Untangling and Restructuring CTDB

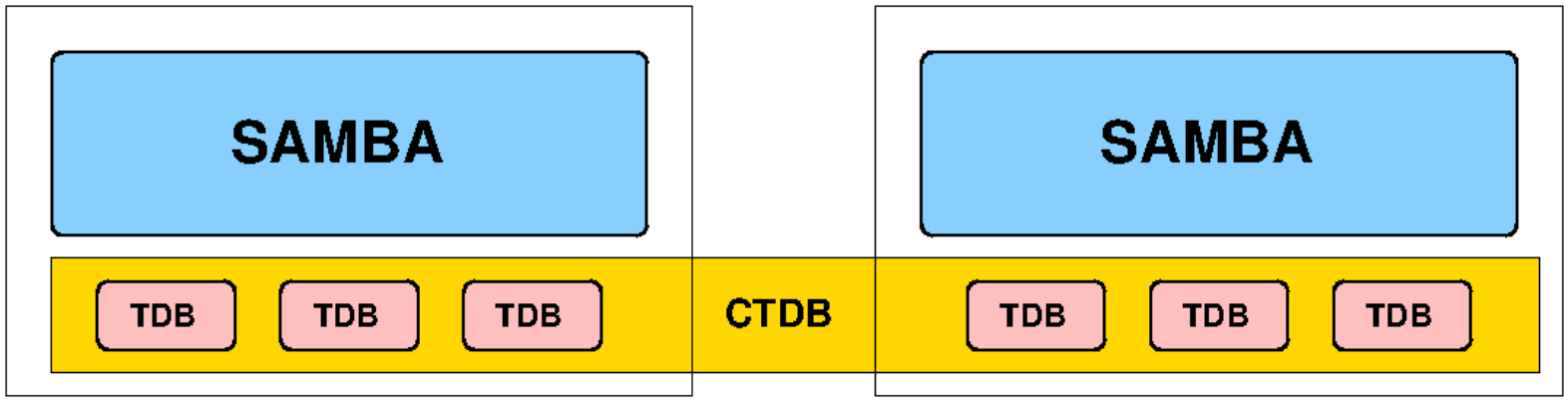
Martin Schwenke <martin@meltin.net>

Samba Team
IBM (Australia Development Laboratory, Linux Technology Center)

What are we talking about?

What are we talking about?

What is CTDB?



What are we talking about?

What does CTDB do?

What are we talking about?

What does CTDB do?

- Cluster membership and leadership

What are we talking about?

What does CTDB do?

- Cluster membership and leadership
- Cluster database and database recovery

What are we talking about?

What does CTDB do?

- Cluster membership and leadership
- Cluster database and database recovery
- Cluster-wide messaging transport for Samba

What are we talking about?

What does CTDB do?

- Cluster membership and leadership
- Cluster database and database recovery
- Cluster-wide messaging transport for Samba
- Service management and monitoring

What are we talking about?

What does CTDB do?

- Cluster membership and leadership
- Cluster database and database recovery
- Cluster-wide messaging transport for Samba
- Service management and monitoring
- IP address management, failover and consistency checking

The plan?

The plan?

What is the goal?

The plan?

What is the goal?

- CTDB scalability and performance

The plan?

What is the goal?

- CTDB scalability and performance
- Reduce barrier to entry for new CTDB developers

The plan?

What is the goal?

- CTDB scalability and performance
- Reduce barrier to entry for new CTDB developers
- Encourage wider use

The plan?

What is the goal?

- CTDB scalability and performance
- Reduce barrier to entry for new CTDB developers
- Encourage wider use
- Parallelise CTDB database daemon?

The plan?

What is the goal?

- CTDB scalability and performance
- Reduce barrier to entry for new CTDB developers
- Encourage wider use
- Parallelise CTDB database daemon?
- Remove non-database functions from database daemon

The plan?

What is the goal?

- CTDB scalability and performance
- Reduce barrier to entry for new CTDB developers
- Encourage wider use
- Parallelise CTDB database daemon?
- Remove non-database functions from database daemon
- Cleanly split out cluster, service, IP management

The plan?

How do we get there?

The plan?

How do we get there?

- I told you last year!

The plan?

How do we get there?

- I told you last year!
- So far it has looked very little like I described. . .

The plan?

How do we get there?

