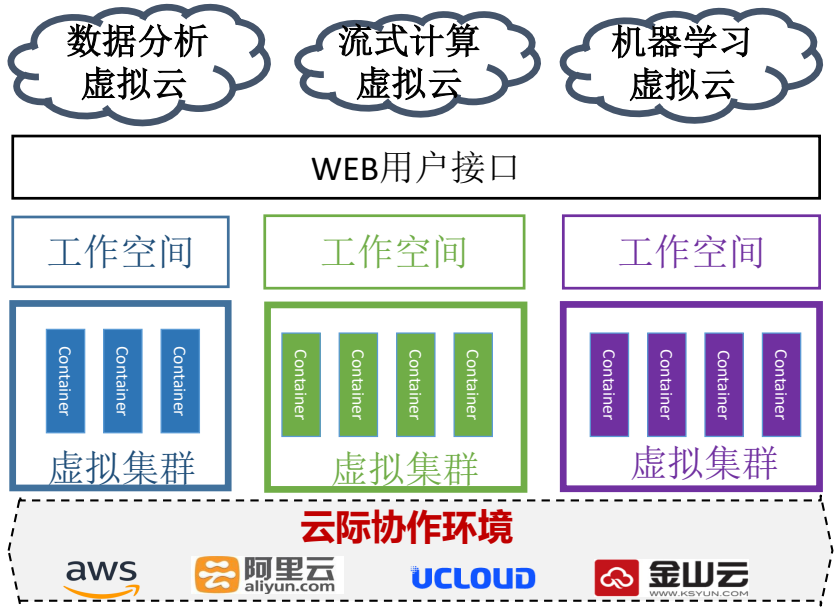
虚拟专云

虚拟专云：面向特定领域提供云化解决方案

软件定义的开源虚拟专云管理系统

个性定制；精细管理

友好易用；跨云部署



虚拟专云管理系统Docklet： https://github.com/unias/docklet

科学家云上工作室虚拟专云

- 大数据系统软件国家工程实验室发布，方便科学家在云上开展科研实验

高性能计算虚拟专云

- 中俄合作，接入无锡神威超算资源，为用户提供方便易用的超云工作空间

轻量级数据处理虚拟专云

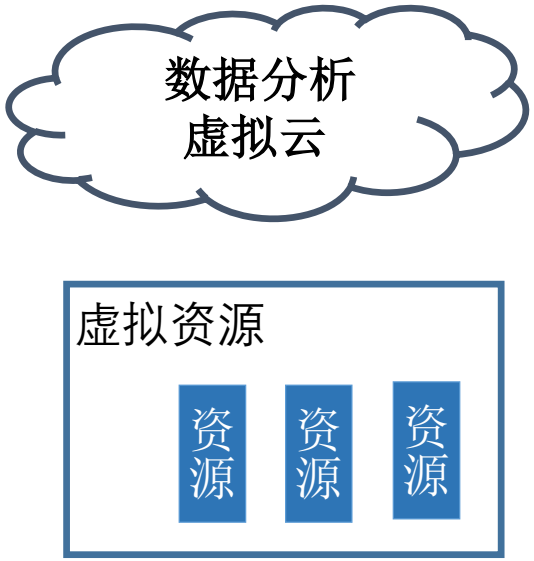
- 北京大学部署，提供数据分析与可视化、虚拟实验环境等服务

用户视角的虚拟专云资源管理

用户视角:

☺ 云提供了弹性计算的能力

☹ 云上的资源都是要收费的!



云际协作环境

aws 阿里云 aliyun.com UCLOUD 金山云 WWW.KSYUN.COM

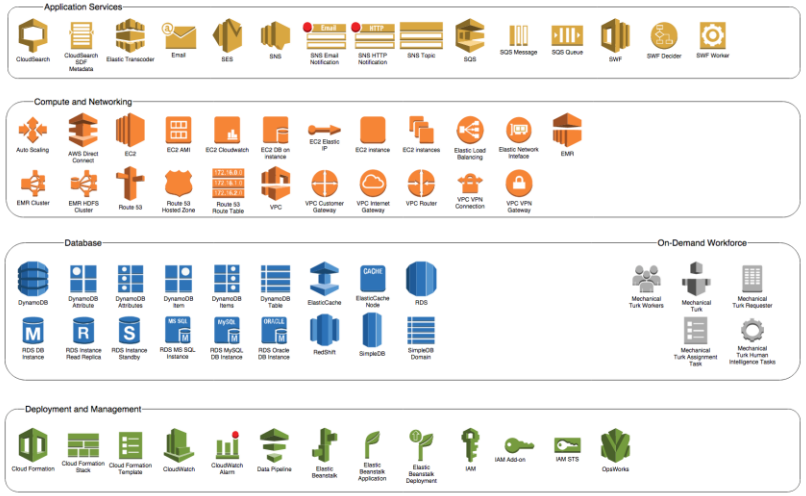
虚拟专云中资源管理的核心问题

UCLOUD

阿里云 aliyun.com

腾讯云

不同云服务商产品在在性能、质量、价格等方面均存在差异，如何选？



单个云服务商提供的产品越来越细化，不恰当使用造成大量开支，如何用？

如何结合领域应用需求，从实体云资源中选取并定制虚拟专云服务，来充分发挥云的弹性计算和按量付费的优势？

云厂商提供：基于分类的资源选取

- 公有云厂商一般会对计算资源进行分类，用户可以按照自己的需求选择相应的资源
- 基于分类的资源选取方式有一定指导意义，但存在如下问题
  - 需依赖用户专业知识进行选择
  - 缺乏不同云厂商之间的横向对比

实例类型	细分系列
通用型	A1、T3、T2、M5、M5a、M4
计算优化型	C5、C5n、C4
内存优化型	R5、R5a、R4、X1e、X1、z1d
加速计算型	P3、P2、G3、F1
存储优化型	H1、I3、D2

由于上述问题的存在，近年来关于云计算资源选取的研究成为一个热点

相关工作：面向实例类型的资源选取

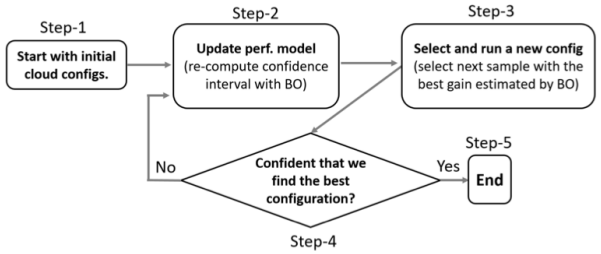
- 问题简述：给定一个应用，如何选择适合该应用的最优实例类型？
- 两类代表性方法
  - 基于性能建模的方式：视作典型的机器学习回归/分类/推荐问题，对应用的性能做精确的建模
  - 基于搜索的方式：不需要深入了解应用本身的特性，使用贝叶斯优化等技术，一步一步逼近最(次)优解

