

temBoard



Pierre Giraud

Meetup - Genève - 23 Mai 2019

temBoard

TITRE : temBoard

SOUS-TITRE :

DATE: Pierre Giraud



ARCHITECTURE

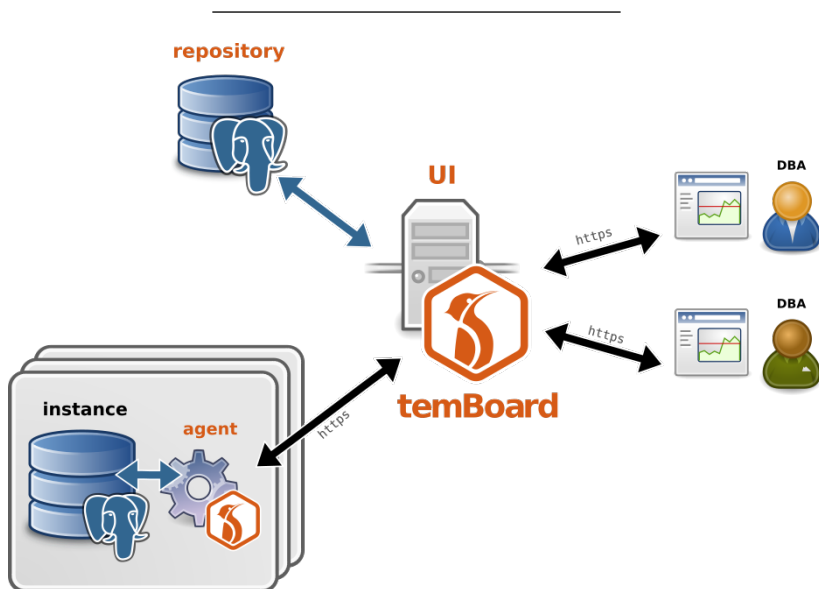


Figure 1: temBoard architecture

AGENT

- Mono-instance
- Pas de dépendances
- API REST
- Authentification
- Packagé pour centos/RHEL 6/7 et Debian

TemBoard agent est installé à côté de l'instance PostgreSQL.

Développé en python compatible 2 et 3.

1 instance -> 1 agent

L'agent récupère des métriques à intervalle régulier pour les envoyer au serveur (UI).

Il peut aussi recevoir des requêtes envoyées par le serveur pour l'affichage en temps réel.

Échange de données entre l'agent et le serveur via authentification dédiée et via protocole *HTTPS*.

SERVEUR

(aussi appelé UI)

- Interface Web
 - Python 2.7 / Tornado / SQLAlchemy
- Base de données historique et metadonnées
 - PostgreSQL 9.4+
- Authentification
- Packagé pour CentOS/RHEL 7 et Debian
- Un serveur pour un parc d'agents / instances

L'interface utilisateur de temboard est développée en python 2.7 et repose sur le framework web Tornado.

Une base de données, appelée *repository* est nécessaire à son fonctionnement, en effet, celle-ci va permettre de stocker :

- la liste des comptes utilisateurs habilités à se connecter à l'interface;
- la liste des instances Postgres à gérer;
- l'historique des données collectées.

L'accès à cette interface est protégée par une authentification utilisateur.

FONCTIONNALITÉS

PLUGINS

- Fonctionnement par plugins
 - Activation / désactivation
-

- Tableau de bord
- Configuration Postgres
- Supervision
- Activité
- Maintenance

TABLEAU DE BORD

(*Plugin Dashboard*)

- Affichage en temps réel
 - Métriques système : usage CPU, mémoire, *loadaverage*.
 - Métriques Postgres : Cache Hit Ratio, Sessions, TPS.
 - Statut de chaque métrique calculé selon des seuils (*alerting*).
-

CONFIGURATION

(*Plugin pg_conf*)

- Configuration des paramètres de l'instance (`postgresql.conf`)

Configuration de `pg_hba.conf` et `pg_ident.conf` supprimée car cela va au-delà du périmètre de temBoard.

