

Monitoring as a service

...

Fault detection with sensu and puppet

Tomasz Tarczyński @ttarczynski

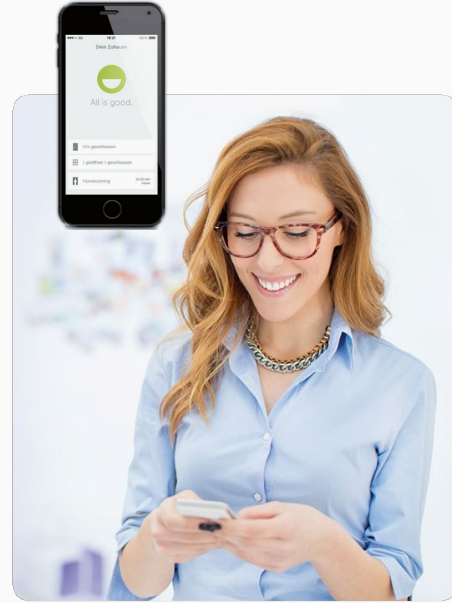
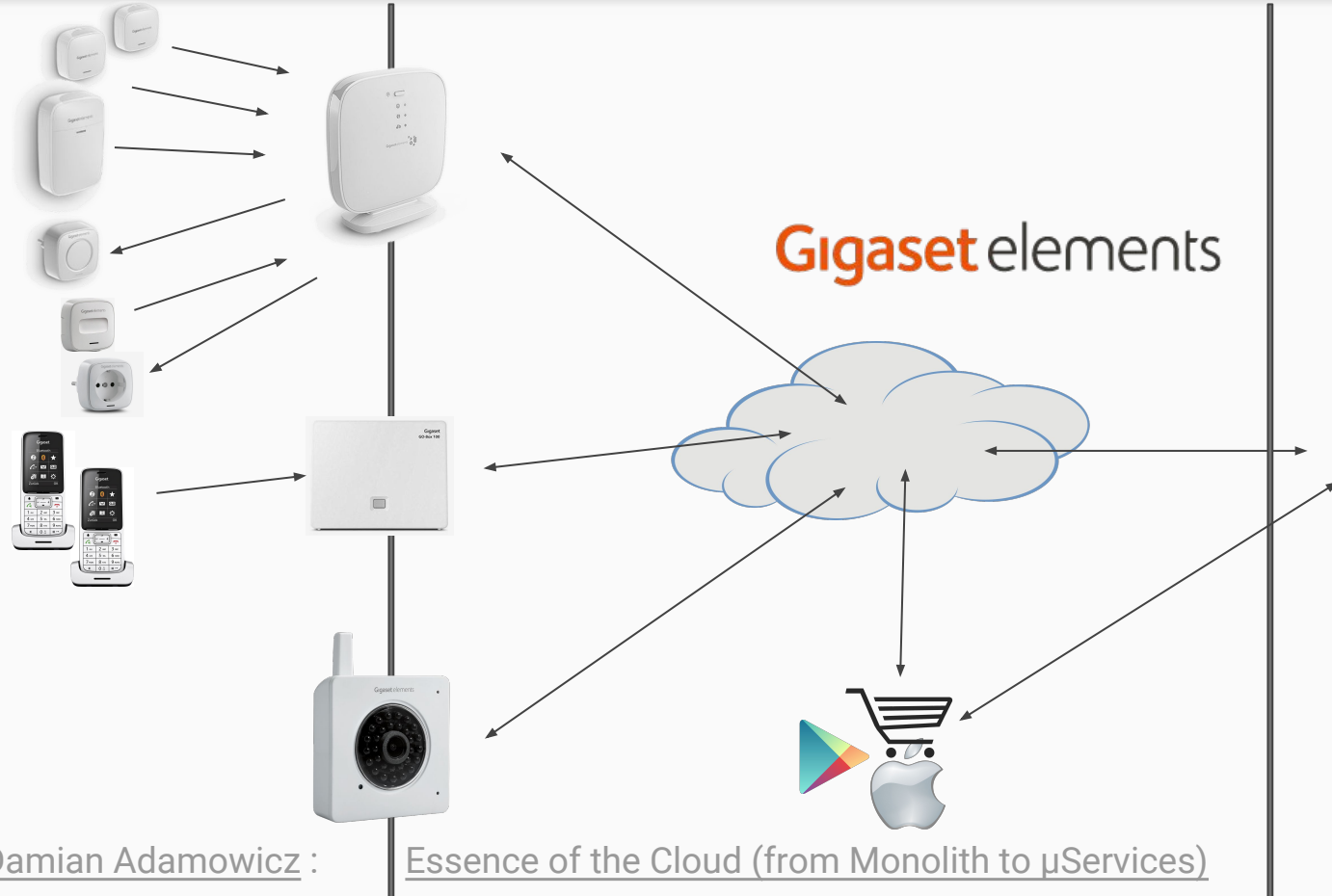
Plan

