Dovecot 3.0 - Das Ende der Community?

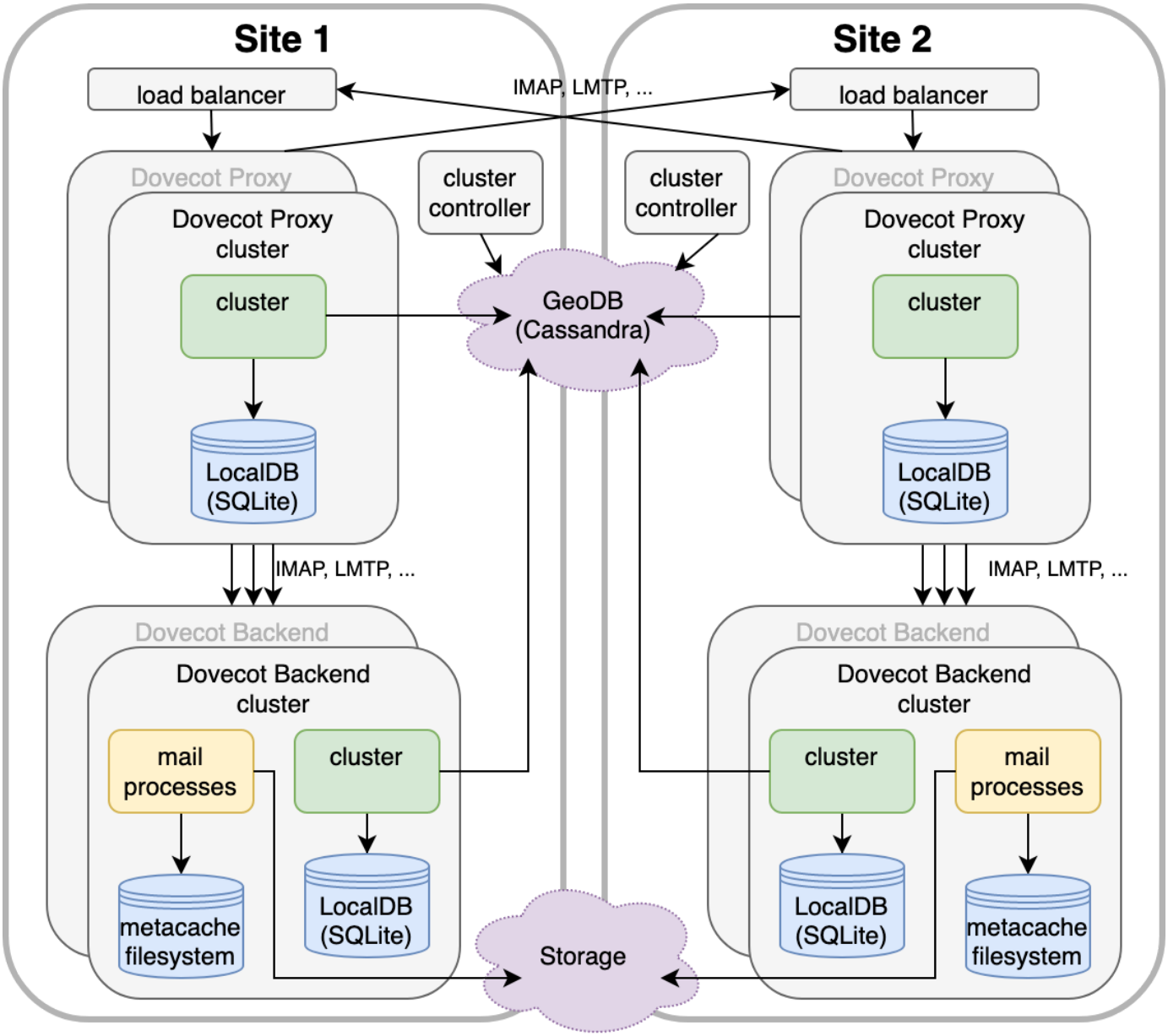
- Fazit

Dovecot 2.4 CE und 3.0 Pro

- Nach über 2 Jahren Entwicklung erwarten wir bald das neue Major Release von Dovecot
- *Community Edition* und *Professional* werden sich ab dem Release in ihren Versionsnummern unterscheiden
- scheinbar dieses Mal kein frühes Release der Open-Source Variante
- Dovecot Cluster in der 3.0 Pro ist die sichtbarste Änderung

