

任务(Task)

任务模块中处理2类任务：

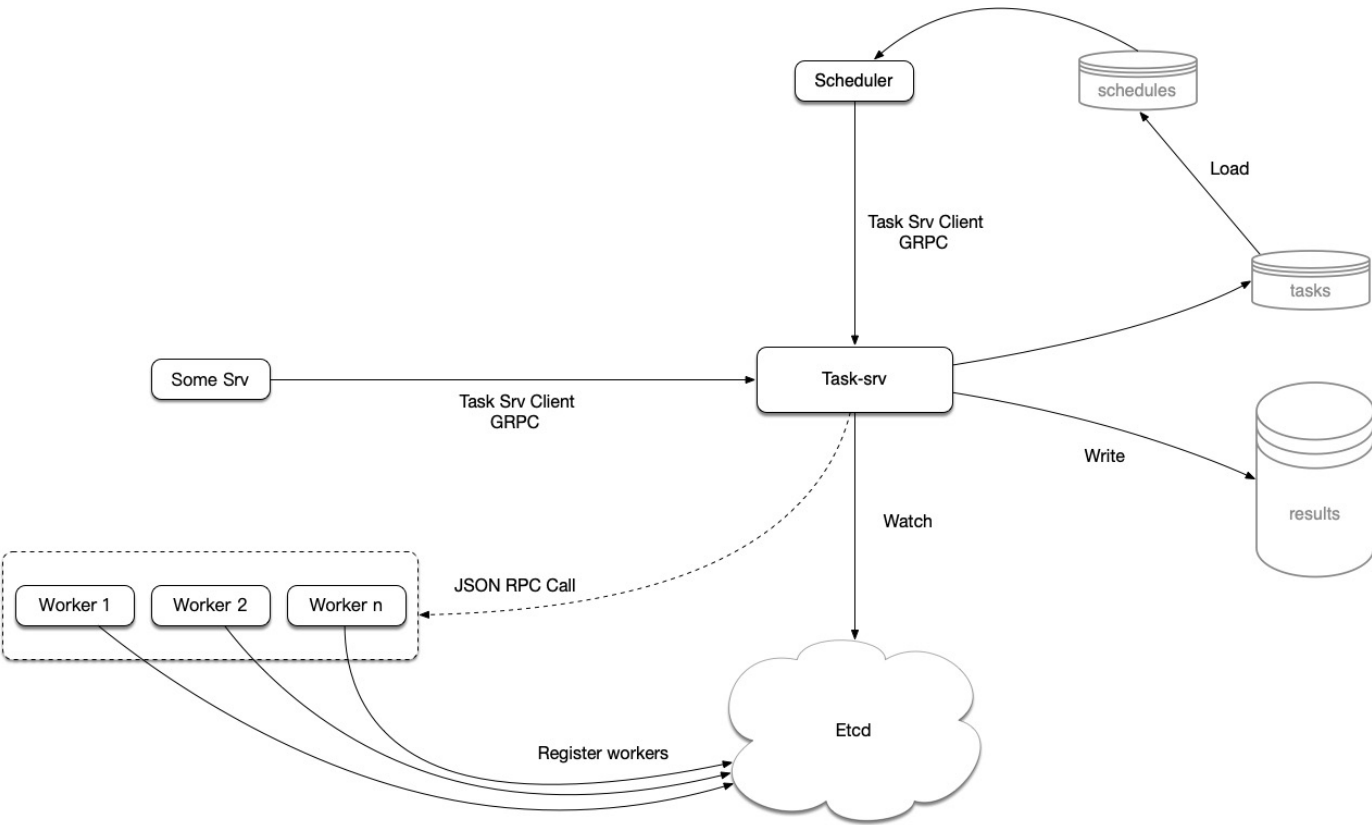
- 1. 计划任务：需要在计划中自动发起/执行的任务。用户可以配置执行时间，如：每天的凌晨3点、每10分钟等等。计划任务的计划格式类似Crontab，所有计划任务配置存放在schedules表内。
- 2. 立即执行任务：调用后就马上执行的任务。可在task模块的前端由访问用户发起，也可以由系统的其他模块通过RPC调用发起。

一、架构及角色：

角色：

角色	中文名	说明
Scheduler	计划任务调度器	负责调度计划任务，启动后它会从schedules表中读出所有计划任务信息，并按照表达式所描述的时间触发对应任务。
Worker	执行器	负责执行具体任务，它可以启动一个或多个实例（每个进程为一个实例）。 启动后它会在etcd中注册自身，注册信息包括：endpoint、服务名等。 执行任务发起时会向results表中创建记录，并实时向该表更新任务状态和任务日志。

