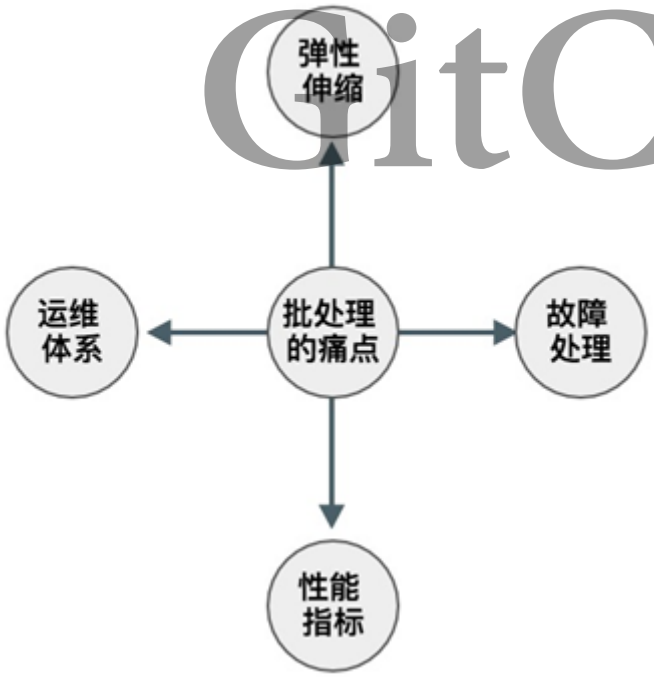
肖德时：分布式任务调度平台实践探讨

今天我想分享的是批处理平台的技术心得。批处理系统从片面的角度来讲类似Linux系统中Cron Table，从大的方向来看是批量业务的调度平台。此次依托数人云3年来对容器技术的积累和对批处理开源项目的整合过程，和大家探讨一下实践分布式任务调度的心路历程。从理论上来讲，做分布式系统并不是企图加快单一任务处理速度，而是通过并行的方式合理利用资源，通过加大对任务的同时批处理业务容量来加快业务的运营速度。

举个典型的例子，就是当前视频直播中需要的视频文件的解码，就是一种典型的批处理业务。本来，批处理业务和容器技术的关系并不紧密，我们通过跨领域的交叉设计，希望通过容器技术的封装能力帮助批处理系统快速建立更多的高密度的批处理运行时环境，更高效的利用资源。

定时任务无处不在，在多任务处理时如何进行秒级调度？与容器如何碰撞？是我比较关心的主题。那么，从我的角度来看，批处理系统的痛点多在4个纬度上受到用户的关注：弹性伸缩、故障处理、运维体系、性能指标。



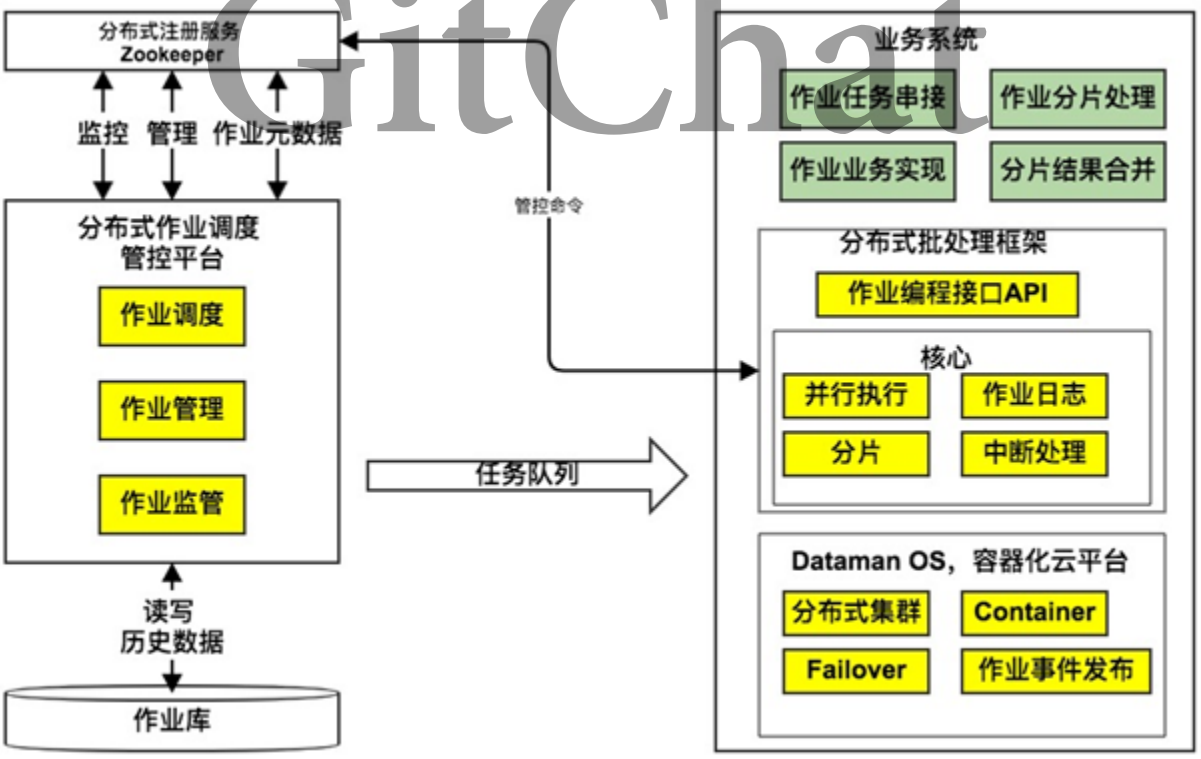
我结合金融行业说下，在金融行业高速发展的今天，业务规模快速扩张，随着业务的发展，需处理的数据量越来越大，后台批处理业务占60%以上，如数据接口导入、数据预处理、估值表生成、凭证生成、对账、日终批处理、报表生成等。