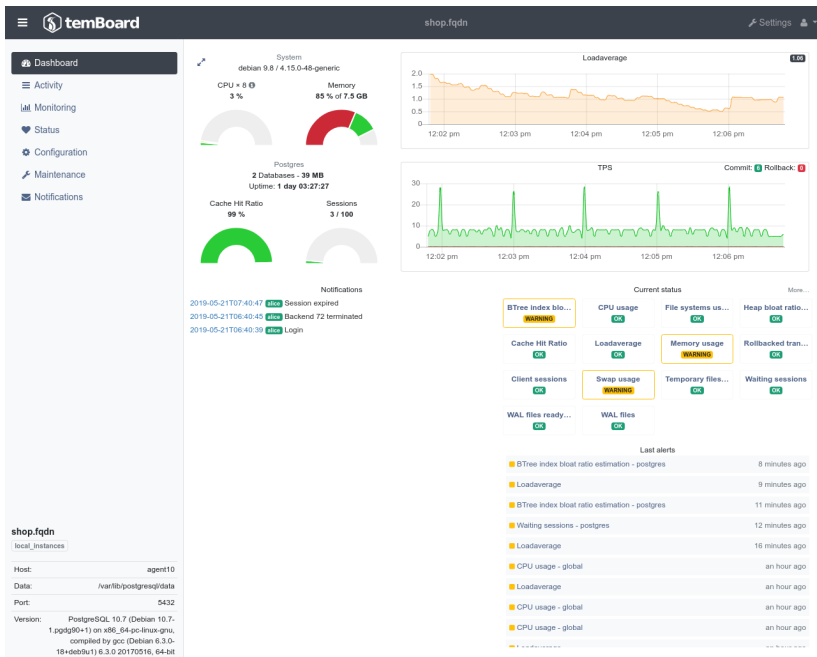


Figure 2: temBoard Dashboard

The screenshot shows the temBoard interface with the 'Configuration' section selected in the left sidebar. The main content area displays the 'Autovacuum' configuration page. At the top, there is a search bar and a category dropdown set to 'Autovacuum'. The configuration is organized into a table with columns for the parameter name, its description, and its current value. A green 'On' button is visible at the top right of the configuration area. At the bottom, there is a 'Save and reload configuration' button.

Parameter	Description	Value
autovacuum	Starts the autovacuum subprocess.	On
autovacuum_analyze_scale_factor	Number of tuple inserts, updates, or deletes prior to analyze as a fraction of relpages.	0.1
autovacuum_analyze_threshold	Minimum number of tuple inserts, updates, or deletes prior to analyze.	50
autovacuum_freeze_max_age	Age at which to autovacuum a table to prevent transaction ID wraparound.	200000000
autovacuum_max_workers	Sets the maximum number of simultaneously running autovacuum worker processes.	3
autovacuum_multixact_freeze_max_age	Multixact age at which to autovacuum a table to prevent multixact wraparound.	400000000
autovacuum_naptime	Time to sleep between autovacuum runs.	1min
autovacuum_vacuum_cost_delay	Vacuum cost delay in milliseconds, for autovacuum.	20ms
autovacuum_vacuum_cost_limit	Vacuum cost amount available before reaping, for autovacuum.	-1
autovacuum_vacuum_scale_factor	Number of tuple updates or deletes prior to vacuum as a fraction of relpages.	0.2
autovacuum_vacuum_threshold	Minimum number of tuple updates or deletes prior to vacuum.	50

Figure 3: temBoard Configuration

SUPERVISION

(Plugin Monitoring)

- Collecte périodique de métriques par l'agent (système et PostgreSQL)
- Envoi à l'agent (push)
- Agrégation et historisation côté serveur
- Graphiques
- Seuils d'alerte