---

Dovecot 3.0 (Pro) Cluster

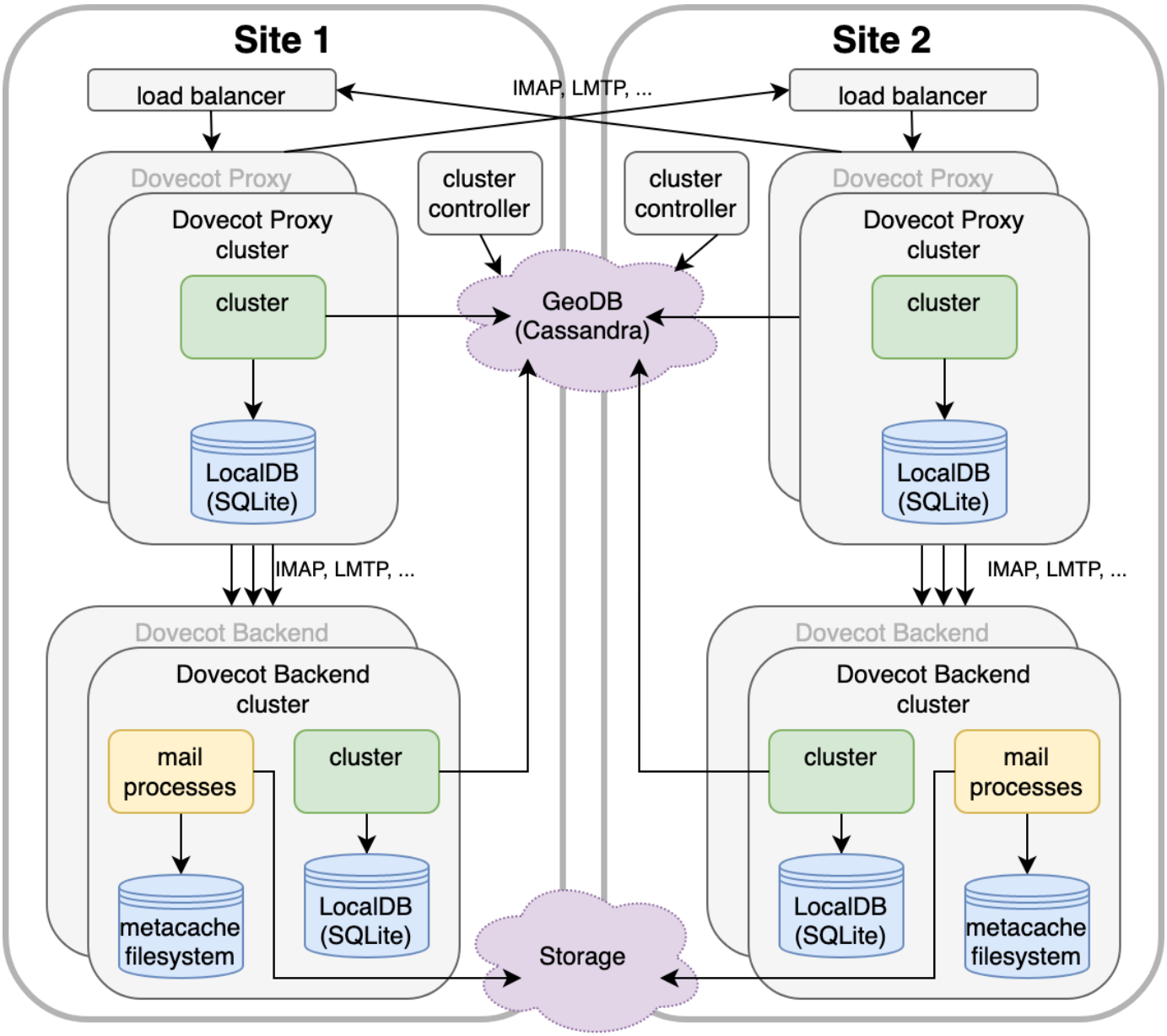


Dovecot 3.0 (Pro) Cluster Konzept

- der Dovecot Cluster ist (auch) ein Ersatz für den Director
  - Director: naiver und oft halb-uninformierter Verbindungsverteiler
  - Cluster: dedizierte Steuerung der Verbindung auf Grund verfügbarer Status Daten
- Funktionalität ggü. Director wurde dabei stark erweitert
- Cluster Controller ist die übergeordnete Instanz zur Steuerung
- zentrale Status-Datenbank "GeoDB"
- locale SQLite Caches
- auch die Backends sind im Cluster eingebunden
- typisches Konzept für Container/Kubernets Infrastrukturen mit einer statischen Single-Point-Of-Truth Instanz

---

Dovecot 3.0 (Pro) Cluster



Dovecot 2.4 CE - new Features

(soweit jetzt bekannt)

- Flatcurve FTS - bisher eigenständiges Projekt
- alle `doveadm` Kommandos bekommen einen expliziten `-u` Parameter
- *mail-crypt* bekommt Support für Eliptic Curve Algorithmen
- OpenSSL 3 Support in den Paketen von Dovecot

Removed Features in Dovecot 2.4/3.0

- Dovecot Director
- Replication
- Single Instance Storage: *fs-sis*
- Alte Volltextsuchen: *fts-lucene, fts-squat*
