

Dot3 : Cartographier une Topologie MariaDB / MySQL Complexe

Aurélien LEQUOY · 21 mars 2026

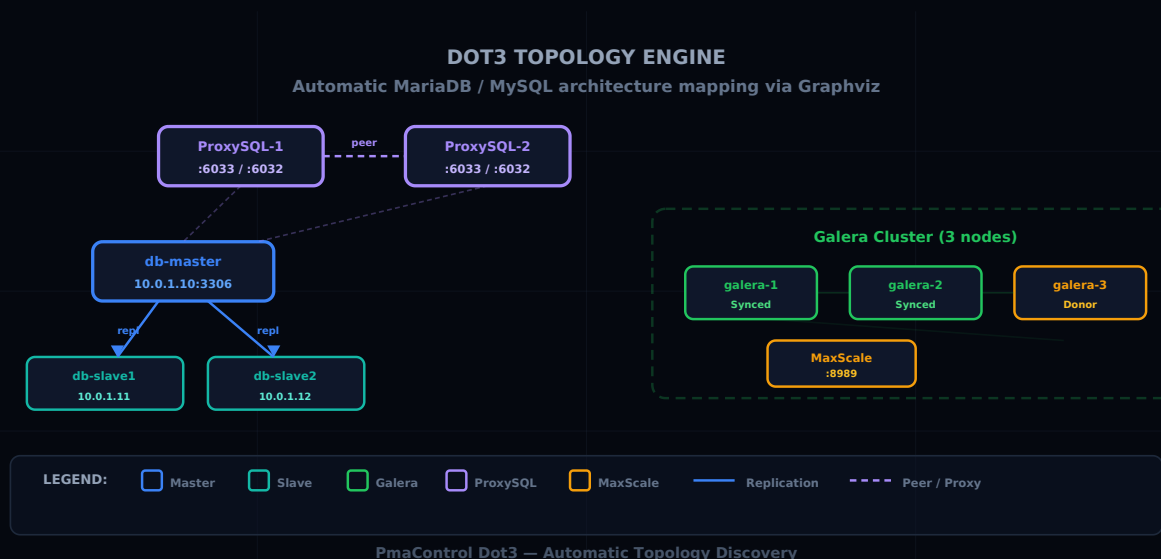
PMACONTROL

TOPOLOGY

GRAPHVIZ

PROXYSQL

ARCHITECTURE



Le problème : des topologies invisibles

Vous avez 40 instances MariaDB / MySQL. Certaines sont en réplication Master-Slave. D'autres en Master-Master. Un cluster Galera de 5 noeuds. Deux ProxySQL en peer-to-peer. Un MaxScale pour le routage lecture. Des tunnels SSH pour les accès inter-DC. Des alias DNS qui masquent les IP réelles.

Question simple : **pouvez-vous dessiner l'architecture complète en moins de 5 minutes ?**

Dans la plupart des organisations, la réponse est non. La topologie vit dans un wiki obsolète, un Confluence jamais mis à jour, ou dans la tête du DBA senior qui est en congé.

Dot3 résout ce problème.

Qu'est-ce que Dot3 ?

Dot3 est le moteur de cartographie topologique intégré à PmaControl. Il utilise **Graphviz** (le langage DOT) pour générer des représentations visuelles des architectures de bases de données.

Le nom "Dot3" vient de la troisième itération du moteur. Les deux premières versions utilisaient des approches simplistes (listes de connexions, arbres statiques). Dot3 introduit la **résolution dynamique des endpoints** et le **rendu contextuel**.

Résolution des endpoints

La première étape de Dot3 est de comprendre ce qui se cache derrière chaque endpoint configuré dans PmaControl. Un endpoint peut être :

1. Connexion directe MariaDB / MySQL

Le cas le plus simple. Dot3 se connecte au port 3306 (ou autre), exécute `SHOW VARIABLES` et `SHOW SLAVE STATUS` / `SHOW REPLICA STATUS`, et identifie le rôle du serveur (master, slave, standalone).

2. ProxySQL

Dot3 détecte ProxySQL par la présence du **port admin** (6032 par défaut). En se connectant au port admin, il récupère :

- `runtime_mysql_servers` : liste des backends avec leur hostgroup
- `runtime_mysql_replication_hostgroups` : mapping writer/reader
- `proxysql_servers` : les peers ProxySQL (pour le clustering)

Cela permet de tracer les liens entre le ProxySQL et ses backends, et entre les ProxySQL peers.

3. MaxScale

MaxScale est détecté via son API REST (port 8989). Dot3 interroge `/v1/servers` et `/v1/services` pour reconstruire la topologie des backends et des services de routage.

4. VIP et adresses flottantes

Quand un endpoint est une VIP (Virtual IP gérée par keepalived ou Pacemaker), Dot3 résout l'adresse vers le serveur physique actif. La VIP est représentée comme un noeud intermédiaire avec un style visuel distinct (losange au lieu de rectangle).