基于性能建模的方式

$$time = \theta_0 + \theta_1 \times (scale \times \frac{1}{machines}) + \theta_2 \times \log(machines) + \theta_3 \times machines$$

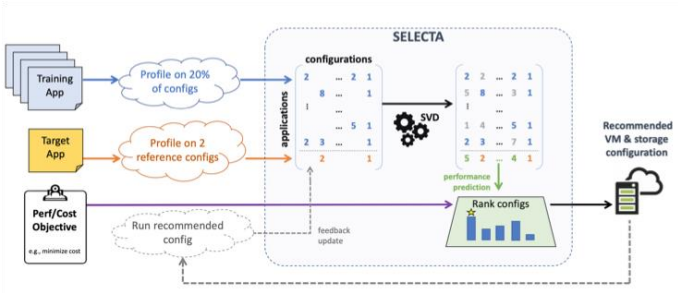
NSDI’16 Ernest

基于搜索的方式



NSDI’17 CherryPick

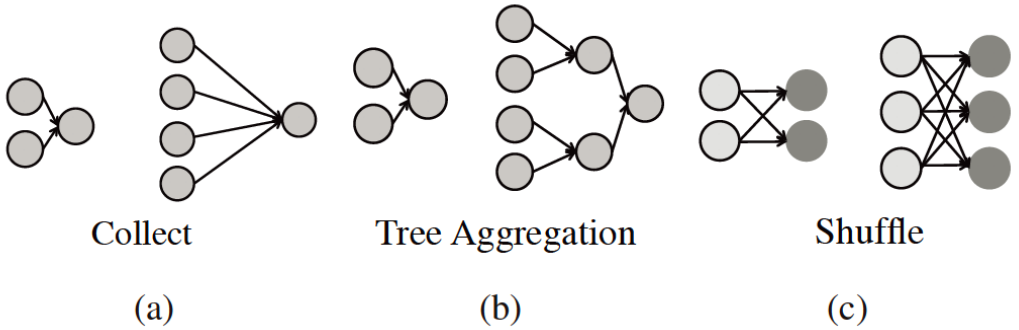
基于推荐的方式



ATC’18 Selecta

Ernest*：面向数据处理作业的性能预测

- 研究动机：部分数据处理作业有固定的行为模式，据此可对应用进行建模，以预测其在某种配置下的性能
- 该工作总结出数据处理作业的三种常见行为模式，使用回归模型对应用的性能建模：
  - Collect: *O(scale * 1 / machines)*
  - Tree Aggregation: *O(log(machines))*
  - Shuffle: *O(machines)*
- 在小数据输入下训练模型，在真实场景（数据量较大）中将模型用于预测

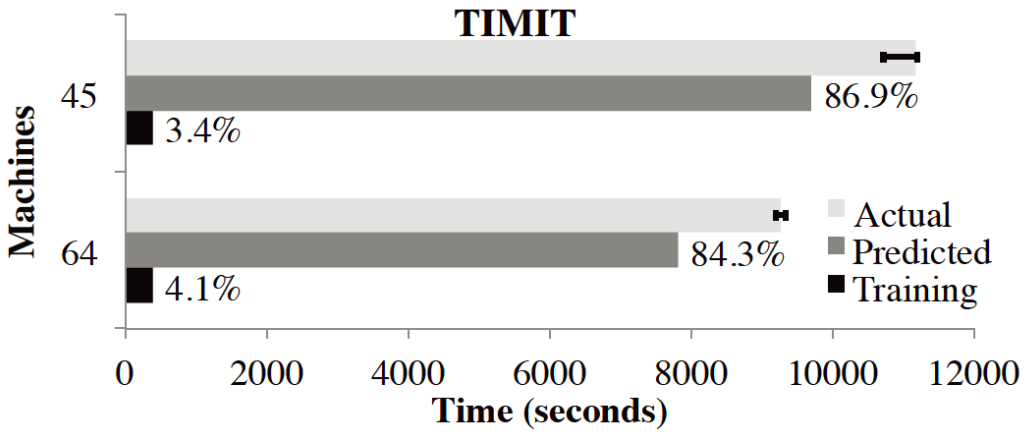


$$time = \theta_0 + \theta_1 \times (scale \times \frac{1}{machines}) + \theta_2 \times \log(machines) + \theta_3 \times machines$$

*Venkataraman et al., Ernest: Efficient Performance Prediction for Large-Scale Advanced Analytics, NSDI'2016

Ernest：面向数据处理作业的性能预测

- 实验设计：在多个workload上对Ernest进行评测，评测指标主要有两个：
  - 预测效果：Predicted Time/Actual Time
  - 训练开销：Training Time/Actual Time
- 实验效果：
  - 预测时间与真实时间的误差比例大多在12%以内
  - 训练开销相对真实事件的占比在5%以内
- 存在的问题：
  - 仅针对某一类数据处理作业，计算模式更为复杂的作业无法通过这种方式进行有效的建模
  - 针对每一种实例都需要训练单独的模型开销较大



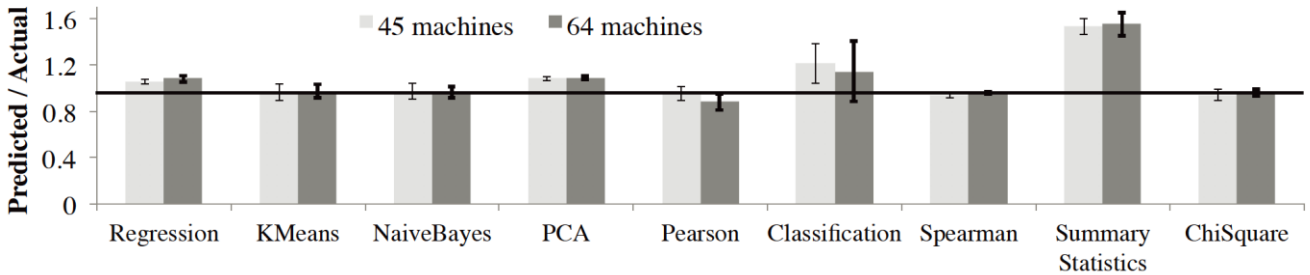
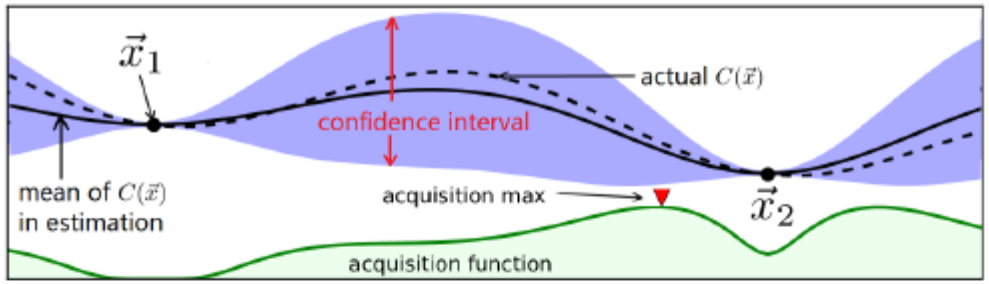