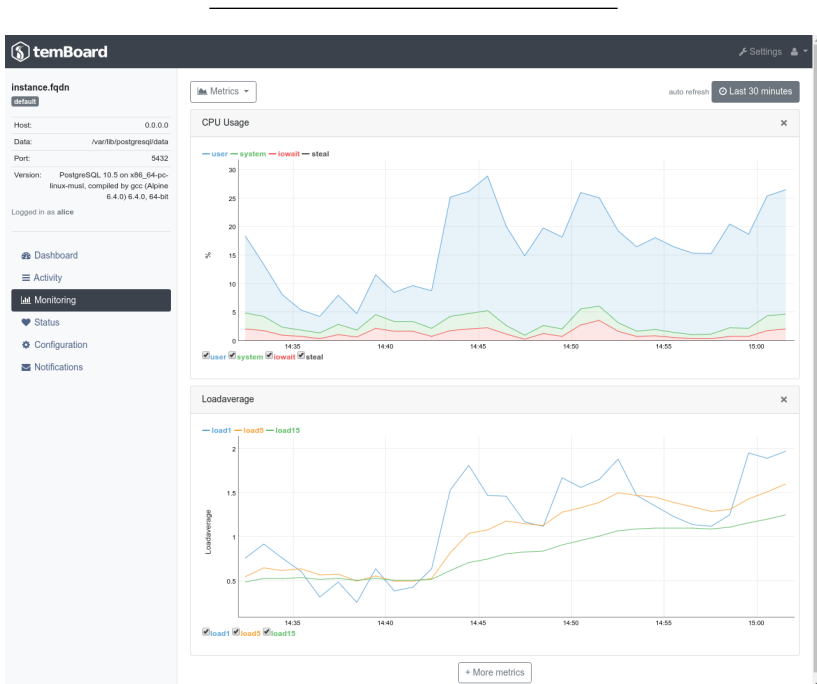


Figure 4: temBoard Monitoring

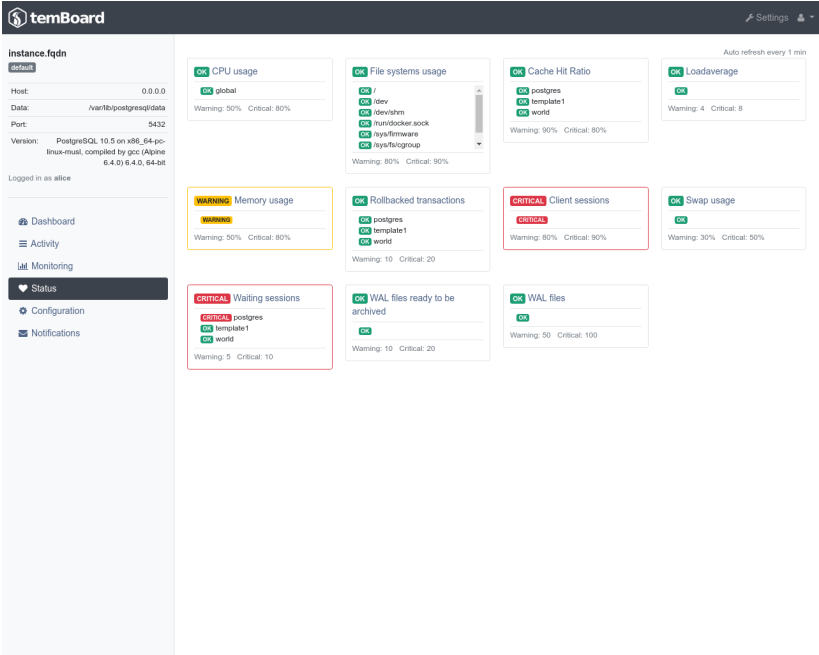


Figure 5: temBoard Alerting

ACTIVITÉ

(Plugin Monitoring)

- (quasi) Temps réel
- Affichage des requêtes en cours d'exécution
 - base, utilisateur
 - CPU, mémoire, I/O
 - durée
- Requêtes bloquées
- Requêtes bloquantes
- Possibilité de terminer les backends

The screenshot shows the 'Activity' page in temBoard. At the top, there are tabs for 'Running', 'Waiting' (2), and 'Blocking' (1). A 'Terminate' button is visible. Below is a table of processes:

PID	Database	User	CPU	mem	Reads	Write/s	IOW	W	State	Time	Query
10789	world	bob	N/A	N/A	N/A	N/A	N/A	N	idle in tra...	62.34 s	UPDATE country SET region='Europe' WHERE code='MLD';
10772	world	alice	N/A	N/A	N/A	N/A	N/A	N/A	active	59.35 s	DELETE FROM country WHERE code='MLD';
10917	postgres	postgres	N/A	N/A	N/A	N/A	N/A	Y	active	0.01 s	UPDATE pgbench_branches SET tbalance = tbalance + -4346 WHERE tid...
10915	postgres	postgres	N/A	N/A	N/A	N/A	N/A	Y	active	0 s	UPDATE pgbench_branches SET tbalance = tbalance + -1823 WHERE tid...
10914	postgres	postgres	N/A	N/A	N/A	N/A	N/A	Y	active	0 s	UPDATE pgbench_tellers SET tbalance = tbalance + 3456 WHERE tid...
10918	postgres	postgres	N/A	N/A	N/A	N/A	N/A	N	active	0 s	END;
10916	postgres	postgres	N/A	N/A	N/A	N/A	N/A	Y	active	0 s	UPDATE pgbench_tellers SET tbalance = tbalance + 2426 WHERE tid...

Below the table, there is a section for 'shop.fqdn' with details like 'local_instances', 'Host: agent10', 'Data: /var/lib/postgresql/data', 'Port: 5432', and 'Version: PostgreSQL 10.7 (Debian 10.7-1.pgd90+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 6.3.0-18+deb8u1) 6.3.0 20170516, 64-bit'.

Figure 6: temBoard Activity

MAINTENANCE

(Plugin Maintenance)

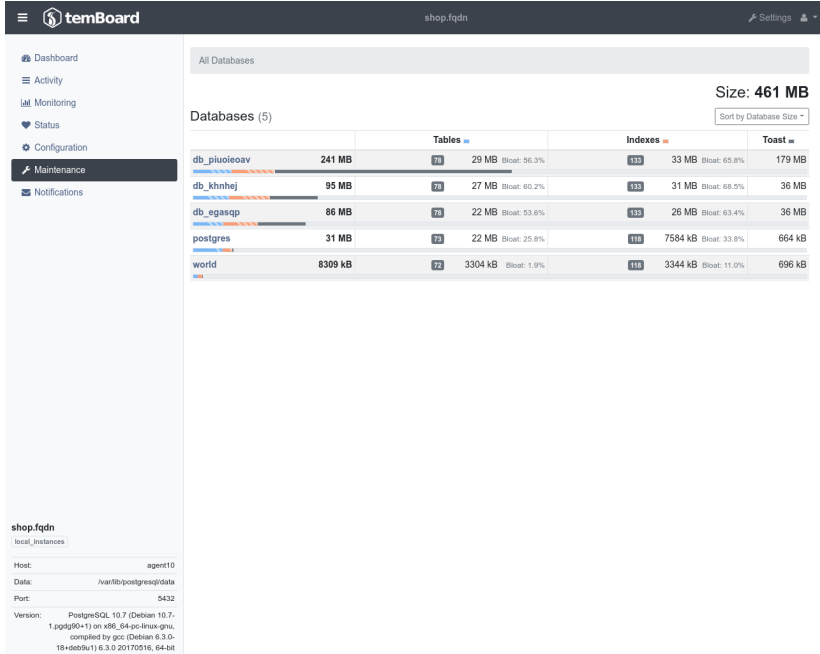


Figure 7: temBoard Maintenance

temBoard shop.fqdn

All Databases / Database: db_piroleoav / Schema: lbhzbpawzpf / Table: table_with_bloat

Table:table_with_bloat

Total	Heap	Indexes	Toast	Fill Factor
7424 kB	3472 kB	3944 kB	8192 bytes	100%
~ 10001 ROWS (~ 0 dead)	Bloat: 54% (2928 kB)	Bloat: 85% (3480 kB)		

Last ANALYZE: 2 minutes ago
1 analyzers - 0 auto analyzers

Last VACUUM: 2 minutes ago (auto)
0 vacuums - 1 auto vacuums

⚠ Overall table bloat is high. You should consider running a Full VACUUM.
Table bloat wastes space and slows down queries.