- Authentifizierung: *TCP wrapper support, checkpassword, shadow*
- XZ Kompressions-Verfahren
  - Achtung: muss migriert werden
- Global ACL Files - sind jetzt Global ACL Directory
- schwache Passwort Schemes werden deaktiviert

Kein Open-Source High-Available Dovecot mehr?

- Director und Replication sind unsere derzeit wichtigsten HA-Komponenten im Dovecot
- Die Idee des Directors läßt sich relativ einfach nachbauen
  - Aufwand sollte aber nicht unterschätzt werden
- Replication ist in 2.3 weiter Open-Source
  - sollte auch in 2.4 noch funktionieren, aber wurde aus dem Code entfernt
  - externes Plugin denkbar
- Vielleicht wird es aber auch Zeit für neue Ideen beim Backend Storage

---

Dovecot Director im Detail

- Ist ein Layer 7 Proxy
  - Kennt also im Gegensatz zu einem Layer 4 Proxy (IP) den User
- Kann alle Dovecot Protokolle sprechen
  - IMAP, POP3, SMTP, Sieve, Doveadm
- Kennt die verfügbaren Backends
  - Das Backend muss kein Dovecot sein
  - Kann die User auf die Backends verteilen
- Könnte den User auch schon authentifizieren
- mehrere Direktoren tauschen Zustandsinformationen über ein Ring Protokoll aus
- User und Backends sind über `doveadm` Kommandos managebar