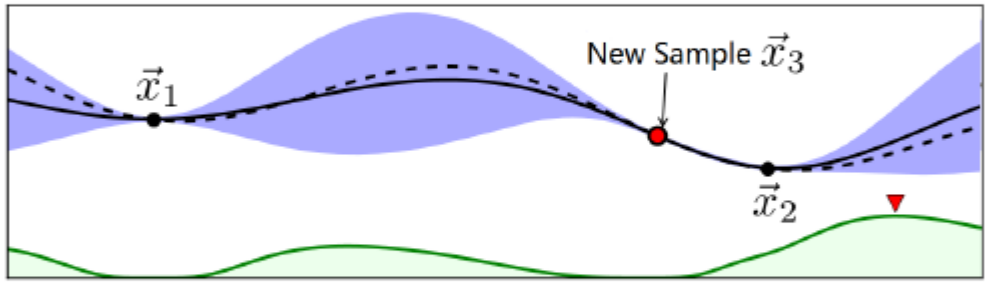
Figure 10: **Prediction accuracy using Ernest for 9 machine learning algorithms in Spark MLlib.**

CherryPick*：基于搜索的云资源选取

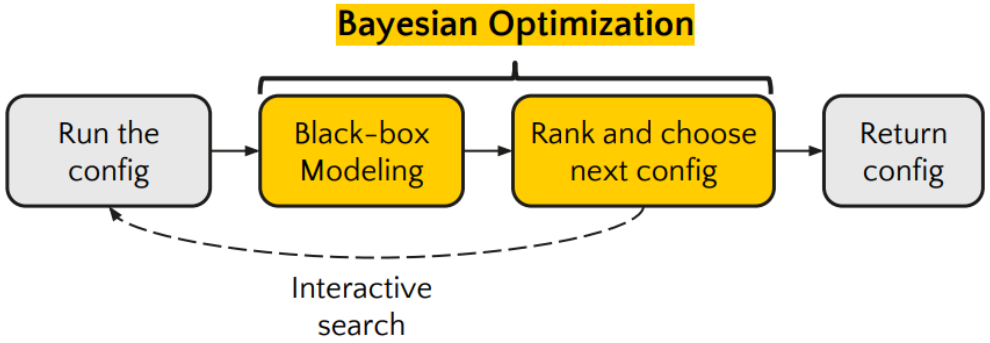
- 研究动机：不同应用在行为特征上差异很大，难以对它们建立统一的性能/花费模型；为不同应用建立不同的模型开销又过大
- 解决方案：
  - 不建立精确的模型，仅使用有限的信息通过搜索（贝叶斯优化）的方式得到“较优”（just accurate enough）的配置
  - 动态地搜索下一个可能的配置，每次搜索都会改变置信区间
  - 当期望的提升（Expected Improvement/EI）低于某个阈值时，停止搜索



(a) t = 2



(b) t = 3



*Alipourfard et al., CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics, NSDI'2017

CherryPick：基于搜索的云资源选取

- 实验设计：
  - benchmark： TPC-DS, TPC-H, TeraSort, etc.
  - 对比的baseline: coordinate search（坐标下降法）， random search, Ernest
- 实验效果：
  - 与baseline相比，准确率更高
  - 使用较小的搜索代价，有45-90%的几率找到最优的配置

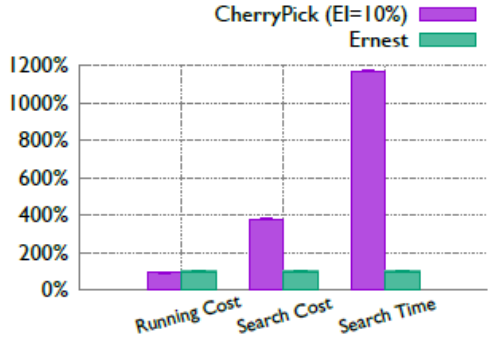


Figure 9: Comparing Ernest to *CherryPick* (TPC-DS).

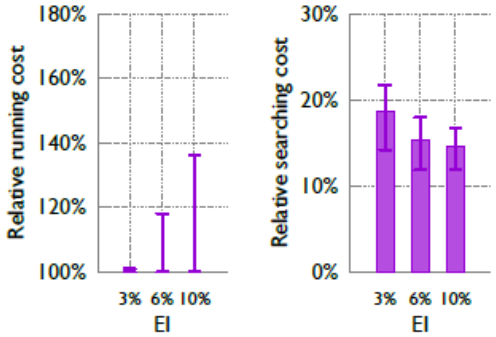
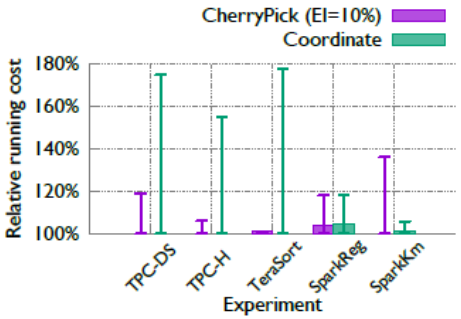
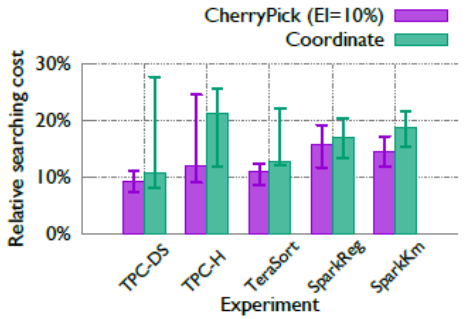


Figure 10: Search cost and running cost of SparkKm with different EI values.



(a) Running cost



(b) Search cost

Figure 7: Comparing *CherryPick* with coordinate descent. The bars show 10th and 90th percentile.