⚠ Overall index bloat is high. You should consider running a Full VACUUM or REINDEX.
Index bloat wastes space and slows down queries.

Indexes (6)

Index Name	Size	Bloat	Action	Reindex
table_with_bloat_id_key (btree)	1984 kB	88.3%	CREATE UNIQUE INDEX table_with_bloat_id_key ON lbhzbpawzpf.tab...	Reindex
table_with_bloat_id_key (btree)	1984 kB	88.3%	CREATE UNIQUE INDEX table_with_bloat_id_key ON lbhzbpawzpf.tab...	Reindex
table_with_bloat_id_key (btree)	1984 kB	88.3%	CREATE UNIQUE INDEX table_with_bloat_id_key ON lbhzbpawzpf.tab...	Reindex
table_with_bloat_cat_idx (btree)	1968 kB	87.8%	CREATE INDEX table_with_bloat_cat_idx ON lbhzbpawzpf.table_wit...	Reindex
table_with_bloat_cat_idx (btree)	1960 kB	88.2%	CREATE INDEX table_with_bloat_cat_idx ON lbhzbpawzpf.table_wit...	Reindex
table_with_bloat_cat_idx (btree)	1944 kB	88.9%	CREATE INDEX table_with_bloat_cat_idx ON lbhzbpawzpf.table_wit...	Reindex

shop.fqdn

local_instances

Host: agent10

Date: /var/lib/postgresql/data

Port: 5432

Version: PostgreSQL 10.7 (Debian 10.7-1.pgd910-1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 6.3.0-18-raspbian) 6.3.0 20170316, 64-bit

Figure 8: temBoard Maintenance

FONCTIONNALITÉS À VENIR

MODE PULL (VS. PUSH)

REQUÊTES LENTES

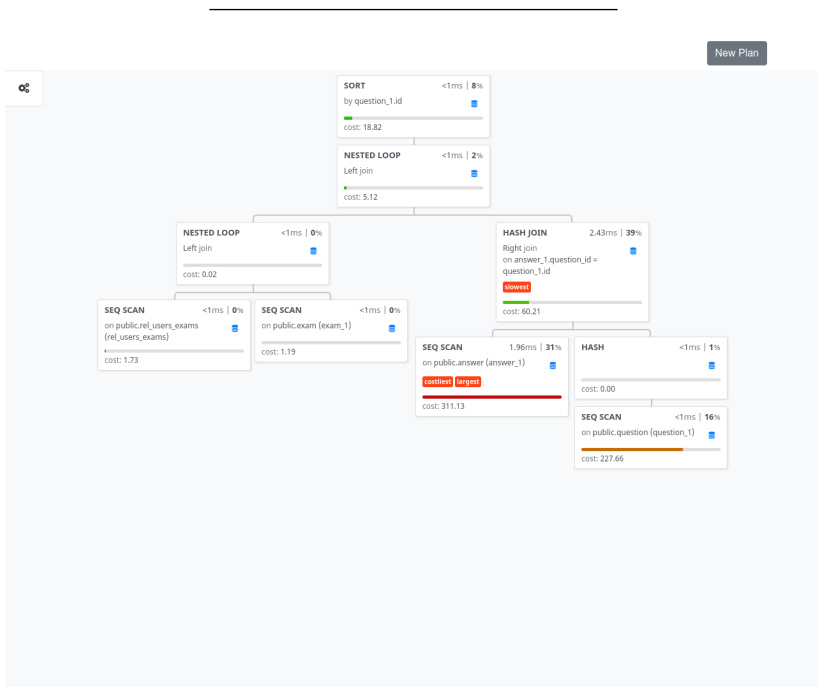


Figure 9: Explain

MERCI

QUESTIONS / REMARQUES ?

Your browser does not support the video tag.

LIENS

- [Code](#)
- [Documentation](#)
- [Quickstart docker](#)