架构图：



Worker调用流程简述：

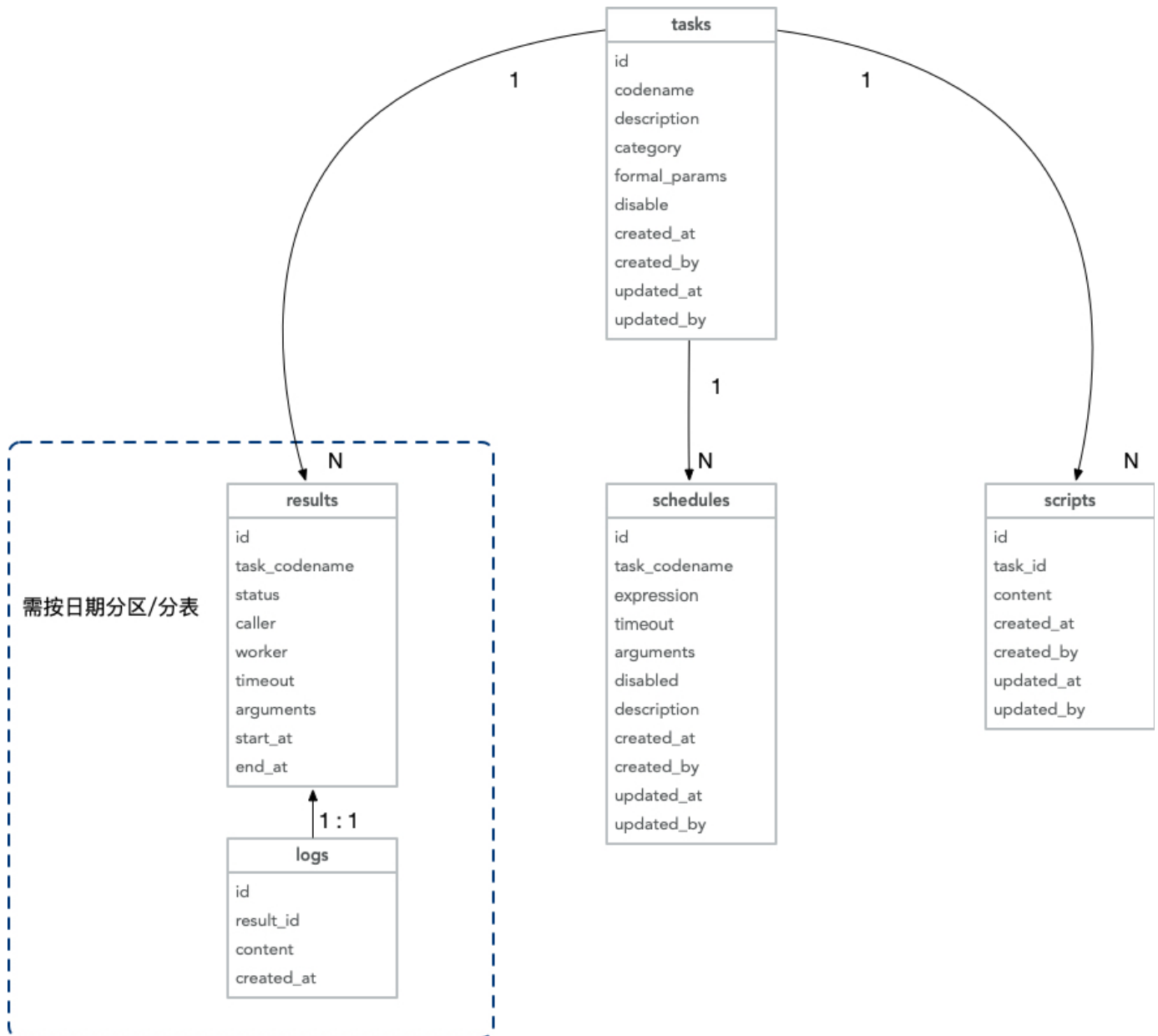
1. 各服务根据TaskWorker类编写自己的Worker实例。Worker实例启动后会以JSON RPC形式提供服务，同时将自己的Endpoint注册到etcd内。（注册的key格式是："/td/eago/workers/<所属服务>.<WorkerName>"）
2. 由任务服务的管理员添加一个任务配置，配置中需要标明：所属模块、任务名、任务描述等。若需要将任务配置成周期性计划任务，还需要在task-scheduler中选择已经添加的任务，并配置调度间隔（语法类似cron）。
3. 任务服务的管理员可以在Web页面中的“任务”菜单直接调用已有任务。
4. 其他服务若想调用任务，则需要通过GRPC调用task服务的CallTask方法。
 1. CallTask方法会先创建任务执行日志记录，此时任务状态变为Pending；然后会验证任务的参数，如果参数不正确或无法反序列化则标记任务状态为调用错误；
 2. 然后CallTask方法会查找任务所对应的Worker，如果找不到Worker，则任务状态会被标记为找不到Worker；
 3. 找到Worker后调用对应Worker的RunTask方法执行任务，同时生成调用合法性信息；
 4. Worker接收到任务调用请求后会验证合法性信息，验证通过后会执行任务；若验证失败Worker会主动与CallTask请求断开链接，随后CallTask方法会将任务标记为调用错误；
5. 当模块下的Worker收到RunTask方法的请求后同样会先返回ACK，使得任务状态变为Running。然后Run方法会执行具体任务操作，同时通过MicroRPC调用task模块的TaskLog方法追加任务日志，任务结束后通过MicroRPC调用task模块的TaskDone方法结束任务，同时给出成功或失败的标记。

