

Tech FAQ

2020-05-29 星期五 14:15

Q: Vendor和WokerCluster传输文件的方式是怎么样

A: 1.URL地址, 2.本机磁盘挂载的路径, 3.二进制数据流 JobCenter通过addressing和data来做判断, 统一成URI形式。在WorkerNode拉取任务数据时, 返回对应的URI。

Q: Vender和JobCenter以及WokerCluster的通信方式是什么

A: JobCenter提供HTTP接口, 由Vendor和WokerCluster被动调用传输数据

Q: JobCenter的数据存储关系是什么形式

A: 使用Redis存储, 其中只有job的task有关系, 通过job_id来关联, 其他的都是独立模型, 内部使用hash的key->value建立映射

Q: JobCenter的数据存储空间在容器内还是在容器外

A: 启动容器时挂载磁盘作为存储空间, 程序仅提供服务, 不提供存储

Q: 数据如何备份

A: 由于WokerCluster只需要提供服务, 这里不需要做备份。只有JobCenter需要做备份。由于Redis的数据由挂载磁盘提供存储, 所以可以由外部工具进行管理备份。

Q: 节点热备多活和负载均衡怎么实现的

A: JobCenter多个节点时, 由Vendor和WokerCluster通过检查活跃状态来选择节点。对于WokerCluster, 只有当前没有任务时才会去JobCenter获取任务。如果JobCenter节点宕机导致的进行中的任务停止, 需要单独设计Redis的多活同步才能实现热切换。否则需要修复该节点才能恢复这个节点的任务

Q: 如何保证WokerNode取到的文件不会重复

A: Redis中list的push和pop是原子操作

Q: 如何对文件和任务进行优先级的排序, 以及分组, 角色, 优先级的扩展

A: 新建任务时, 完成对任务和文件的存储之后, 按照分组, 角色和优先级建立不同的队列。Vendor发送对自己的分组和角色信息, 以及任务的优先级信息。WokerNode按照自己的分组和角色, 由JobCenter来获取对应下优先级的任务文件。

Q: 如何保证分组不被滥用

A: 建立任务时, 需要传一个group的key, 用于WokerNode匹配group_key, 匹配成功才能拿到任务

Q: 任务结果比较大, 是存到数据库吗

A: 结果存到磁盘文件, 以hash为文件名, 建立对应索引

Q: 每次检查是否有新完成的任务时, 需要遍历所有记录判断完成状态吗

A: 对于task, 在job的assign时, 对每个job生成对应的task的set。每当一个task完成时, 从task的set中移除, 并判断如果移除的set为空之后, 就往job_done的set里添加一条job_id记录, 供vendor获取。这样能保证稳定获取到最新的job和task状态, 也能保持持久的高效

Q: 如果task被get取走之后, 没有finish返回结果, 怎么保证task不会丢失

A: 初始化task后, 会放到waiting集合, 被get取走时, 标记start_time, 并放入pending。如果daemon检测到start_time超时, 则返回空的结果集。也可以主动reset重新push到队列中, 并且放入waiting集合

Q: 如果最后一个task超时了, 导致waiting和pending集合不为空, 怎么判断job是完成的

A: daemon检测超时的task时, 如果从waiting的集合移除之后, 判断集合为空, 则按照finish最后一个task的逻辑标记job完成

Q: 升级和重启服务时, 如何保证任务数据不丢失和异常中断

A: 在monitor中检查任务和文件队列时都为0。为零时, 先停止/卸载vendor服务, 然后是停止/卸载worker服务, 最后停止/卸载pnew服务。升级代码之后, 顺序反过来, 先启动/安装pnew服务, 然后启动/安装worker服务, 最后启动/安装vendor服务。