

SysDB – System DataBase

Ein Datenaggregator für System-Informationen

Sebastian 'tokkee' Harl

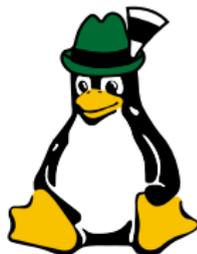
<sh@tokkee.org>

SysDB developer

Grazer Linxutage 2014

05. April 2014

Graz





HINWEIS:

SysDB ist noch in einem frühen Entwicklungsstadium.

Flaming, Bashing oder andere Formen von konstruktivem Feedback sind sehr willkommen! :-)

Motivation / Hintergrund



Warum SysDB?

- Wer verwendet Nagios/Icinga/Naemon/OpenNMS/etc.?



Warum SysDB?

- Wer verwendet Nagios/Icinga/Naemon/OpenNMS/etc.?
- Wer verwendet außerdem Puppet/Chef/etc.?



Warum SysDB?

- Wer verwendet Nagios/Icinga/Naemon/OpenNMS/etc.?
- Wer verwendet außerdem Puppet/Chef/etc.?
- Wer verwendet außerdem collectd/Munin/etc.?



Warum SysDB?

- Wer verwendet Nagios/Icinga/Naemon/OpenNMS/etc.?
- Wer verwendet außerdem Puppet/Chef/etc.?
- Wer verwendet außerdem collectd/Munin/etc.?
- Wer hat außerdem eine CMDB?



Warum SysDB?

- Wer verwendet Nagios/Icinga/Naemon/OpenNMS/etc.?
- Wer verwendet außerdem Puppet/Chef/etc.?
- Wer verwendet außerdem collectd/Munin/etc.?
- Wer hat außerdem eine CMDB?
- Wer ist mit seinem Setup zufrieden? ;-)



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.
- Zweiter Gedanke: wenn man dass dann schon hat, kann man die Daten auch gleich noch mit Monitoring Informationen annotieren.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.
- Zweiter Gedanke: wenn man dass dann schon hat, kann man die Daten auch gleich noch mit Monitoring Informationen annotieren.
- Und wenn man dann schon so weit ist, sollte man es gleich generisch machen.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.
- Zweiter Gedanke: wenn man das dann schon hat, kann man die Daten auch gleich noch mit Monitoring Informationen annotieren.
- Und wenn man dann schon so weit ist, sollte man es gleich generisch machen.
- Dann stellte ich fest, dass ich Web-UI-Entwicklung gar nicht mag.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.
- Zweiter Gedanke: wenn man dass dann schon hat, kann man die Daten auch gleich noch mit Monitoring Informationen annotieren.
- Und wenn man dann schon so weit ist, sollte man es gleich generisch machen.
- Dann stellte ich fest, dass ich Web-UI-Entwicklung gar nicht mag.
- Außerdem braucht eine skalierbare Anwendung ein gutes Backend.



- Eigentlich wollte ich ein neues collectd Frontend für große, heterogene Umgebungen.
- Erste Idee: Gruppierung von collectd-Datensätzen an Hand von Puppet Facts der entsprechenden Hosts.
- Zweiter Gedanke: wenn man dass dann schon hat, kann man die Daten auch gleich noch mit Monitoring Informationen annotieren.
- Und wenn man dann schon so weit ist, sollte man es gleich generisch machen.
- Dann stellte ich fest, dass ich Web-UI-Entwicklung gar nicht mag.
- Außerdem braucht eine skalierbare Anwendung ein gutes Backend.
- → die Idee von SysDB war geboren

Die System DataBase



- SysDB sammelt Informationen zu beliebigen Hardware- und Software-Systemen
- Einfache Beispiele:
 - Hosts und deren Eigenschaften („Facts“)
 - Services und deren Eigenschaften
 - Monitoring-Informationen (z.B. Zustand)
 - Performance-Daten (z.B. CPU-Auslastung)
- SysDB sammelt diese Informationen von verschiedenen Systemen und korreliert zusammengehörige Informationen