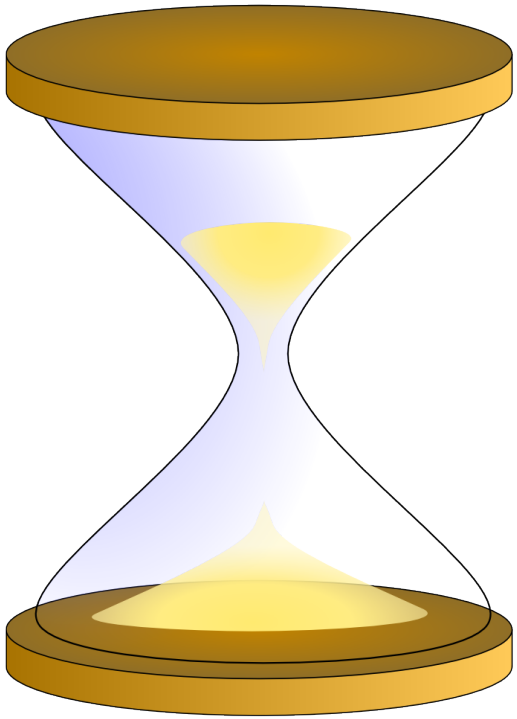
- I told you last year!
- So far it has looked very little like I described...
- Slow progress...

The plan?

How do we get there?

- I told you last year!
- So far it has looked very little like I described...
- Slow progress...
- ...one bite at a time...

Twelve months of untangling



Twelve months of untangling

What has been happening?

- Recovery helper
- NFS support factoring
- IP allocation
- NAT gateway
- LVS support
- TCP connection killing
- Recovery lock
- Monitoring in recovery daemon

Twelve months of untangling

Recovery helper

- Actually a bug fix to avoid recovery deadlock...

Twelve months of untangling

Recovery helper

- Actually a bug fix to avoid recovery deadlock...
- ...more from Amitay later

Twelve months of untangling

Recovery helper

- Actually a bug fix to avoid recovery deadlock...
- ...more from Amitay later
- New protocol and client code to support

Twelve months of untangling

Recovery helper

- Actually a bug fix to avoid recovery deadlock...
- ...more from Amitay later
- New protocol and client code to support
- New helper `ctdb_recovery_helper`

Twelve months of untangling

Recovery helper

- Actually a bug fix to avoid recovery deadlock...
- ...more from Amitay later
- New protocol and client code to support
- New helper `ctdb_recovery_helper`
- All new code — no nested event loops!

Recovery helper

- Actually a bug fix to avoid recovery deadlock...
- ...more from Amitay later
- New protocol and client code to support
- New helper `ctdb_recovery_helper`
- All new code — no nested event loops!
- Drop in replacement for existing recovery code

Twelve months of untangling

NFS support

- This change is confined to scripts. . .

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`

Twelve months of untangling

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`
- Refactored into single `60.nfs`

Twelve months of untangling

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`
- Refactored into single `60.nfs`
- Now have `CTDB_NFS_CALLOUT` configuration variable

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`
- Refactored into single `60.nfs`
- Now have `CTDB_NFS_CALLOUT` configuration variable
- Default is `nfs-linux-kernel-callout`

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`
- Refactored into single `60.nfs`
- Now have `CTDB_NFS_CALLOUT` configuration variable
- Default is `nfs-linux-kernel-callout`
- Sample `nfs-ganesha-callout`

NFS support

- This change is confined to scripts...
- We had `60.nfs` and `60.ganesha`
- We had a request for `60.glusternfs`
- Refactored into single `60.nfs`
- Now have `CTDB_NFS_CALLOUT` configuration variable
- Default is `nfs-linux-kernel-callout`
- Sample `nfs-ganesha-callout`
- José has been working on `nfs-ganesha-callout` recently

IP allocation

- IP allocation algorithm depends on IP addresses and node states

Twelve months of untangling

IP allocation

- IP allocation algorithm depends on IP addresses and node states
- CTDB data structures were deep in the code

IP allocation

- IP allocation algorithm depends on IP addresses and node states
- CTDB data structures were deep in the code
- Several interface points between IP allocation algorithm and surrounding code

Twelve months of untangling

IP allocation

- IP allocation algorithm depends on IP addresses and node states
- CTDB data structures were deep in the code
- Several interface points between IP allocation algorithm and surrounding code
- Introduced more abstract data structures

IP allocation

- IP allocation algorithm depends on IP addresses and node states
- CTDB data structures were deep in the code
- Several interface points between IP allocation algorithm and surrounding code
- Introduced more abstract data structures
- IP allocation is now separate "module"

IP allocation

- IP allocation algorithm depends on IP addresses and node states
- CTDB data structures were deep in the code
- Several interface points between IP allocation algorithm and surrounding code
- Introduced more abstract data structures
- IP allocation is now separate "module"
- Next step: IP allocation daemon?