二、接口对照表：

子模块名	接口名	功能描述	API调用条件	SRV调用条件
任务 task	NewTask	创建任务	<ul style="list-style-type: none">• 超级用户• 模块管理员 (role==task_admin)	不支持
	RemoveTask	删除任务		
	EditTask	编辑任务		
	ListTasks	列出所有任务		
	CallTask	调用任务	<ul style="list-style-type: none">• 超级用户• 模块管理员 (role==task_admin)	仅供Worker调用
	SetTaskStatus	设置任务状态	不支持	
	AppendTaskLog	追加任务日志		
任务结果分区 ResultPartition	NewPartitionWithCreateTables	创建新的任务结果分区并同时创建结果表	仅测试	不支持
	ListPartition	列出所有任务结果分区		
任务结果 Result	ListResults	列出所有任务结果		不支持
	GetResult	查询任务结果		
任务日志 Log	ListLogs	列出所有任务日志		不支持
执行器 Worker	ListWorkers	列出所有正在运行的执行器		不支持

三、数据库设计：

1. 关系图：



2. 数据库定义：

2.1 tasks（任务）：

字段名	字段显示名	类型	约束项	说明
id	ID	int(11)	PK, Unique index, Not null, Auto inc	
category	任务类别	int(11)	Index, Not null	内置任务=1、bash任务=100、python任务=101
codename	任务代号	varchar(100)	Unique index, Not null	<p>可以理解为任务名，只允许小写字母、整数数字、下划线、小数点；</p> <p>必须以小写字母开头，且至少包含一个"."。</p> <p>*业务意义为“以点分割开模块名与任务名”，如：task.flush_worker、notify.send_email等</p>

description	说明	varchar(500)	Not null, Default ''	
formal_params	形参	varchar(4000)	Not null, Default '{}'	形参必须是JSON格式。如： {"name": "string", "age": "int8"}
disabled	是否禁用	tinyint(1)	Not null, Default 0	
created_at	创建时间	datetime	Not null	
created_by	创建者	varchar(100)	Not null	
updated_at	更新时间	datetime		
updated_by	更新者	varchar(100)	Not null, Default ''	

2.2 schedules（任务计划）：

字段名	字段显示名	类型	约束项	说明
id	ID	int(11)	PK, Unique index, Not null, Auto inc	
task_codename	任务代号	varchar(100)	Index, Not null	
expression	计划表达式	varchar(50)	Not null	与Cront表达式相同的格，如：" <code>*/*3*****</code> "
timeout	超时时间	int(11)	Not null, Default 0	
arguments	实参	varchar(4000)	Not null, Default '{}'	执行任务传递的实际参数，必须是JSON格式。 如：{"name": "test", "ttl": 90}
disabled	是否禁用	tinyint(1)	Not null, Default 0	
description	说明	varchar(500)	Not null, Default ''	
created_at	创建时间	datetime	Not null	
created_by	创建者	varchar(100)	Not null	
updated_at	更新时间	datetime		
updated_by	更新者	varchar(100)	Not null, Default ''	

2.3 result_partitions（结果分区）：

字段名	字段显示名	类型	约束项	说明
id	ID	int(11)	PK, Unique index, Not null, Auto inc	
partition	分区	varchar(100)	Unique Index, Not null	分区通常的格式是年+月，例如：202103、202104...

2.4 results_<partition>（结果）：

字段名	字段显示名	类型	约束项	说明
id	ID	int(11)	PK, Unique index, Not null, Auto inc	
task_codename	任务代号	varchar(100)	Index, Not null	
status	状态	int(11)	Index, Not null	0=成功, 1=准备中, 2=执行中...
caller	调用者	string(100)	Not null	标识谁调用的任务，可以是用户或系统组件
worker	执行器	varchar(100)	Not null	标识由哪个执行器实例执行
timeout	超时时间	int(11)	Not null, Default 0	
arguments	实参	varchar(4000)	Not null, Default '{}'	执行任务传递的实际参数，必须是JSON格式。 如：{"name": "test", "ttl": 90}

start_at	开始时间	datetime	Not null	
end_at	结束时间	datetime		

2.5 logs_<partition>（日志）：

字段名	字段显示名	类型	约束项	说明
id	ID	int(11)	PK, Unique index, Not null, Auto inc	
result_id	结果ID	int(11)	Index, Not null	
content	内容	varchar(4000)	Not null, Default ''	
created_at	创建时间	datetime	Not null	