jason(7) | cloud and database-wrangler - Linux man page

NAME

jason tevnan - cloud and database-wrangler with a penchant for automation and sous vide cooking

SYNOPSIS

jason [--store SQL|NOSQL] [--administrate] [--automate] [--monitor] [--virtualize] [--containerize] [--script] [--think]

DESCRIPTION

I worked in a variety of environments, sometimes outside of my comfort-zone where I have been forced to learn and run technologies in production at the same time. I pursue issues with a passion (as any good DBRE/SRE would) and don't settle for it just works. I love learning new technologies and wrapping my head around new paradigms, ideally while enjoying dark-roast coffee at my standing desk.

OPTIONS

--store=*SQL*

MySQL is my true love.

- Verifying database workloads, identifying schema bottlenecks, tuning innodb settings, while providing an HA database can be a lot of fun
- Scaled Runtastic's core database starting at 1M users to 99M users and the ability to handle 100k QPS
- Migrated 90+ microservice DBs from Cloudsql to RDS with 7 seconds of downtime
- Partitioning a table or verifying the innodb engine status are things i consider fun
- Experience finding and debugging DBaaS provider based issues while gathering all the metrics possible
- Designed and implemented upgrade system for 200+ mysql 5.7 databases to 8.0 with 3 seconds of downtime

PostgreSQL is different but fun

- Used it with all its geo-spatial features for OpenStreetMap
- Automated replicating to another database with slony before 9.0 came out with log shipping
- Automated replication and fail-over with repmgrd/patroni
- Had lots of fun automating pgbouncer with consul as a HA write endpoint
- Have come to embrace MVCC and the pros and cons that it brings
- Have solidified my understanding of the pg_% tables and the plethora of information they provide
- Love (and sometimes hate) the pg_stat_Statements extension

Oracle is what i cut my teeth on.

- Had to learn on highly performant, always-on database systems
- Learned what a missing index can do, which queries to avoid and how to find the meaning of all those v$ views
- The experience I gathered understanding B-Tree index look-ups, helped me to understand look-ups across all database systems

MSSQL

- Enjoyed a love/hate relationship with Azure SQL server as a core database
- Love the depth of the query planner and the insight into live statistics
- Hate the fact that indexes are built around fragmentable structures and inline statistics calculation is enabled by default

--store=***NOSQL***

MongoDB is webscale... Schema-less?!

- Attended admin training courses to understand: schema-less can also be a curse since it is incredibly challenging when defining a scalable, future proof indexing concept
- Enjoy shard key definitions for collections to ensure the fewest scatter-gather queries, Optimizing storage read ahead settings and playing with Wiredtiger cache options
- Have lots of experience running the robust HA setup which is very easy to run from an operations standpoint

Cassandra forgives a lot.

- I find the replication approach and ring setup amazingly robust with anisotropic read repairs and the ability to stream and receive data while accepting queries
- Scaled Runtastic's cluster from 4 to 8 to 32 and finally to 64 nodes with almost no downtime

Redis and Redis cluster are my nemesis.

- Run instances under normal conditions as well as high load condition
- Have found the right balance between RAM and IOs to ensure that they can flush to disk
- Since Redis is the core of many ruby queueing solutions, I have been forced to face my fears, tuning BGSAVE cycles and finding the optimal fsync to AOF ratios

CouchDB eventually consistent fun

- Was thrown into cold water with multiple clusters running at high load
- When building the prometheus exporter for a CouchDBaaS provider, learned about many of the internal stats and their meanings.
- Translated that knowledge into practical experience by building Kubernetes powered scale-out deployments to load test, while starting an ever growing comprehensive cluster administration documentation.

--administrate

(run, play, break, repeate)

LAMP

- Traditional stack (e.g. apache2, nginx)
- Extended web servers (e.g. trinidad, passenger)

Loadbalancers

- Experience debugging performance bottlenecks
- Setup instances which handle > 80k rpm
- HaProxy, nginx

Golden Oldies

- LOTS of experience with the normal Linux stack (e.g. bind, dhcpd, ldap, openvpn, ssh, memcached ...)
- After > 15 years experience with all aspects of the os, i still think Linux is the best for servers

Distributed fun

- zookeeper
- rabbitmq
- activemq
- nats (with jetstream)
- consul (for service discovery)
- All in clusters running at least 3 nodes
- Very interesting (i.E. challenging) to scale

GitLab

- Run at scale (gitlab.com) as well as company wide implementations.
- Experience with the joys and pains of CI implementation and administration.

--automate

(Automation, testing and auditing is inevitable in today's world of highly fluctuant infrastructure)

Chef

- Wrote and deployed cookbooks for every aspect of Runtastic's infrastructure
- Try to ensure that all infrastructure code has full test coverage
- Test-Kitchen, inspec and chefspec are my friends

Ansible

- Wrote and deployed roles to automate cache layer deployments
- Discovered the love/hate relationship in the python's jinja2

Terraform

- Compiled modules to simplify complex deployments
- Wrote a provider to interface with OpenNebula
- Used to deploy all aspects of Cabify and Fonoa's non application layer infrastructure

Kubernetes

- flux - to run large and small infrastructure
- ArgoCD - and the app-of-app-of-apps
- Helm - the joys and the follies

--monitor

(No observability, means not knowing anything)

Nagios3

- Wrote and deployed numerous checks
- Running an nrpe based deployment with full automation
- > 10k checks distributed across 1k servers

Collectd

- Wrote and integrated checks for nfs-iostat and mongodb
- Running and fully automated with a graphite front end
- Collecting > 100k metrics an hour