从用户关注的4个纬度切入问题主要表现在：

- 弹性伸缩
    - 批处理作业只能运行在一个服务节点上，无法适应业务发展，此为目前业务系统的最大性能瓶颈点。

- 故障处理
    - 当作业发生故障时，未实现故障自动转移，严重时影响业务进展。
- 性能指标
    - 作业调度与作业执行耦合：作业调度与作业执行线程耦合在一起，随着作业规模的增长，严重影响系统性能，也对开发运维带来一定难度。
    - 联机业务和批处理业务耦合，当在进行比较消耗资源的批处理或批处理服务发生严重故障时直接影响到了在线业务。
- 运维体系
    - 不便于定位作业运行时问题，不便于了解作业运行进展、负载情况、也缺少作业运行时的性能指标，不便于对作业进行调优等。

有了这些分析，在设计实现数人云分布式任务调度平台过程中，采用作业调度与作业执行分离的架构来简化业务系统批处理的开发和运维工作；采用中间件和平台化的思路提升其应用的范围及价值；采用java系统作业的调度、执行与管控；最后产生的效果是实现了以分布式调度为理念而设计的——多服务节点协同并行处理能力及运行环境的适用能力。



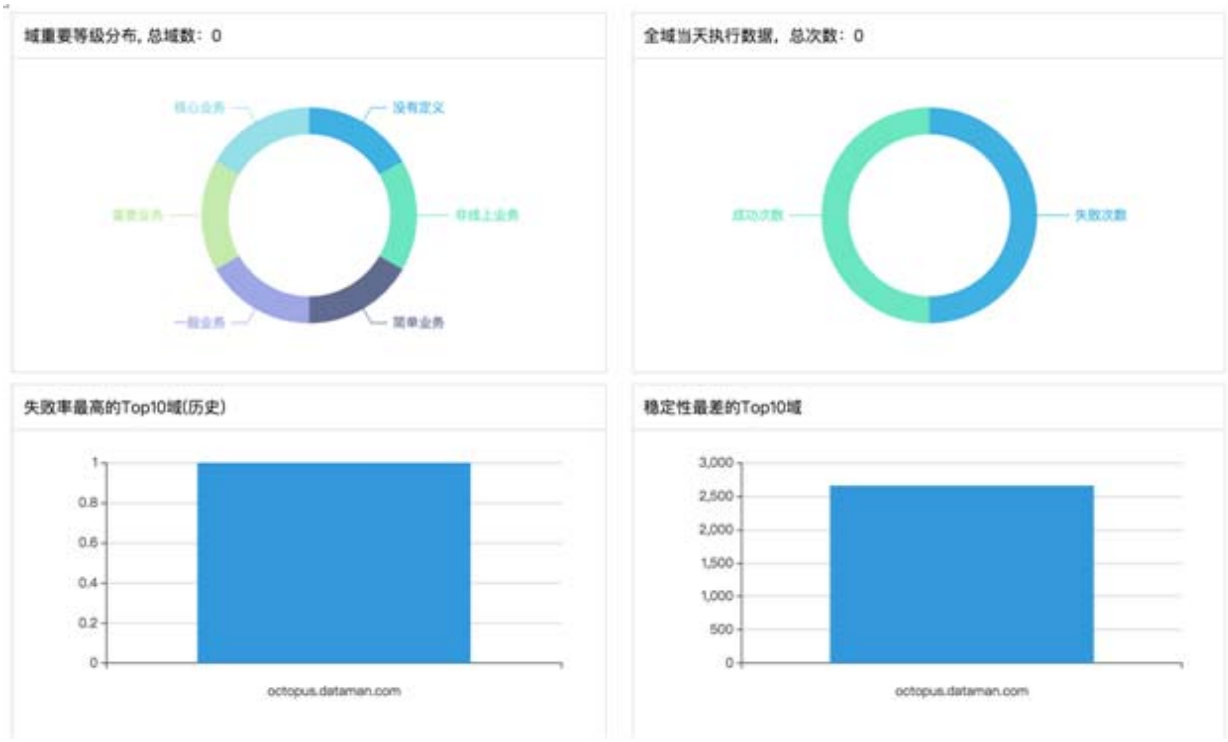
在弹性伸缩方面，我们采用基于容器云平台的目的是快速构建多节点的任务执行节点，容器的好处是封装了一套完整的任务执行的环境，并且可以快速扩容服务节点数量，这个批处理设计模型如何自己实现会需要大量的测试和业务打磨，所以在技术选型上，我们选用了Mesos作为企业级环境的底座。毕竟Apple，MS，Netflix、Uber等大厂都在基于此技术上构建了自己的业务平台。

