

# Monitorare la salute di PostgreSQL

Marco Nenciarini

[marco.nenciarini@2ndQuadrant.it](mailto:marco.nenciarini@2ndQuadrant.it)

<http://www.2ndQuadrant.it>

@mnencia



# Marco Nenciarini

---

- DBA, sviluppatore e sysadmin presso 2ndQuadrant
  - Database OLTP business critical
  - Data warehousing
- Membro della comunità di PostgreSQL
  - Co-Fondatore di ITPUG
- Debian Developer



# Monitoraggio: perché?

---

- Mantenere le applicazioni efficienti
- Ottimizzare l'utilizzo delle risorse di sistema
- Mantenere PostgreSQL in salute
- Garantire continuità operativa

# Cosa monitorare

---

- Analisi dei file di log
- Statistiche interne del database
- Utilizzo delle risorse di sistema
- Stato dei servizi (allarmi)

# Analisi dei log

---

- Punto di vista dello sviluppatore
  - Errori applicativi
  - Prestazioni delle query
- Punto di vista del sistemista
  - Errori di configurazione
  - Esaurimento delle risorse

# Configurazione dei log

---

- impostazioni di default
  - solo errori
  - no timestamp
- modificare il postgresql.conf
  - parametri log\_\*

# Strumenti di analisi dei log

---

- grep
  - non sottovalutate la potenza della shell Unix
- pgFouine e cloni
- tail\_n\_mail
- ...

# Estensioni di PostgreSQL

---

- pg\_stat\_statements (9.2)
  - permette di spostare la normalizzazione dei log nel database
  - fornisce dati pre-aggregati
- nessuna GUI al momento
- è il futuro

# Statistiche all'interno del DB

---

- PostgreSQL raccoglie statistiche per praticamente tutto
- Viste pg\_stat\_\*
  - alto livello (tabelle, indici, sequenze)
- Viste pg\_statio\_\*
  - basso livello (letture, scritture, cache)
- Fotografia dello stato attuale del sistema

# Statistiche esterne al DB

---

- CPU

- tipo di utilizzo: user, iowait, idle
- numero di context-switch per secondo

- Memoria

- memoria usata da postgres
- memoria usata per cache

# Statistiche esterne al DB

---

- IO disco
  - letture/scritture
    - operazioni/sec
    - byte/sec
  - attesa media
  - percentuale di utilizzo

# Rendere leggibili i numeri

---

- Difficilmente usabili
- Delta
  - Numero di operazioni al secondo
- Grafici
  - Rendono evidenti i trend
  - Permettono di identificare picchi e cambiamenti repentini

# Strumenti per il monitoraggio

Una carrellata non esaustiva

# check\_postgres

---

- [http://bucardo.org/wiki/Check\\_postgres](http://bucardo.org/wiki/Check_postgres)
  - script perl
  - può controllare moltissime cose
    - 58 azioni: sia check che metriche
  - raccoglie dati ma non storicizza
  - output compatibile con Nagios, Cacti, MRTG ....



# Sysstat

---

- <http://sebastien.godard.pagesperso-orange.fr>
  - Storizza statistiche di sistema (locale)
  - linea di comando
    - iostat, pidstat, ...
    - sar
    - grafici: isag, kSar, sar2rrd ...



# collectd

---

- <http://collectd.org>
  - colleziona statistiche (locali e remote)
  - Scritto in C, veloce, portabile, flessibile
  - Non solo PostgreSQL e Sistema Operativo
    - oltre 90 plugin

# Visualizzare i dati

---

- Graphite: <http://graphite.wikidot.com>
- Cacti: <http://www.cacti.net>
- Munin: <http://munin-monitoring.org>
- Nagios: <http://www.nagios.org>
- Zabbix: <http://www.zabbix.com>
- Zenoss: <http://community.zenoss.org/index.jspa>



# Conclusioni

---

- Il monitoraggio è una attività cruciale
  - log, statistiche, prestazioni
- Monitorate tutto ciò che potete
  - Conoscere la normalità fornisce gli strumenti per identificare ciò che normale non è

# Domande?

---

- [marco.nenciarini@2ndQuadrant.it](mailto:marco.nenciarini@2ndQuadrant.it)
- Twitter: @mnencia
- [www.2ndQuadrant.it](http://www.2ndQuadrant.it)
- [blog.2ndQuadrant.it](http://blog.2ndQuadrant.it)
- [www.pgbarman.org](http://www.pgbarman.org)
- [www.postgresql.org](http://www.postgresql.org)



# Grazie!

Licenza Creative Commons BY-NC-SA 3.0

<http://creativecommons.org/licenses/by-nc-sa/3.0/it/deed.it>

