

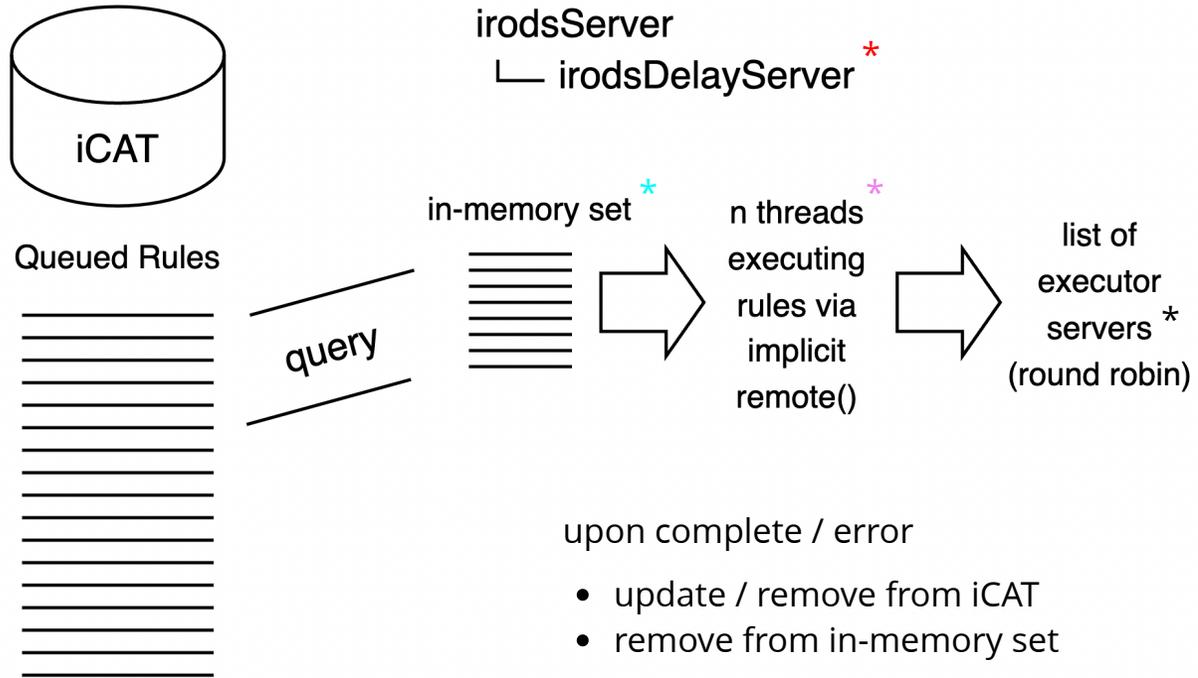


## **Delay Server Migration**

Terrell Russell, Executive Director  
Kory Draughn, Chief Technologist  
iRODS Consortium

July 5-8, 2022  
iRODS User Group Meeting 2022  
Leuven, Belgium

- iRODS rules can be enqueued via 'delay()' to execute later
- Queued Rules live in the iCAT database and are processed by the irodsDelayServer in priority order.
- The irodsDelayServer sleeps most of the time, but spawns an irodsAgent every 30 seconds (by default) to check the Delay Queue for any delayed rules that need to be run.



advanced\_settings

\* delay\_server\_sleep\_time\_in\_seconds

\* maximum\_size\_of\_delay\_queue\_in\_bytes

\* number\_of\_concurrent\_delay\_rule\_executors

\* delay\_rule\_executors

## 4.2.5

- single producer, multiple consumer
- moved from processes to threads
- refactored to use in-memory set

## 4.2.8

- ported to use query processor

## 4.2.9

- rule context stored in iCAT

## 4.3.0

- cron-like facility (restarts)
- implicit remote() + list of executors

- No irodsServer restarts required
  - large deployments are under continuous load
- No double spends
  - only one irodsDelayServer per Zone at any time
- Hands-free migration in case of disaster
  - if/when the irodsDelayServer isn't coming back
- Visibility
  - easy to interrogate, debug

- Use the transactional database to store zone-wide information
  - single source of truth
- Split roles of leader and successor
  - affords different code for servers in different roles
- Identical algorithm running on all iRODS servers
  - each responsible for their own behavior

R\_GRID\_CONFIGURATION Table

<b>namespace</b>	<b>option_name</b>	<b>option_value</b>
delay_server	leader	<hostname>
delay_server	successor	<hostname>

```
$ hostname
05f4be918c0f

$ iadmin get_delay_server_info
{
  "leader": "other.server.example.org",
  "successor": ""
}

$ iadmin set_delay_server $(hostname)

$ iadmin get_delay_server_info
{
  "leader": "other.server.example.org",
  "successor": "05f4be918c0f"
}

$ iadmin get_delay_server_info
{
  "leader": "05f4be918c0f",
  "successor": ""
}
```

```
if self == leader
    if successor defined and not self
        gracefully finish and exit
    else
        if necessary, start irodsDelayServer
else if self == successor
    run health check on leader
    if leader is not running
        promote self to leader in iCAT
    else
        save health stats
else
    if necessary, gracefully finish and exit
```

R\_GRID\_CONFIGURATION Table

namespace	option_name	option_value
delay_server	leader	<hostname>
delay_server	successor	<hostname>



```
$ iadmin set_delay_server <hostname>
```

## Dark Alleys

- database credentials / connection required
- control plane as process / blocking ourselves
- who is the parent process?

## Glory

- better PIDs
- elegant solution

## Future Work

- remove database credentials requirement
- detect and skip a redundant implicit remote()
- advanced setting for sleep time between migration algorithm runs (0=never)

Thank you!