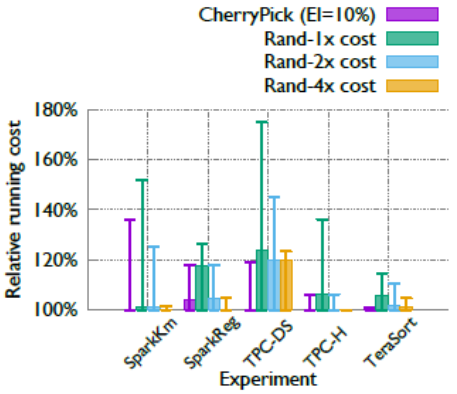
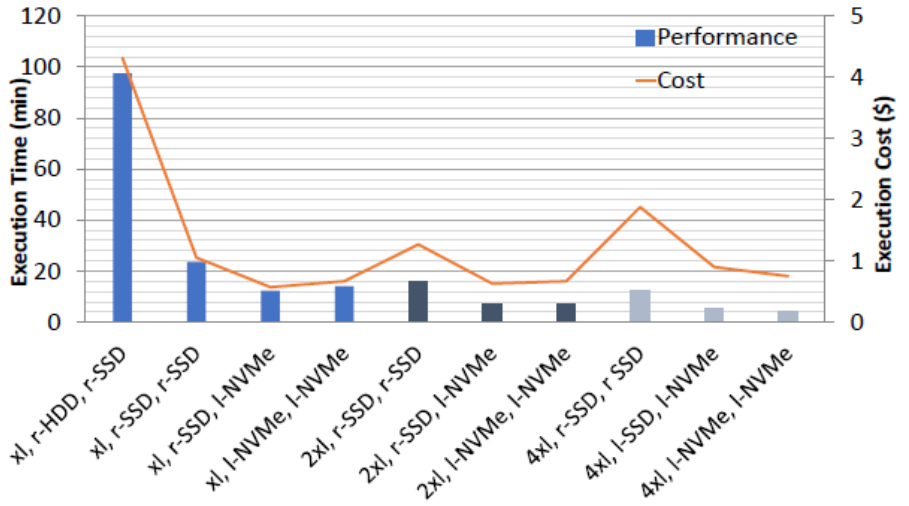


Figure 8: Running cost of configurations by *CherryPick* and random search. The bars show 10th and 90th percentile.

Selecta*：基于协同过滤的云资源推荐

- 研究动机：重点考虑存储介质（Local/Remote HDD/SSD/NVMe）对应用性能的影响，为数据处理作业选取合适的虚拟机配置
- 解决方案：
  - 训练应用在20%的配置上运行，构成一个形如 $(app_i, config_j) \rightarrow perf_{ij}$(性能/花费)的矩阵
  - 目标应用在2个配置上运行，使用奇异值分解，补全矩阵，推测目标应用在其他配置上的性能/花费
  - 根据推测出的值进行排序，给用户返回“最优”的配置



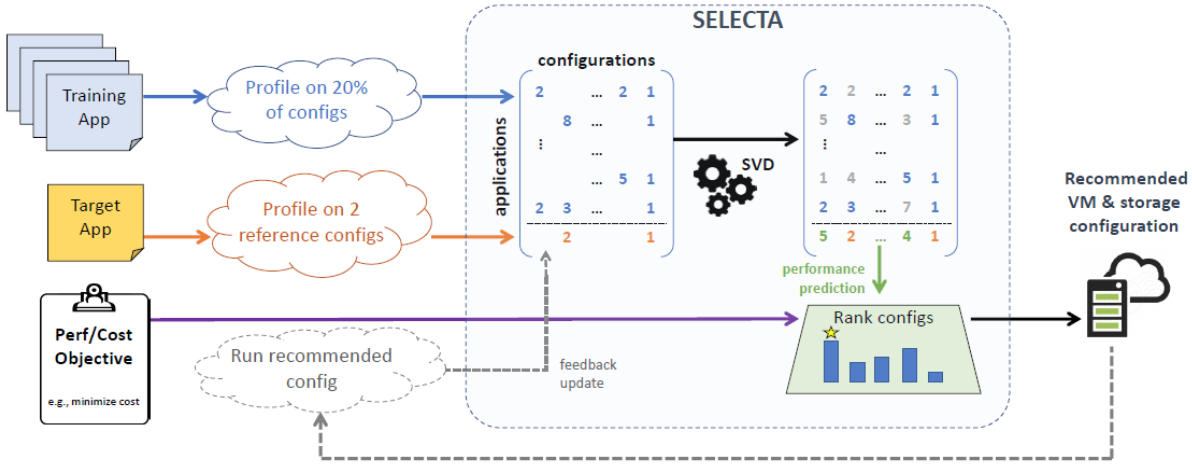


Figure 3: An overview of performance prediction and configuration recommendation with Selecta.

*Klimovic et al., Selecta: Heterogeneous Cloud Storage Configuration for Data Analytics, ATC'2018

Selecta：基于协同过滤的云资源推荐

- 实验设计：
  - 选择6种机型，4种存储类型作为待选配置集合
  - benchmark：TPC-DS, TPC-BB
  - baseline：Random Forest，Max/Min Cost，etc.
- 实验效果：
  - 在做少量训练程序预跑(20%)的情形下，可以达到较高的准确率(>90%)
  - 比baseline的预测准确率更高

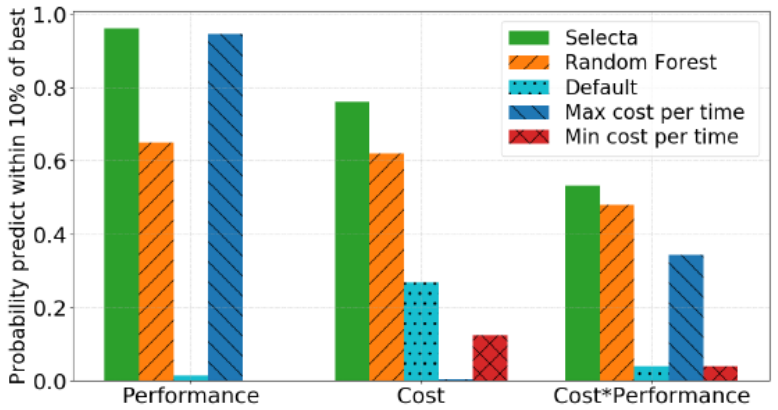
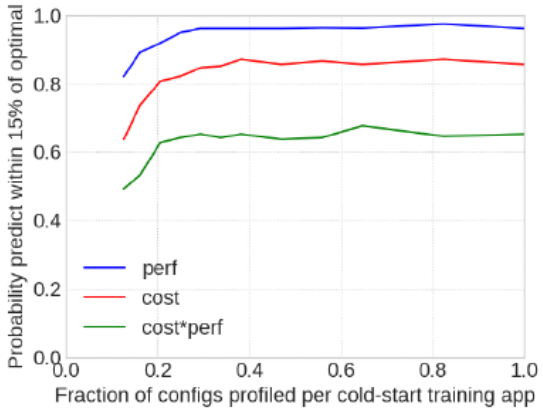
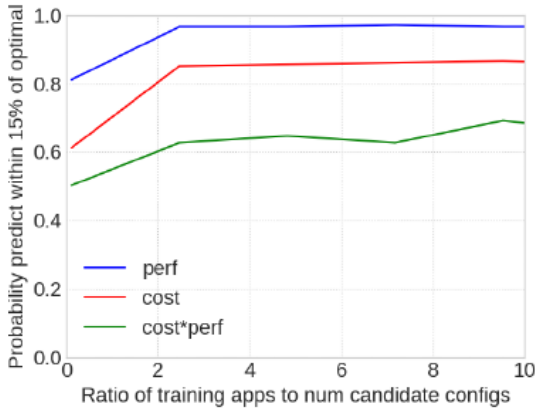


Figure 7: Selecta's accuracy compared to baselines.