Cacti

- Implemented Percona's graphing suite for mysql
- Collect all core database metrics from connections to innodb flush times

Prometheus

- Alert-manager, recording-rule, exporter - oh my. Very powerful solution with an ever growing community? Count me in.
- Wrote recording/alerting rules with unit tests
- Experience with some storage engine and memory shenanigans
- Visualization with graphana
- Wrote exporters for databases and weather stations

Cloud Services

- Very familiar with New Relic, Pingdom, Dynatrace, PagerDuty, VictorOps

--virtualize

(control your destiny - as much as you can)

OpenNebula

- Experienced every phase of growth from 8 hypervisors to 60
- Have run opennebula as an EC2 replacement as a native cloud (extensive API) and as a simple server manager
- In the process of automating setup and configuration via teraform

KVM

- Qemu based
- NFS and Ceph storage backend
- Currently use it as a minikube virtualizer

Virtualbox

- Runtastic's pre-production system ran on vbox for a long time (hard to imagine)
- Mainly running older cookbook tests with vbox

Cloud Services

- Automate Google Cloud Platform (GCP) and Azure instance deploys with terraform
- Experience the joys (its so easy) and pains (why is the db rebooting?) of not controlling your hypervisors

--containerize

(run it like mike)

Docker

- Write dockerfiles to encapsulate many applications
- Build typical applications as well as X based, multi-arch, multi-stage ones
- Automated container builds with GitLab CI and BATs

Kubernetes

- Wrote many manifests for different applications, ranging from banal to complex
- Run my own cluster on RaspberryPis for all my home needs
- Gave a talk at SFSCon about using Flux to automate manifest deployments: link

Nomad

- Experience running complex and simple jobs
- Integrated with other HashiCorp products (Consul, Hashiui)

LXC

- As a plugin for new test-kitchen deployments
- Played around a bit LXD

--script

Bash

- wrote extensive bash scripts for automation with unit tests (bats)
- found out that bash has its limits :)

Golang

- Enjoy writing and maintaining a go backend for a research project
- Wrote gobench to benchmark schemas in mysql and postgres for high throughput
- Learned about api design the hardway while using grpc

--think

imho

- vim > emacs
- zsh > bash
- tmux > screen

HISTORY

Cabify (05.2023 - .)

Database Reliability Engineer

- Built automation for seemless mysql 5.7 to 8.0 live migrations
- Took on a more staff engineering role focused on mentoring and teaching
- Drove redefinition of higher level engineers role through out the company
- Re-imagine how database related support is handled with data and chat-gpt
- Fully remote

Fonoa (01.2022 - 04.2023)

Database/Site Reliability Engineer

- Automate, run, manage all database related technologies: MySQL/Postgres (Google Cloudsql), MsSQL (Azure)
- Ensured that monitoring and alerting was availible with end2end testing
- Build exporter for missing SQL server metrics
- Create runbooks for oncall team members with little database context to ensure service continuity
- Write design guides to help developers understand their schema and engine decisions
- Fully remote
- Downsized out of a job :/

Cabify (05.2018 - 11.2021)

Database Reliability Engineer

- Tasked with automating, managing, running all database related technologies: MySQL (Google Cloudsql), Couchdb (Cloudant), Redis, Memcached, Elasticsearch
- Made fully monitored, highly available database creation self service
- Build exporters for missing observability in DBaaS platform
- Automate no-downtime sql based CI powered schema changes
- Continually document and assist developers in making persistence decisions
- Support developers in identifying design bottlenecks in query pattern, database design.
- Fully remote

GitLab (04.2017 - 04.2018)

Senior Production Engineer

- Memeber of a small fully remote team
- Scale gitlab.com (millions of users) using GitLab (typically built for thousands of users) in a cloud environment
- Collaborate on developing HA solution for PostgreSQL in the GitLab omnibus package
- Strove to fully automate environments from terraform to multi-tiered HA stack
- Build a back-end agnostic solution for secrets in chef
- Use chef to automate all-the-things
- Fully remote

Runtastic GmbH (04.2012 - 03.2017)

Infrastructure Architect

- Define setup and strategy for each upcoming stack
- Ensure scalability of technologies and concepts
- Setup workflows for automation and deployments

Head of Operations

- Organize small team while fighting to stay ahead of growth
- Very challenging for me to lead a team of inexperienced ops and shaping our infrastructure

Operations Engineer

- Nested under the web development team
- Start automation
- Improve uptime through monitoring and derive future actions
- Conceptualize private cloud based on opennebula

APEX Gaming (04.2010 - 03.2012)

Head of Customer Care

- Setup ticketing workflow based on ITIL best practices
- Created automated master/slave setup with slony for PostgreSQL 8.3/8.4
- Spent time training staff in the casino headquarters to be first level support techs

Knapp Systems Integration (09.2007 - 03.2010)

Technical Project Lead

- Introduce metric collection to visualize hardware utilization for the customer
- Manage customer care projects
- Responsible for everything from planning to doing
- Largest project was complete warehouse upgrade to a medium sized 24x7 cosmetic distributed
- Organized and held numerous on-site training courses around the world

Second Level Support Engineer

- Field production problems in a 24x7 environment
- Handle issues ranging from PLC (Siemens S7) to tablespace cleanups on a core Oracle instance

FH Joanneum - University of Applied Science

Bachelor of Science in Software Engineering

- Extra-occupational program

SEE ALSO

- Email: jason.tevnan@gmail.com
- Phone: +43.650.2167444
- LinkedIn: https://at.linkedin.com/in/jason-tevnan-5390b4a8

BUGS

Prone to flu if left in rain.

AUTHOR

Jason Tevnan (jason.tevnan@gmail.com)