尤其国内开源项目当当的elastic-job批处理平台(https://github.com/dangdangdotcom/elastic-job)受到很多厂家的采用，给我们提供了一手的学习实践的经验。那么，我们又做了什么优化呢？从设计理念中，我们更多考虑了业务实际场景中的一种情况，就是每次任务处理完之后，是重置当前进程环境还是退出。虽然容器的快速启停确实可以解决这个小问题，但是仍然不够快。

毕竟一个任务的环境初始化是需要消耗时间的，即使用容器启动也是有那么几十毫秒的损耗。另外，我们对Mesos集群的使用，在于提供可以FailOver的高容错环境，并没有直接让Mesos来调度批处理任务。实际上，作业节点注册到ZK后，任务的分发和结果的收集都是在JVM里面解决的，和Mesos集群本身没有关系，减少了对集群系统的直接依赖。后面，我们还会对Kubernetes集群做支持。

在故障处理方面，重要的并不是让任务永远不出错。在创建任务的时候，能提供立即执行一次的操作。让执行结果能立即体现出来，这样，给任务指定指定的时间去跑才不会出错。在创建任务的细节上，比如，把跑批时间参数如：0/5 * * * * ? 预测出固定的时间，让用户看到直接的时间会更好。

对于事后的历史结果的留存也需要做到详细完整，保证故障排错。这块使用统计面板来监控就好。



还有更贴心的作业预警面板也是需要的。

在性能指标方面，大量的工作主要是定义好性能指标并大量压测并调优系统。比如我们的产品定义：

1. 调度频率

   定时作业最快支持5s的时间间隔，对比：公有云如阿里ScheduleX，都是分钟级间隔。

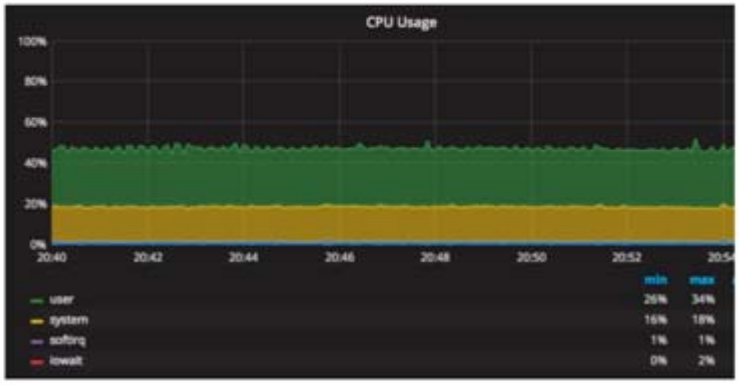
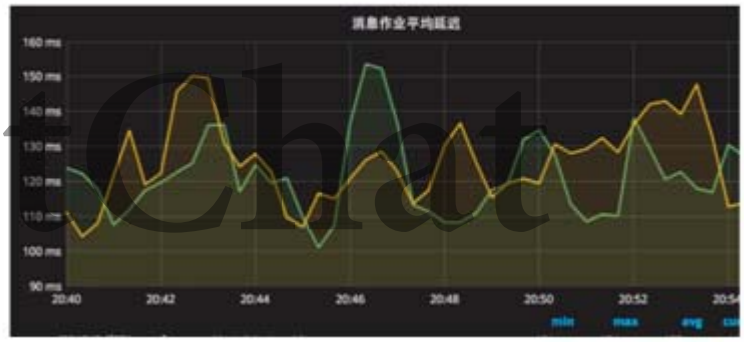
2. 支持作业容量

最多支持管理100个Zookeeper集群，作业总量支持到500K+。

3. 消息作业并发量

支持单节点100K TPS的并发。

通过这个目标，我们通过本地搭建的作业系统环境就可以压测了。压测数据通过grafana展现出来。测试样例如下：

类别	平均值	最小值	最大值	图表
CPU使用情况	47%	43%	55%	
消息作业平均延迟	124ms	101ms	154ms	

在运维体系这块，就是能不能通过一个管理平台就把运维需要管理的事情都考虑进去。比如组织关系的体现，暂停时间的管理，配置数据的备份和查询，ZK元数据的导出备份工作，调用链的跟踪设计实现，各种监控面板的实现。这块难度不大，就是需要考虑的细节会非常多，我们也是在参考和落地实践中摸索这些问题怎么解决。

最后，分布式任务调度平台在企业架构体系里面必不可少的组件。国内企业在经过这几年云计算的高速发展，开始意识到数字化转型过程中必须要经历架构方面的变革。传统的批处理系统已经不能适应业务发展的需要，不防参考业内开源领域的最佳实践构建自己的分布式调度平台。