NAT gateway

- Had daemon support: NAT gateway master capability

Twelve months of untangling

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword
- No capability needed, so no daemon support!

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword
- No capability needed, so no daemon support!
- New helper script: "`ctdb_natgw master|list|status`"

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword
- No capability needed, so no daemon support!
- New helper script: "`ctdb_natgw master|list|status`"
- "`ctdb natgw master|list|status`" runs helper

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword
- No capability needed, so no daemon support!
- New helper script: "`ctdb_natgw master|list|status`"
- "`ctdb natgw master|list|status`" runs helper
- NAT gateway event script also calls out to helper

NAT gateway

- Had daemon support: NAT gateway master capability
- "`ctdb natgwlist`" calculated NAT gateway master node
- Capability unset on a node indicated "slave-only"
- Observed that NAT gateway nodes file lines could be augmented with "`slave-only`" keyword
- No capability needed, so no daemon support!
- New helper script: "`ctdb_natgw master|list|status`"
- "`ctdb natgw master|list|status`" runs helper
- NAT gateway event script also calls out to helper
- NAT gateway support now reduced to 2 non-core scripts

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP
- "`ctdb lvsmaster`" calculated LVS master node

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP
- "`ctdb lvsmaster`" calculated LVS master node
- Re-implemented using same model as NAT gateway

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP
- "`ctdb lvsmaster`" calculated LVS master node
- Re-implemented using same model as NAT gateway
- New helper script: "`ctdb_lvs master|list|status`"

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP
- "`ctdb lvsmaster`" calculated LVS master node
- Re-implemented using same model as NAT gateway
- New helper script: "`ctdb_lvs master|list|status`"
- LVS support reduced to 2 non-core scripts

Twelve months of untangling

LVS

- Had daemon support: LVS capability, single public IP
- "`ctdb lvsmaster`" calculated LVS master node
- Re-implemented using same model as NAT gateway
- New helper script: "`ctdb_lvs master|list|status`"
- LVS support reduced to 2 non-core scripts
- Simplified IP takeover code due to absence of single public IP

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin
- Much faster than talking to daemon

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin
- Much faster than talking to daemon
- `SOCK_PACKET` drops packets...

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin
- Much faster than talking to daemon
- `SOCK_PACKET` drops packets...
- Bidirectional killing, packets got mixed up!

Twelve months of untangling

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin
- Much faster than talking to daemon
- `SOCK_PACKET` drops packets...
- Bidirectional killing, packets got mixed up!
- Some internal filtering and tuning needed

TCP connection killing

- Was combination of "`ctdb killtcp`" and daemon support
- Daemon side validated server-side IP address
- Daemon also sent "tickle ACKS", listened for responses and sent RSTs
- No need to validate server-side IP address
- New helper `ctdb_killtcp` reads connections from stdin
- Much faster than talking to daemon
- `SOCK_PACKET` drops packets...
- Bidirectional killing, packets got mixed up!
- Some internal filtering and tuning needed
- Helper invoked directly from event script

Twelve months of untangling

Recovery lock

- `fcntl(2)` lock on cluster filesystem

Twelve months of untangling

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss

Twelve months of untangling

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"

Twelve months of untangling

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"
- Want to split this...

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"
- Want to split this...
- ...and allow other forms of cluster mutex than `fcntl(2)` lock

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"
- Want to split this...
- ...and allow other forms of cluster mutex than `fcntl(2)` lock
- New helper `ctdb_mutex_fcntl_helper`

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"
- Want to split this...
- ...and allow other forms of cluster mutex than `fcntl(2)` lock
- New helper `ctdb_mutex_fcntl_helper`
- Or: `CTDB_RECOVERY_LOCK=\`
  `"!/my/cluster/mutex/helper args ..."`

Recovery lock

- `fcntl(2)` lock on cluster filesystem
- Lock is taken on first recovery...
- ...and released on election loss
- Combination of "cluster master lock" and "recovery lock"
- Want to split this...
- ...and allow other forms of cluster mutex than `fcntl(2)` lock
- New helper `ctdb_mutex_fcntl_helper`
- Or: `CTDB_RECOVERY_LOCK=\`
  `"!/my/cluster/mutex/helper args ..."`
- Recovery lock not split yet

Monitoring in recovery daemon

- Recovery daemon runs `main_loop` at 1 second intervals

Monitoring in recovery daemon

- Recovery daemon runs `main_loop` at 1 second intervals
- Cluster leadership/elections, nodes states/flags, database recovery, IP failover & monitoring are all intertwined

Twelve months of untangling

Monitoring in recovery daemon

- Recovery daemon runs `main_loop` at 1 second intervals
- Cluster leadership/elections, nodes states/flags, database recovery, IP failover & monitoring are all intertwined
- Continuously revisit and improve. . .