5. Tunnels SSH

Pour les architectures multi-DC avec des tunnels SSH, Dot3 trace le chemin complet : client -> tunnel SSH -> endpoint réel. Le tunnel apparaît comme une connexion pointillée dans le graphe.

6. Alias DNS

Dot3 résout les alias DNS (CNAME) et affiche à la fois le nom d'alias et le hostname réel, évitant la confusion quand un même serveur est référencé par plusieurs noms.

Le rendu

Une fois tous les endpoints résolus, Dot3 génère un graphe DOT qui est compilé par Graphviz en SVG (ou PNG). Voici comment les différentes topologies sont représentées.

Réplication Master-Slave

```
Master [shape=box, color=blue, label="db-master\n10.0.1.10:3306"]
Slave1 [shape=box, color=green, label="db-slave1\n10.0.1.11:3306"]
Slave2 [shape=box, color=green, label="db-slave2\n10.0.1.12:3306"]
Master -> Slave1 [label="repl"]
Master -> Slave2 [label="repl"]
```

Les flèches de réplication vont du master vers les slaves (sens des données). Le lag de réplication est affiché sur l'arête quand il est significatif.

Réplication Master-Master

Les réplications bidirectionnelles sont représentées avec des **flèches doubles**. Dot3 détecte ce cas quand les deux serveurs s'identifient mutuellement comme master via `SHOW SLAVE STATUS`.

```
Master1 -> Master2 [dir=both, label="M-M"]
```

Cluster Galera

Les noeuds Galera sont regroupés dans un **subgraph** avec un cadre en pointillés. Chaque noeud est coloré selon son état wsrep :

- **Synced** : vert
- **Donor/Desynced** : orange
- **Joining** : jaune
- **Disconnected** : rouge

```
subgraph cluster_galera_prod {
  label="Galera Prod (3 nodes)"
  style=dashed
  galera1 [color=green]
  galera2 [color=green]
  galera3 [color=green]
}
```

ProxySQL peer-to-peer

Les liens entre ProxySQL peers sont tracés avec des **flèches bidirectionnelles pointillées**, distinguées visuellement des liens de réplication MariaDB / MySQL.

Le bug des icônes PNG dans SVG

Lors du développement de Dot3, nous avons voulu intégrer des icônes pour identifier visuellement les types de noeuds : une icône MariaDB, une icône ProxySQL, une icône MaxScale.

La première implémentation utilisait des **icônes PNG embedées dans le SVG** via des balises `<image>` avec un `href` en base64. Cela fonctionnait dans les navigateurs mais posait plusieurs problèmes :

1. **Graphviz ne supporte pas correctement les PNG embedés** dans sa sortie SVG — les positions étaient décalées
2. **Le rendu était flou** sur les écrans Retina/HiDPI
3. **La taille des fichiers SVG explosait** à cause du base64

La solution : remplacer toutes les icônes PNG par des **icônes SVG natives**. Nous avons recréé `proxysql.svg`, `maxscale.svg` et les autres en SVG vectoriel pur. Le résultat :

- Rendu net à toute résolution
- Positionnement correct dans Graphviz
- Taille des fichiers divisée par 5

Exemple concret

Voici une topologie réelle (anonymisée) générée par Dot3 pour un client e-commerce :



Ce schéma est généré **automatiquement** par Dot3. Aucune configuration manuelle du layout. Dot3 découvre la topologie, résout les endpoints, et Graphviz s'occupe du placement.

Intégration dans PmaControl

Dans l'interface PmaControl, la carte topologique est accessible via l'onglet **Topology**. Elle se rafraîchit automatiquement et reflète l'état en temps réel :

- Un slave en retard de réplication apparaît avec une arête rouge
- Un noeud Galera en état `Donor` change de couleur
- Un ProxySQL qui perd un backend le montre en grisé

Le SVG généré est interactif : cliquer sur un noeud ouvre la fiche détaillée du serveur dans PmaControl.

Limites et évolutions prévues

Dot3 fonctionne bien pour des topologies de 1 à 100 noeuds. Au-delà, le layout Graphviz peut devenir confus. Les évolutions prévues :

1. **Filtrage par tag** : n'afficher qu'un sous-ensemble de la topologie (par environnement, par client, par DC)
2. **Mode diff** : comparer deux snapshots topologiques pour visualiser les changements

3. **Export Terraform** : générer une description Infrastructure-as-Code de la topologie découverte

Conclusion

La topologie d'une infrastructure MariaDB / MySQL n'a pas à être un mystère documenté dans un wiki périmé. Dot3 la découvre, la résout et la dessine automatiquement.

C'est la différence entre "je crois que la réplication va vers ce serveur" et "je vois que la réplication va vers ce serveur, avec 3 secondes de lag, depuis 14 jours".