Adding High Availability to Condor Central Manager

Gabi Kliot
Technion – Israel Institute of Technology
Outline

- Current Condor pool
- Motivation for Highly Available Central Manager
- The solution - HA Daemon
- Performance impacts
- Testing
- Future Work
Current Condor Pool

Collector

Negotiator

Central Manager

Startd and Schedd

Startd and Schedd

Startd and Schedd

Startd and Schedd

Startd and Schedd

Startd and Schedd
Why Highly Available Central Manager

- Central manager is a single-point-of-failure
  - Negotiator’s failure - No additional matches will be possible
  - Collector’s failure – negotiator is out of job, tools querying collector won’t work, etc.

- Our goal
  - Allow continuous pool functioning in case of failure
Highly Available Condor Pool

- Startd and Schedd
- Idle Central Manager
- Active Central Manager
- Idle Central Manager
- Startd and Schedd
- Startd and Schedd
- Startd and Schedd
- Startd and Schedd
Highly Available Central Manager

- Our solution - Highly Available Central Manager
  - **Automatic** failure detection
  - **Transparent** failover to backup matchmaker (no global configuration change for the pool entities)
  - "**Split brain**" reconciliation after network partitions
  - **State replication** between active and backups
  - No changes to Negotiator/Collector code
Highly Available Central Manager
How it works

- Collector’s HA is provided by **redundancy**
- Negotiator’s HA is provided by **HA daemons**
How it works – Election

Collector

HAD

Collector

HAD

Collector

HAD

Collector

Election msg

Election msg

Election msg

Workstation – Startd and Schedd

Workstation – Startd and Schedd
How it works – Election

I’m alive

Leader HAD

Collector

HAD

Collector

Negotiator

I’m alive

HAD

Collector

Workstation – Startd and Schedd

Workstation – Startd and Schedd

©Gabriel Kliot, Technion

Condor week – March 2005
How it works – basic scenario

Collector

Leader HAD

Collector

I’m alive

I’m alive

Negotiator

Idle CM

Active CM

Idle CM

Workstation – Startd and Schedd

Workstation – Startd and Schedd
How it works – crash event

- Collector
- Leader HAD
- Negotiator
- Active CM
- Idle CM
- Workstation – Startd and Schedd

I’m alive

©Gabriel Kliot, Technion
Condor week – March 2005
How it works – crash event

Election

HAD

Collector

Idle CM

Active CM

Collector

Leader HAD

Negotiator

Collector

HAD

Idle CM

Workstation – Startd and Schedd

Workstation – Startd and Schedd

©Gabriel Kliot, Technion

12

Condor week – March 2005
How it works – crash event

LEADER HAD

Collector

Active CM

Workstation – Startd and Schedd

HAD

Collector

Idle CM

Workstation – Startd and Schedd

I’m Alive

Negotiator

Leader HAD

Collector

Negotiator
High Availability Daemon

State machine
Performance impact

- **Stabilization time** – the time it takes for HA daemons to detect failure and fix it. Depends on number of CMs and network performance.
- **HAD_CONNECT_TIMEOUT** – the time it takes to establish TCP connection (depends on network type, presence of encryption, etc…)
- Assuming it takes up to 2 seconds to establish TCP connection and 2 CMs are used - new Negotiator is up and running after **48** seconds.
Testing

- Special **automatic distributed testing framework** was built:
  - simulation of node crashes
  - network disconnections
  - network partition and merges

- **Extensive testing** effort:
  - distributed testing on 5 linux machines in the Technion
  - interactive distributed testing in Wisc pool
  - automatic testing with NMI framework

- Already **deployed and fully functioning** for 3 weeks on our production pool in the Technion
Future development

- HAD publishing in Collectors
  - condor_status –had
- Accounting file replication
  - current solution is provided for NFS
- Software High Availability
Collaboration with Condor team

- Compliance with high Condor coding standards
- Peer-reviewed code
- Integration with NMI framework
- Automation of testing
- Open-minded attitude of Condor team to numerous requests and questions
- Unique experience of working with large peer-managed group of talented programmers
Collaboration with Condor team

This work was a collaborative effort of:

- **Distributed Systems Laboratory in Technion**
  - Prof Assaf Schuster, Mark Silberstein, Gabi Kliot, Svetlana Kantorovitch, Dedi Carmeli, Artiom Sharov

- **Condor team**
  - Prof Miron Livni, Nick, Todd, Derek, Erik, Carey, Peter, Becky, Parag, Zack, Dan
You should definitely try it!

- Part of the official 6.7.6 development release
- Full support by the Technion team
- More information:
  - [http://dsl.cs.technion.ac.il/projects/gozal/project_pages/ha/ha.html](http://dsl.cs.technion.ac.il/projects/gozal/project_pages/ha/ha.html)
  - more details + configuration on my tutorial tomorrow
- Contact:
  - gabik@cs.technion.ac.il
  - condor-users@cs.wisc.edu
In case of time
How it works – basic scenario

- **Startd and Schedd**: Runs on workstations.
- **Collector**: Sends heartbeat messages to HAD.
- **Negotiator**: Manages active and backup servers.
- **HAD**: High Availability دائما (leader and follower).
- **Machine A**: Backup server.
- **Machine B**: Active server.
- **Machine C**: Backup server.
- **Workstation**: Runs Startd and Schedd.

Connections:
- Workstation to Collector
- Collector to HAD
- HAD to Negotiator
- Negotiator to Collector
- Collector to HAD
- HAD to Workstation

Heartbeat messages: "I'm alive"
How it works – crash event
Usability and administration

- Configuration sanity check perl script
- Disable HAD perl script
- Detailed manual section
- Full support by Technion team