(a) Sensitivity to input matrix density in *steady state*: 20% density per row suffices for accurate predictions.



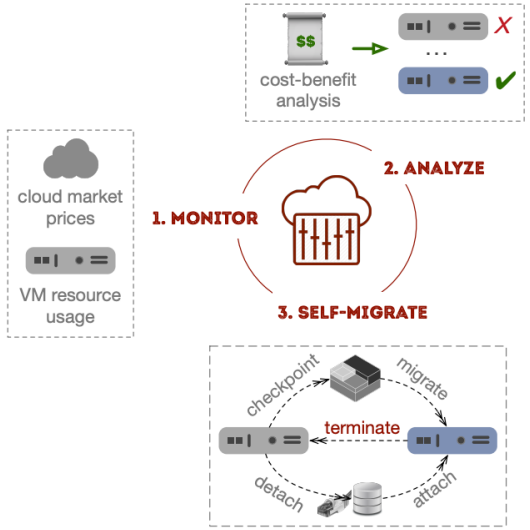
(b) Sensitivity to number of training applications, profiled on all configurations: 2.5× the number of configs suffices.

相关工作：基于计费模式的资源选取

- 问题简述：云厂商提供一些临时的计算资源（如竞价实例），这些资源的计费模式很特殊。利用这类资源，通过一定的策略保证容错，可以进一步降低成本。
- 以AWS Spot Instance为例，主要有两种选取策略：
  - “稳定投资”策略：最小化实例被回收的概率，利用Spot Instance低价格的优势，优化成本
  - “风险投资”策略：最大化实例被回收的概率，以获得厂商的退款*，从而进一步降低成本