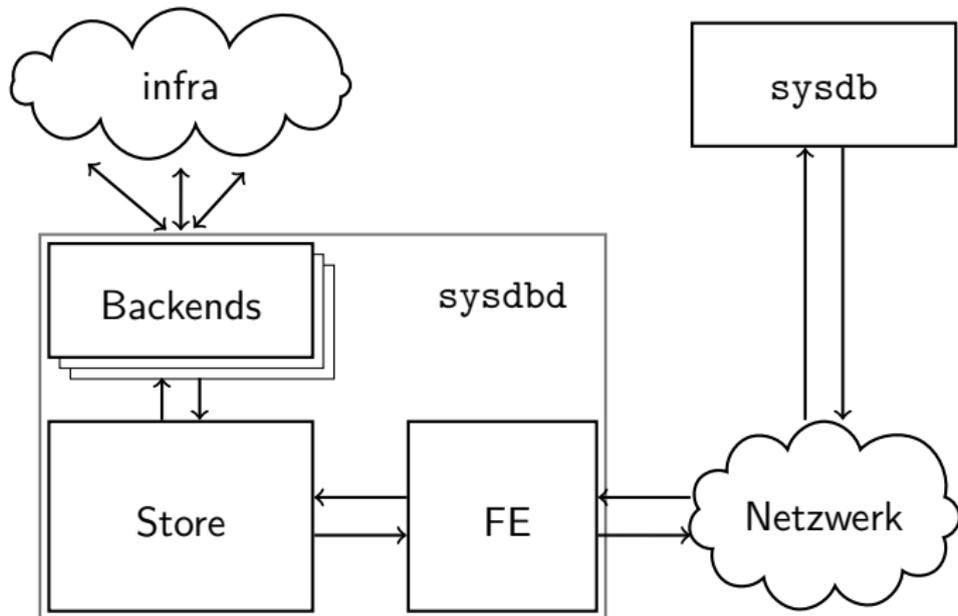
- <https://github.com/tokkee/sysdb>
 - CI: <https://travis-ci.org/tokkee/sysdb>
 - > 60% Code (Function) Unit-Test Coverage im Core
- BSD Lizenz
- Geschrieben in C
- Einfach erweiterbar (einfache Plugin API)
- Einfaches Netzwerk-Protokoll
- Großteil des Codes als (Satz von) Bibliotheken implementiert (wiederverwendbar)



Aktuell verfügbar:

- `collectd::unixsock` – query collectd's UNIX socket interface
- `mk-livestatus` – query Monitoring Systeme (Nagios, Naemon, Icing, Shinken) über Check_MK Livestatus
- `puppet::store-configs` – query Puppet
- `cname::dns` – Normierung von Hostnames mittels DNS
- `syslog` – syslog Logging

Geplant: Passive Datensammlung (z.B. mittels Gearman), Foreman, PuppetDB, `$your_favorite_system` (send patches!) :-)

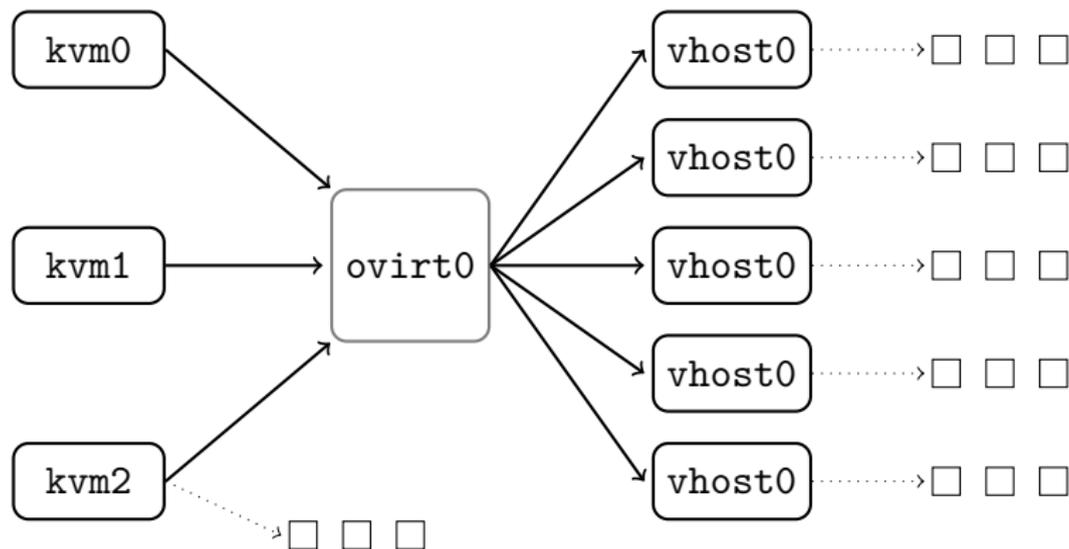




- Die eigentliche Datenbank
- Speichert Objekte verschiedener Typen
 - Host – jede Art physikalischer Ressource
 - Service – Dienst (im weitesten Sinne)
 - Attribute – Eigenschaften von Hosts oder Services
- Normierung von Hosts (geplant: Services)
- Jedes Objekt speichert Zeitpunkt der letzten Aktualisierung und automatisch berechnetes Update-Intervall → wichtig!