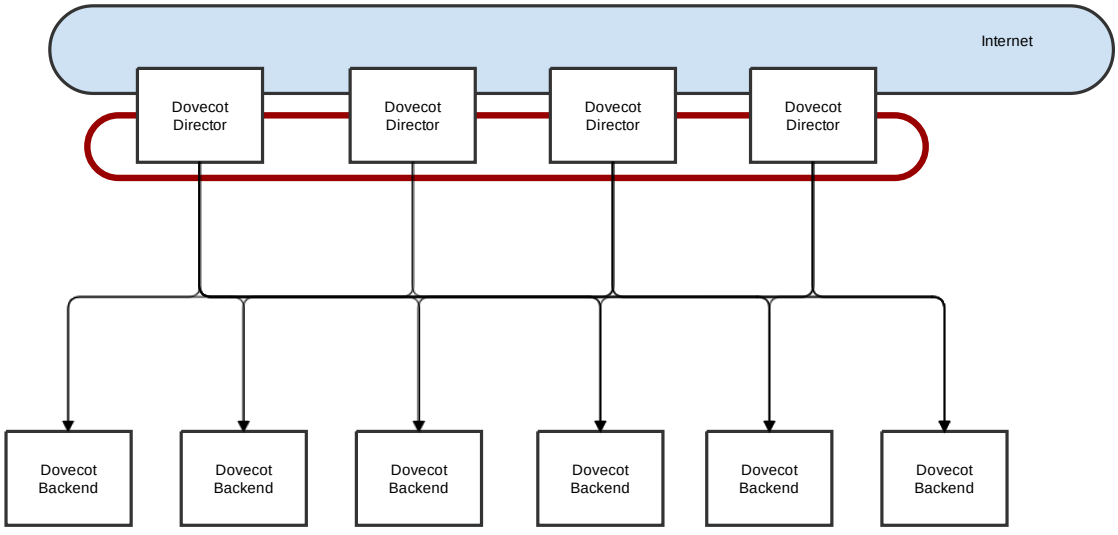
Dovecot Director im Detail

- Genaugenommen reden wir von 3 Services: Auth + Proxy + Director
- Auth identifiziert den User und stellt die User-Informationen bereit
- Proxy kann Verbindungen weiterreichen
  - Proxy Komponente weiterhin in Dovecot 2.4/3.0 erhalten
- Director kümmert sich um die Zuordnung der User zu einem Backend

Verbindungsablauf mit einem Director

- Auth Daemon fragt den User in der Datenbank ab und übergibt die User-Daten an den Director
- Der Director sucht nach einem gesetzten *director_tag* für die Auswahl der richtigen Backend Gruppe
- Aus der Backend-Gruppe wählt der Director das empfehlenswerteste Backend aus
- Diese Zuordnung wird in der Session gespeichert und über das Ring-Protokoll mit den anderen Direktoren ausgetauscht
- Proxy bekommt die User Daten mit den vom Director gesetzten Host

Director Ring



Dovecot Director und seine Problemchen

- Kein Monitoring, ob das ausgewählte Backend verfügbar oder überlastet ist
  - externe Tools wie Dovemon bzw Poolmon steuern den Director von außen
- Der Director lädt die Config beim Start von der Platte, Änderungen werden aber im Speicher gehalten -> Potenziell Split-Brain
- Es sollten Laufzeitdaten (Memory) und Config (File) immer konsistent sein
  - Live-Änderungen in der Infrastruktur sind also Config Anpassung + *Doveadm* Laufzeitanpassung

Verbindungsablauf mit einem Dovecot Proxy ohne Director

- Auth Daemon fragt den User in der Datenbank ab und übergibt die User-Daten an den Director
- in den User-Daten muss das Attribut *Host* vorhanden sein
- Proxy bekommt die User Daten mit den vom Auth Daemon gesetzten Host

- *Wenn wir also Host dynamisch setzen können, haben wir die gleiche Funktionalität*

Ersatz für den Dovecot Director

- Auswürfeln: `echo 10.0.0.$((1 + $RANDOM % 2))` ☺
- Über die DB-Abfrage lösen (Stored Procedures)
- Dovecot hat einen simplen Ansatz mit LUA dokumentiert
  - Director with Lua
  - Tools für das Management werden extra benötigt
- Unsere Idee: HTTP-API als Controller - Lua-Passdb im Dovecot