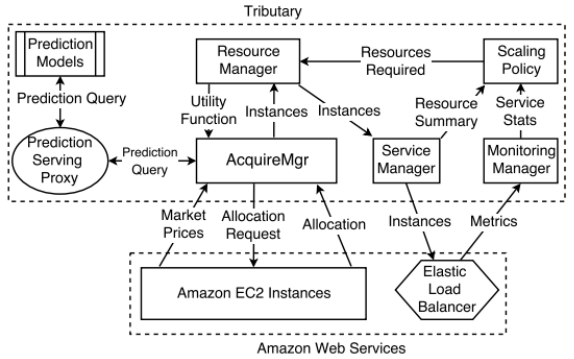
*若Spot实例在第一个小时被回收，用户会获得AWS的全部退款

“稳定投资”策略



SoCC’17 HotSpot

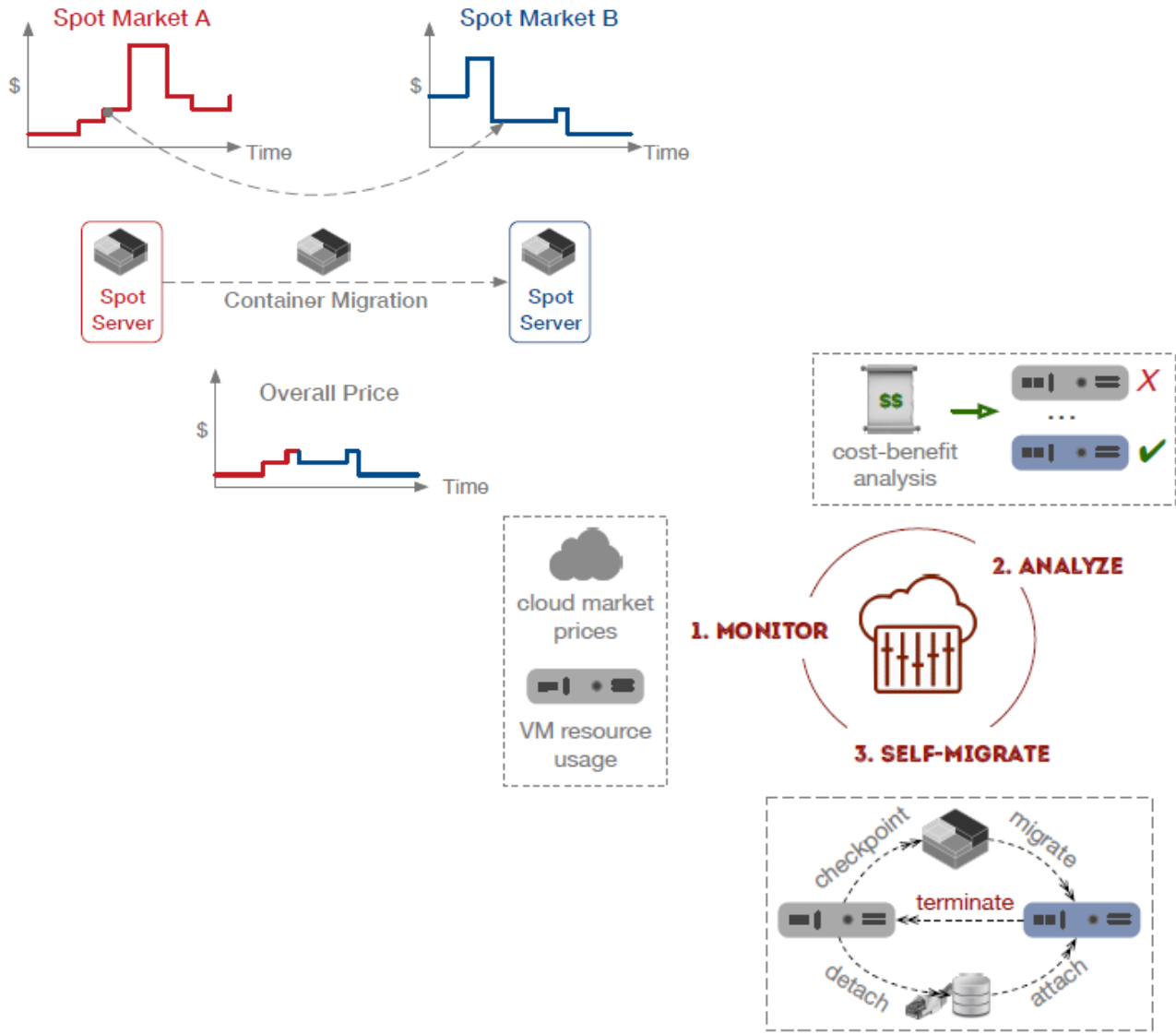
“风险投资”策略



ATC’18 Tributary

HotSpot*：基于竞价实例的容器服务

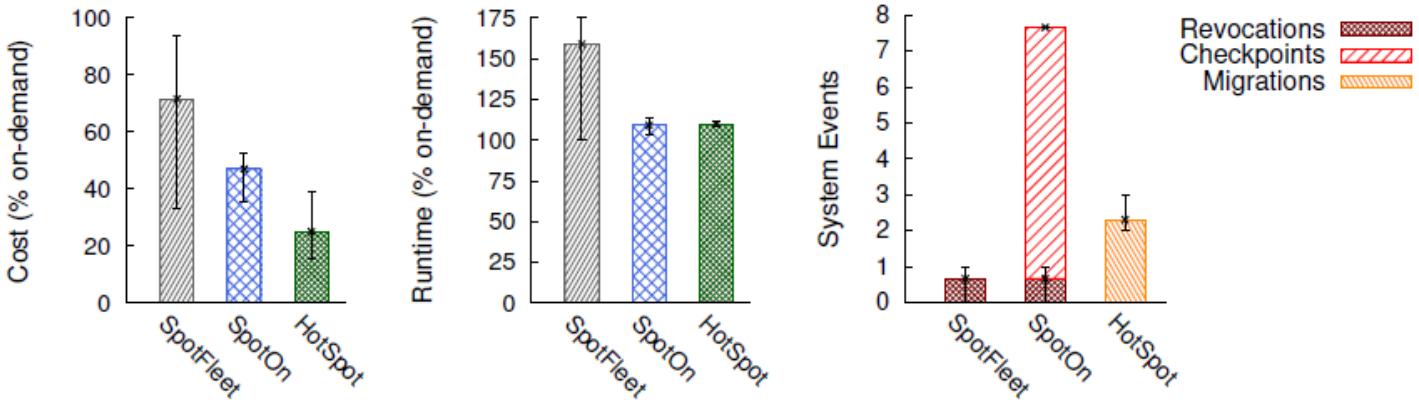
- 研究动机：利用AWS提供的Spot Instance，实时地将容器迁移到更便宜的实例上，实现总体的花费最优
- 实现：
  - 出价远高于Spot实例的市场价格，尽量避免被回收
  - 实时监控当前的Spot实例市场价格，并基于迁移代价和迁移后获得的benefits权衡是否将当前的容器迁移
  - 需要迁移时，先将容器checkpoint到AWS EBS/RAMfs，然后在目标实例上恢复



*Shastri et al., HotSpot: Automated Server Hopping in Cloud Spot Markets, SoCC'2017

HotSpot：基于竞价实例的容器服务

- 实验设计
  - 小规模真实实验
  - 基于Google trace的模拟实验
  - baseline：使用on-demand instance、使用单个spot instance(SpotFleet)、使用单个spot instance且有简单的容错机制(SpotOn)
- 实验效果
  - 整体花费HotSpot最优



Tributary*：基于竞价实例的web service

- 研究动机：面向弹性service类型的作业，利用AWS Spot Instance的退款政策**，通过价格预测的方式选择生命周期短的实例，优化成本
- 技术：
  - 抢占概率预测。使用基于LSTM的模型预测某种Spot实例在一小时内被回收的概率
  - 根据service的负载动态地超配资源，且优先选择生命周期短的Spot实例

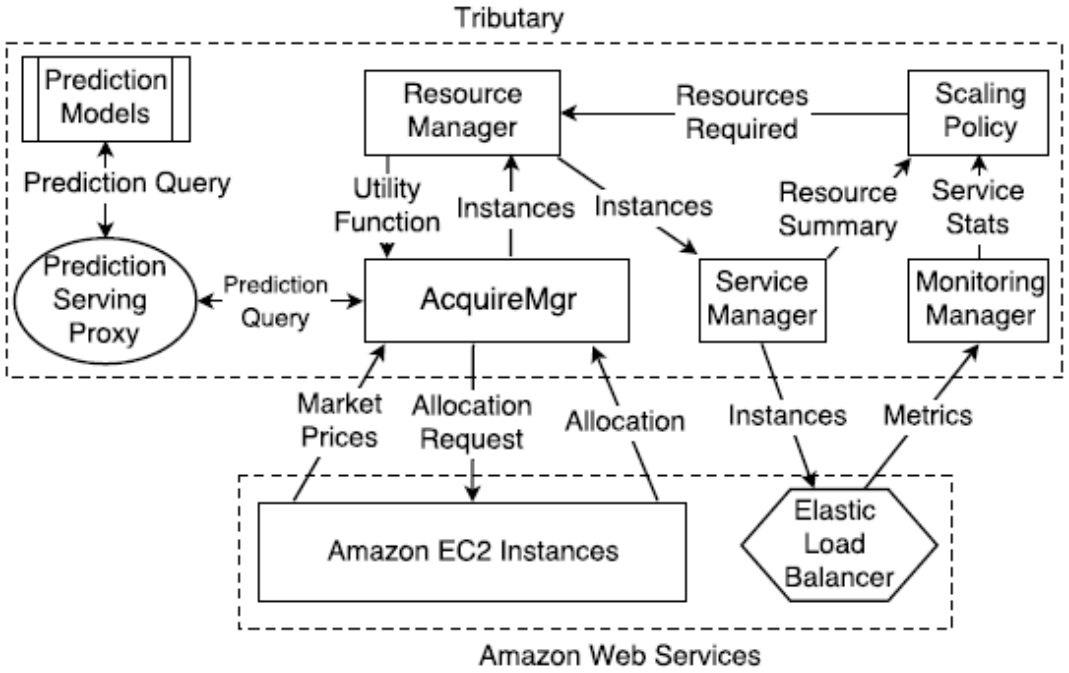
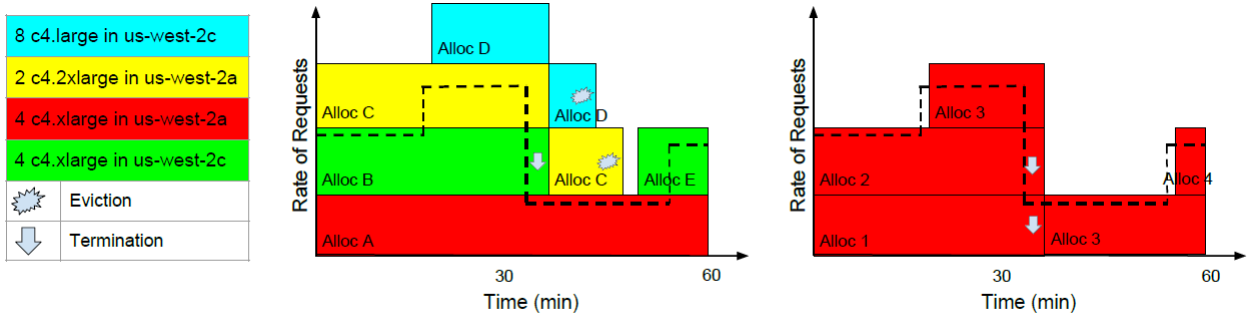


Figure 2: Tributary architecture.