- Schnittstelle zur Abfrage von Objekten
- JSON als primäre externe Repräsentation von Objekten

Der SysDB Store – Beispiel





```
message Host {  
    required string name;  
    required DateTime last_update;  
    required Interval update_interval;  
  
    repeated Attribute attributes;  
    repeated Object children; // Host or Service  
    optional Object parent;  
}
```



```
message Service {  
    required string name;  
    required DateTime last_update;  
    required Interval update_interval;  
  
    repeated Attribute attributes;  
    repeated Object children; // Host or Service  
    repeated Object parent; // mehrere!  
}
```



```
message Attribute {  
    required string name;  
    required Type value;  
    required DateTime last_update;  
    required Interval update_interval;  
  
    optional Object parent;  
}
```

Type ist Zahl, Dezimalzahl, String, DateTime oder Binär.
Geplant: benutzerdefinierte Datentypen.



- Interaktives Client-Programm für SysDB
- Verbindet sich mit einem SysDB-Daemon
- Interaktive Kommando-Shell
- Asynchrone Log-Meldungen vom Server



- Entfernt an SQL angelehnt
- Darauf ausgelegt, auch einfach in RPCs verwendet zu werden
- Noch sehr stark in der Entwicklung ;-)
- Bislang unterstützte Befehle:
 - LIST – Auflistung aller Hosts
 - FETCH <hostname> – Details eines Hosts
 - LOOKUP <type> WHERE <expression> – Details mehrerer Hosts mit angegebenen Eigenschaften



```
sysdb=> LIST;
{"hosts": [{
  "name": "nagios.lxc.tokkee.net",
  "last_update": "2014-04-03 10:26:41 +0200",
  "update_interval": "5m4s"
}, {
  "name": "puppet.lxc.tokkee.net",
  "last_update": "2014-04-05 11:04:08 +0200",
  "update_interval": "5m2s"
}, {
  "name": "whisky.mobile.tokkee.net",
  "last_update": "2014-04-05 11:09:15 +0200",
  "update_interval": "10s"
}]}
```



```
sysdb=> FETCH 'whisky.mobile.tokkee.net';
{
  "name": "whisky.mobile.tokkee.net",
  "last_update": "2014-04-05 11:09:15 +0200",
  "update_interval": "10s",
  "attributes": [{
    "name": "architecture", "value": "amd64",
    "last_update": "2014-04-03 10:26:41 +0200",
    "update_interval": "5m3s"
  },{ ... }],
  "services": [{
    "name": "cpu-0/cpu-idle",
    "last_update": "2014-03-31 23:34:06 +0200",
    "update_interval": "9.716754301s"
  },{ ... }]
```

Anwendungsbeispiele



- CMDB Ersatz bzw. Erweiterung
 - neues Backend zur direkten Abfrage beliebiger Werte aus externen Datenbanken
- Abgleich diverser Backends (intelligenteres Monitoring)
 - Welche Hosts / Services fehlen in welchem Backend?
 - Wie ist der Gesamtstatus meiner Windows-Systeme in Rechenzentrum XYZ?
- Intelligentes Web-Frontend für mehrere Backends
 - Zentrales Dashboard

Ausblick: künftige Entwicklung



- Mehr Dokumentation!
- RDBMS Backend (PostgreSQL mit hstore, aber prinzipiell modular)
- Schnittstelle zur Abfrage von Performance-Daten
- Verteilte Architektur (HA und Load-Balancing)
- Web-Interface (in Go, wenn von mir ;-))
- Ausbau des Typ-Systems und Filter-Syntax
- Unterstützung für andere Arten von Informationen (?):
 - Personen / Gruppen (ACLs, ...)
 - Events / Kalender (Change-Requests, ...)
- ...



Danke für die Aufmerksamkeit!

Fragen, Kommentare, Anmerkungen, Rants?



Feedback:

<http://glt14-programm.linuxtage.at/events/327.de.html>



Kontakt:
Sebastian “tokkee” Harl
<sh@tokkee.org>

<https://github.com/tokkee/sysdb>

Please send patches! :-)