---

```
-- Copyright 2023 Open-Xchange Software Gmbh
...
local DBI = require 'DBI'
...

function auth_passdb_init(args)
  local err
  -- This needs to be changed
  dbd, err = DBI.Connect('MySQL', 'dovecot', 'dovecot', 'dovecot')
  assert(dbd, err)
  dbd:autocommit(true)

  lookup_stmt, err = dbd:prepare('SELECT backends.id AS bid FROM backends JOIN'..
    ' user_backend ON backends.id = user_backend.backend_id WHERE'..
    ' user_backend.user = ?')
  assert(lookup_stmt, err)
  update_stmt, err = dbd:prepare('INSERT INTO user_backend (backend_id, user)'..
    ' VALUES(?, ?)')
  assert(update_stmt, err)
  lookup_backend_stmt, err = dbd:prepare('SELECT * FROM backends WHERE id = ?')
  assert(lookup_backend_stmt, err)
  lookup_backends_stmt, err = dbd:prepare('SELECT backends.id FROM backends WHERE state = 0')
  assert(lookup_backends_stmt, err)

  extra_attributes = args
end

function auth_passdb_lookup(req)
  -- perform lookup
  local success, err = lookup_stmt:execute(req.user)
  ...
end

...
```

Replication

Dsync im Detail

- Der Abgleich von 2 Mailboxen erfolgt bei Dovecot über das dsync Protokoll
- die Funktionen kommen bei der Replication als auch bei doveadm sync/backup/import zum Einsatz
- der Sync erfolgt auf der einen Seite immer mit einem lokalen Dovecot User und auf der anderen Seite mit Mailbox Daten in einem lokalen Verzeichnis, einem Doveadm Server im Netzwerk oder auch via imapc mit einem Remote IMAP Server
  - bei einem lokalen Verzeichnis müssen die Mailbox Daten in einem Dovecot kompatiblem Format vorliegen
- Dovecot kann mit dsync Mailboxen, Subscriptions, ACLs (Berechtigungen/Shares) und Sieve Skripte synchronisieren

---

Replication

Dsync im Detail

- Es werden nur die Änderungen abgeglichen - es erfolgt keine Kopie *des ganzen Ordners*
- Die Replication kennt *fast-sync, full-sync* und *stateful*
- *full-sync* vergleicht alle Mails aller Mailboxen anhand ihrer IDs und ein paar Headern
- *fast-sync* vergleicht nur uidvalidity, uid-next und highest-modseq auf beiden Seiten
  - dsync geht dabei davon aus, dass beide Seiten prinzipiell *in sync* sind und nur eventuell neue Daten dazu gekommen sind oder sich Flags geändert haben
- *stateful* merkt sich den letzten Stand des Sync und überträgt nur folgenden Änderungen

---

Replication

Dsync im Detail

- Kollisionen (neue Mails auf beiden Seiten) können oft gut aufgelöst werden
- Gelöschte Objekte sind über Index Einträge eindeutig markiert und nicht einfach verschwunden
- in seltenen Situationen muss eine Kollision manuell gelöst werden
- Dsync ist erst einmal nur die Möglichkeit des Abgleichs 2er Mailboxen
- Die Replication ist ein Mechanismus dieses zu automatisieren

---

Replication

replicator und aggregator

- Replication ist quasi automatischer dsync mit einem remote Doveadm Server
- Dafür gibt es zusätzlich die Services *aggregator* und *replicator*
- *aggregator*: FIFO in der von Diensten wie imap/lmtp/sieve Änderungen an den Daten angezeigt werden
- *replicator*: Kümmert sich bei neuen Einträgen beim *aggregator* um die Ausführung von dsync
- normalerweise wird ein *stateful* sync gemacht
- im Fehlerfall ein *full-sync*
- ein *full-sync* aller user wird darüber hinaus regelmäßig durch Dovecot angestoßen. Ausschlaggebend ist hier `replication_full_sync_interval`