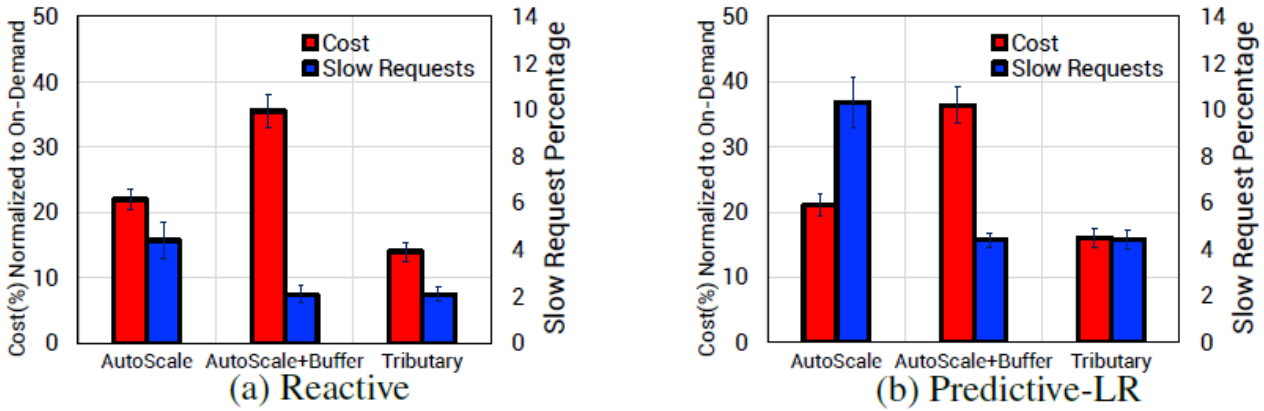
(a) Legend (b) Tributary (c) AutoScale

Harlap et al., Tributary: spot-dancing for elastic services with latency SLOs, ATC'2018

**若Spot实例在第一个小时被回收，用户会获得AWS的全部退款

Tributary：基于竞价实例的web service

- 实验设计（模拟实验）
  - 基于trace的模拟实验：ClarkNet, Berkeley, WITS, WorldCup98
    - baseline: AWS AutoScale
  - 价格预测模块的准确率实验
- 实验效果
  - 性能/花费均优于baseline
  - 预测准确率和F1 Score高于baseline



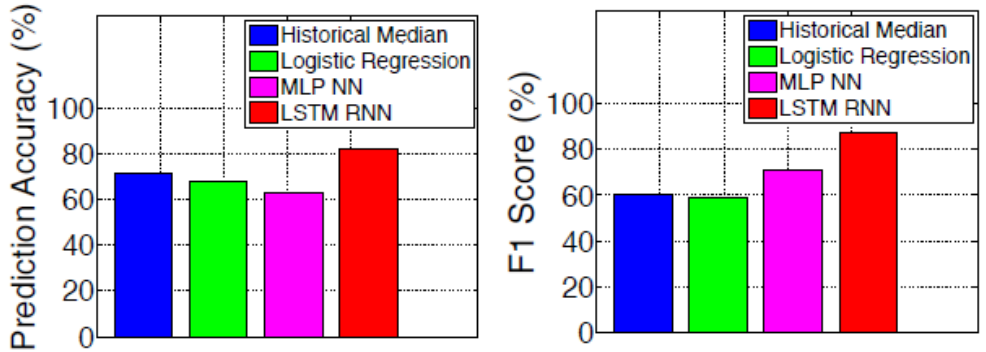


Figure 6: Accuracies and F_1 scores (accounts for data skew) for predicting preemption of AWS spot instances. The LSTM RNN outperforms prior techniques (blue bar) by 11% on the accuracy metric and 27% on the F_1 score metric.

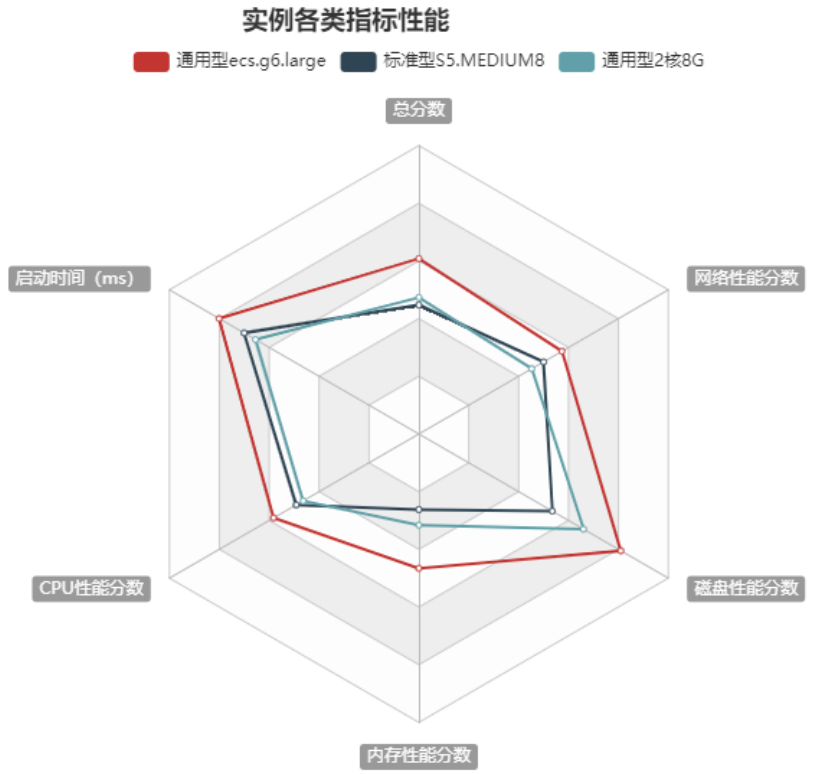
虚拟专云如何选取计算资源?