The pattern?

The pattern?

Helpers!

The pattern?

Helpers!

- Helpers!

The pattern?

Helpers!

- Helpers!
- Helpers!

The pattern?

Helpers!

- Helpers!
- Helpers!
- Call-outs!

The pattern?

Helpers!

- Helpers!
- Helpers!
- Call-outs!
- Helpers!

The pattern?

Helpers for incremental re-write

The pattern?

Helpers for incremental re-write

- Helpers can be used for writing shiny new code...

The pattern?

Helpers for incremental re-write

- Helpers can be used for writing shiny new code...
- ...to replace self-contained parts of the code

The pattern?

Helpers for incremental re-write

- Helpers can be used for writing shiny new code...
- ...to replace self-contained parts of the code
- Works well for infrequently executed code

The pattern?

Helpers for incremental re-write

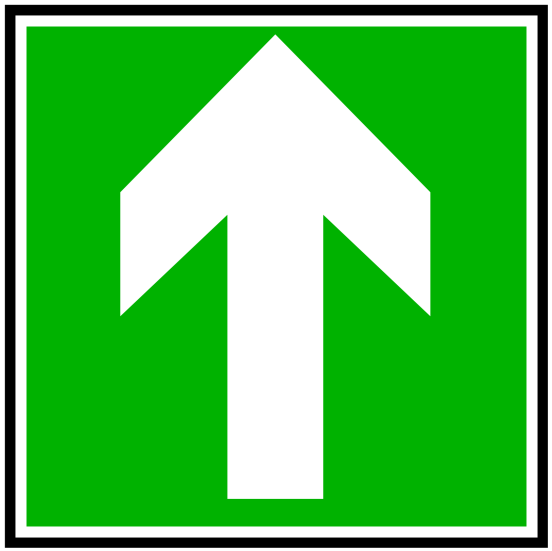
- Helpers can be used for writing shiny new code...
- ...to replace self-contained parts of the code
- Works well for infrequently executed code
- Most of the code we want to move out is (relatively) infrequently executed...

The pattern?

Helpers for incremental re-write

- Helpers can be used for writing shiny new code...
- ...to replace self-contained parts of the code
- Works well for infrequently executed code
- Most of the code we want to move out is (relatively) infrequently executed...
- A lot of it needs to be made more self-contained first!

What's next?



What's next?

Split the recovery lock

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"
- What do you do when it fails?

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"
- What do you do when it fails?
- Split recovery lock into separate cluster & recovery locks

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"
- What do you do when it fails?
- Split recovery lock into separate cluster & recovery locks
- Split out election code

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"
- What do you do when it fails?
- Split recovery lock into separate cluster & recovery locks
- Split out election code
- Drop recovery lock?

What's next?

Split the recovery lock

- Drop support for "`ctdb setreclock ...`"
- What do you do when it fails?
- Split recovery lock into separate cluster & recovery locks
- Split out election code
- Drop recovery lock?
- Depends on handling of election during recovery

What's next?

Split out election handling

What's next?

Split out election handling

- Given work so far, quite easy to factor out

What's next?

Split out election handling

- Given work so far, quite easy to factor out
- Should we then run as a separate daemon?

What's next?

Split out election handling

- Given work so far, quite easy to factor out
- Should we then run as a separate daemon?
- Would this daemon do the recovery master validation?

What's next?

Public IP management/takeover

What's next?

Public IP management/takeover

- Improve API to IP allocation algorithm module?

What's next?

Public IP management/takeover

- Improve API to IP allocation algorithm module?
- IP address reloading helper

What's next?

Public IP management/takeover

- Improve API to IP allocation algorithm module?
- IP address reloading helper
- IP takeover run helper

What's next?

Public IP management/takeover

- Improve API to IP allocation algorithm module?
- IP address reloading helper
- IP takeover run helper
- Move public IP state into a replicated database?

What's next?

Public IP management/takeover

- Improve API to IP allocation algorithm module?
- IP address reloading helper
- IP takeover run helper
- Move public IP state into a replicated database?
- Move TCP connection tracking ("tickles") into a replicated database?

Legal Statement

- This work represents the view of the authors and does not necessarily represent the view of IBM.
- IBM is a registered trademark of International Business Machines Corporation in the United States and/or other countries.
- Linux is a registered trademark of Linus Torvalds.
- Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Other company, product, and service names may be trademarks or service marks of others.

Questions?