Probleme mit der Replication

- Dsync benötigt für den Sync Zugriff auf die Dovecot Index Dateien im User Account
  - Dieser kann bei hoher Last oder bei dauerhaften Zugriff durch andere Dienste (imap/lmtp) zu lange dauern
  - Defekte Index Dateien lassen den Sync immer wieder abbrechen
  - In einigen Dovecot Versionen wurden bestimmte Daten nicht sauber synchronisiert
- Vorteil der doppelten Datenhaltung ist auch der Nachteil: doppelter Storage-Verbrauch

Wegfall der Replication

- Daher ist aus Sicht von Dovecot verständlich, dass die Replication entfernt wird - wenn keine Ressourcen investiert werden sollen, die die bekannten Probleme grundsätzlich lösen
- Dovecot hat die Replication auch nie offiziell für seine **Pro** Version unterstützt
- manuelles Dsync über *doveadm* bleibt für Migrationen usw. aber erhalten
- doveadm import/backup bleiben natürlich auch erhalten

Ersatz für die Replication

- *active/passiv* regelmäßiger dsync mittels Cron oder ähnlichem Trigger
- Der Dovecot Code des Replicator ist und bleibt Open-Source und könnte als externes Plugin gepflegt werden
- Dovecot selbst empfielt für kleinere Projekt NFS und für größere S3 via Obox (was dann aber auch eine **Pro** Lizenz erfordert)
- Das Telekom Dovecot-Ceph Plugin wird immer noch weiter entwickelt.

Dovecot und NFS

- Dovecot bietet einige Optionen um den Betrieb von Mailboxen auf NFS Storage abzusichern
- Aktive Dovecot Sessions benötigen aber auch exklusiven Zugriff auf die Index Dateien eines User Accounts
  - keine konkurierenden Zugriffe verschiedener Dovecots
  - sonst gibt es oft defekte Index Files
  - bei *Maildir*
    - einfache Index-Wiederherstellung aus den Dateinamen
    - aber Performance-Probleme bei großen Speicherbedarf oder langsamen Storage
  - bei *mdbox*
    - saubere Index-Files werden zwingend benötigt
    - bei Wiederherstellung gehen alle Meta-Daten verloren
      - "Ordner" (Mailbox-Zuordnung)
      - Flags wie SEEN, Important etc.
      - ohne regelmäßiges `doveadm purge` können sogar gelöschte Mails wieder auftauchen
- Also: Exklusive Zuordnung des Users auf ein Backend muss stabil sein und muss auch sauber umgezogen werden

---

Also? Das Ende der Dovecot Community?

- Dovecot ist und bleibt Open-Source
- Aber Dinge, die nicht mehr im Scope von Dovecot Pro liegen, werden nicht weiter geführt und aus den offiziellen Dovecot CE Sourcen entfernt
- Focus scheint klar auf Dovecot Pro und sehr großen Infrastrukturen zu liegen
- https://dovecot.org/mailman3/hyperkitty/list/dovecot@dovecot.org/message/5LYQGL2ZOMDDUFUOJU2TUBLLQUWEPZSR/

- > Michael Slusarz: "To focus development efforts, and to provide extreme clarity for users going forward, Dovecot CE for the first time has adopted a defined Vision Statement: "To provide the world's premier open source, standards compliant, full-featured, **single node** email backend server."
- > Michael Slusarz: "A reminder that Dovecot is commercial software, and has been since Timo made this decision 13 years ago"
- Timo Sirainnen @SLAC 2014: https://www.heinlein-support.de/sites/default/files/dovecot_recent_and_future_development.pdf

---

Fazit

- die Dovecot 2.3 ist in aktuellen LTS Versionen enthalten und damit auch noch 5 Jahre von den Distributionen supported
- Wenn die 2.4 released ist können wir uns in Ruhe ansehen wie wir Director und Replication adäquat kompensieren
- Der Dovecot Cluster (Pro) ist ein komplexes System, scheint für große Infrastrukturen aber ein guter Ansatz für ein Director-Replacement zu sein
- Unser folgender Vortrag könnte interessant sein ☺