- 云资源种类、收费模式纷繁复杂，如何根据用户的需求选择合适的资源？
  - 对不同云厂商具有代表性的计算资源进行综合评测，以便于用户进行横向对比
  - 挖掘云厂商的计费模式，对特定的应用制定智能化的策略，寻找花费最优的资源组合
  - 充分利用云的弹性，及时终止不必要的计算，以降低成本

工作1：面向一般场景的计算资源推荐服务

- CloudMeter系统定期自动测评各个云厂商计算资源的性能，用户可方便地从多个维度对比不同厂商的计算资源

比较	实例类型	地域	启动时间	CPU性能	内存性能	磁盘性能	网络性能	总评分
☐	阿里云 通用型ecs.g6.large 2vCPU / 8GB	北京	40.1s	87.5	70	121.7	86.2	365.4
☐	阿里云 计算型ecs.c6.xlarge 4vCPU / 8GB	北京	40.3s	107.5	54.5	121.7	86.4	370.1
☐	阿里云 内存型ecs.r6.large 2vCPU / 16GB	北京	41.6s	88.1	70	120.1	84.1	362.3
☐	腾讯云 标准型S5.MEDIUM8 2vCPU / 8GB	北京	35.1s	73.9	39.4	80.4	75.1	268.9
☐	腾讯云 计算型C3.LARGE8 4vCPU / 8GB	北京	18.0s	109.8	55.5	102.6	63.4	331.4
☐	腾讯云 内存型M5.MEDIUM16 2vCPU / 16GB	北京	37.0s	79	64.5	71.3	89.5	304.4
☐	UCLOUD 通用型2核8G 2vCPU / 8GB	北京	32.8s	69.7	47.5	99.2	68	284.4
☐	UCLOUD 通用型4核8G 4vCPU / 8GB	北京	33.5s	83.8	49.2	101.5	49.1	283.7
☐	UCLOUD 通用型2核16G 2vCPU / 16GB	北京	33.0s	70	46.9	98.3	54.9	270.1



工作2：面向特定领域的计算资源选取与配置技术

- 以构建支持机器学习超参数调整任务的虚拟专用云为例
- 设计理念
  - 使用合适的资源使调参任务高效运行且降低花费，并保证可靠性
    - 利用价格预测，选择即将被回收的竞价实例机器部署训练任务，以降低成本
    - 基于AWS回收前两分钟的通知，将训练的中间结果保存到持久化存储中，保证可用性
  - 充分利用云的弹性，通过模型训练趋势预测的方法提前结束“没有前途”的超参数组合

工作3:基于多云存储的数据资源管理技术DCStore

- 对象存储服务广受欢迎，然而用户使用时面临一些问题
  - 可用性、供应商锁定、隐私安全等
  - 冗余数据增加了开销
- 如何解决这些问题？
  - 利用多云存储，将数据分布在多云上
    - 增强数据可用性，降低锁定风险，保护隐私安全
  - 去冗余：去除数据中的重复内容，降低存储开销和网络开销

工作3：基于多云存储的数据资源管理技术DCStore

③ 云端代理传输服务器

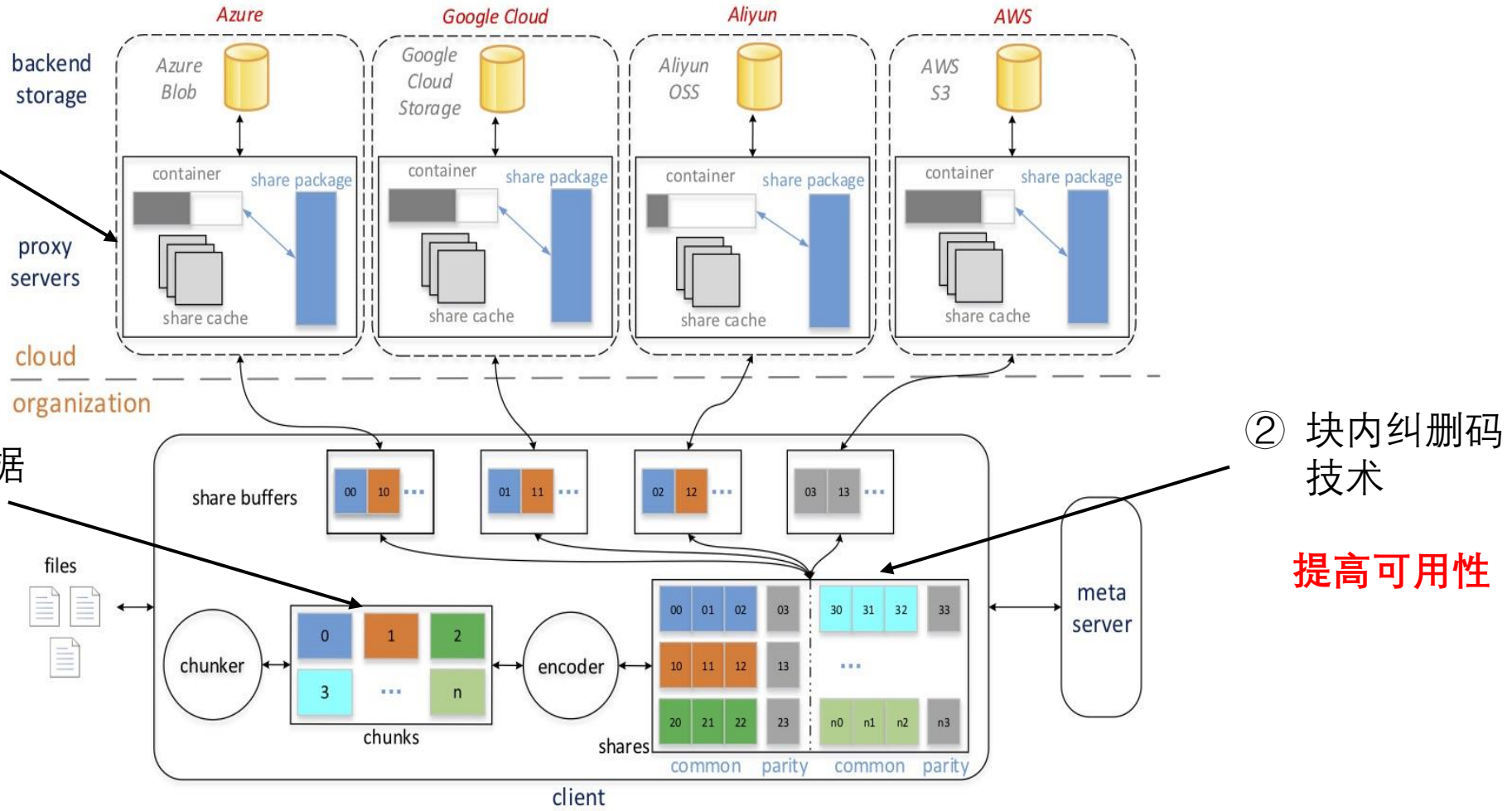
提升性能

① 重复数据删除

优化成本

② 块内纠删码技术

提高可用性



效果：文件冗余度越高（如备份数据），DCStore的效果越好，速度快，费用低

谢谢，敬请批评指正！