1. Making monitoring a service
2. Evolution of monitoring
3. Infrastructure as code
4. SENSU architecture and configuration
5. Extra addons

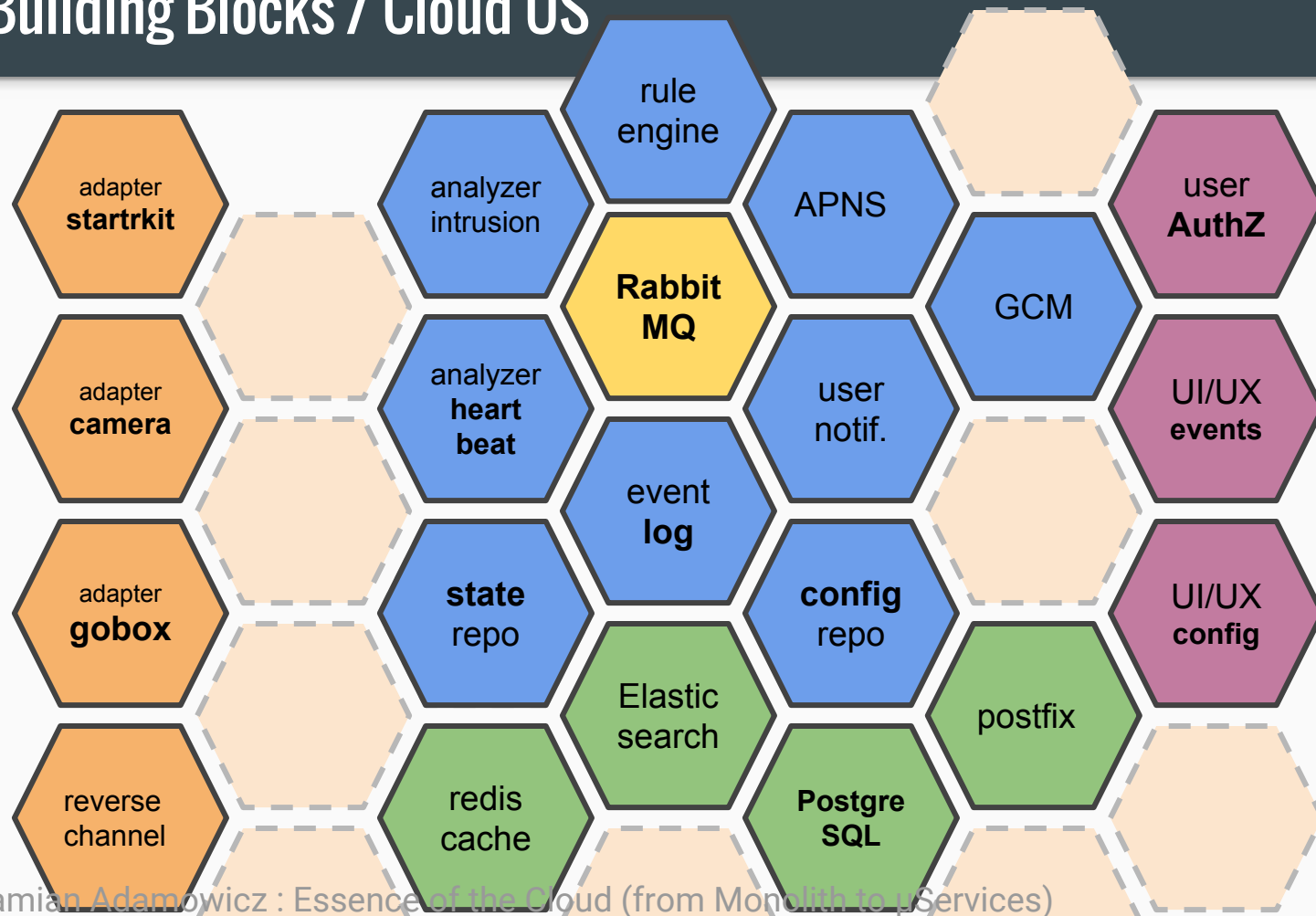
Gigaset elements

- Smart Home system
- Internet of Things

Gigaset elements



Building Blocks / Cloud OS



Monitoring As A Service

James Turnbull - Monitoring As A Service - Modernity and Self-Service

USTREAM

Monitoring As A Service Modernity and Self-Service

James Turnbull
@kartar



00:12

40:02



JAMES TURNBULL

THE ART OF MONITORING



James Turnbull @kartar

Making monitoring modern

It should focus on:

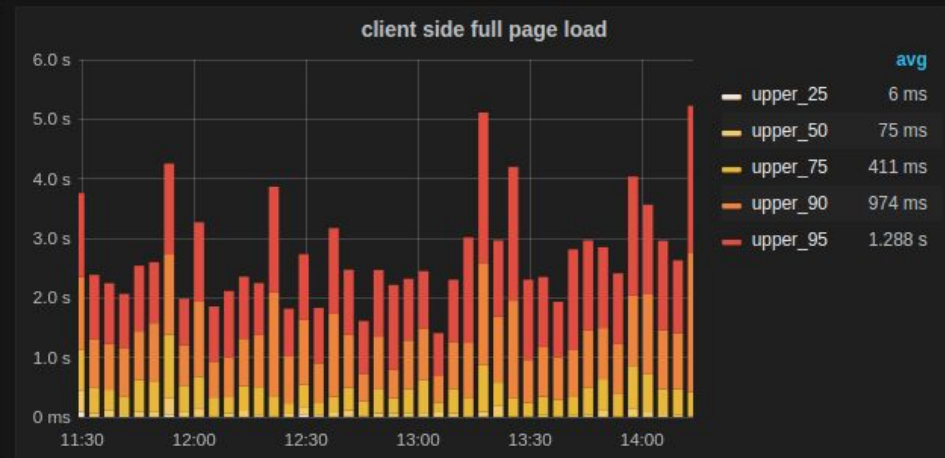
1. **Business logic**
2. Applications
3. Services
4. Infrastructure



Making monitoring modern

- Focus on user experience and performance
- Metrics
- Automation





Making monitoring a service

- **You are not the only customer of your monitoring**
- Customers:
 - **The business**
 - Engineering
- Attach monitoring to product development

Making monitoring a service

- **Self-service**
- Present an API not a ticket
- Dashboards / Logging / Alerting

Our old setup

- **Logging** – ELK stack
- **Measurements** – Nagios performance data
- **Fault detection** – Nagios
- **Alerting** – e-mail / SMS-gateway



Our old setup: Nagios

- 10k checks – 4 Nagios servers
- Check_MK – for better scaling
- PNP4Nagios – performance data / graphs
- Nagios-API – REST
- Nagdash – dashboard to observe alerts from all instances

Limitations of the old setup

- Poor automation
 - most configs edited manually
 - inconsistent environments
- Complicated setup
 - client / server / check_mk
- Alert fatigue
 - many false positive alerts
- Poor notifications
 - no incident lifecycle
- Performance
 - checks every 5 minutes (or worse)

Decompose it

- **Logging**
- **Measurements**
- **Fault detection**
- **Alerting**

– ELK stack



– ?

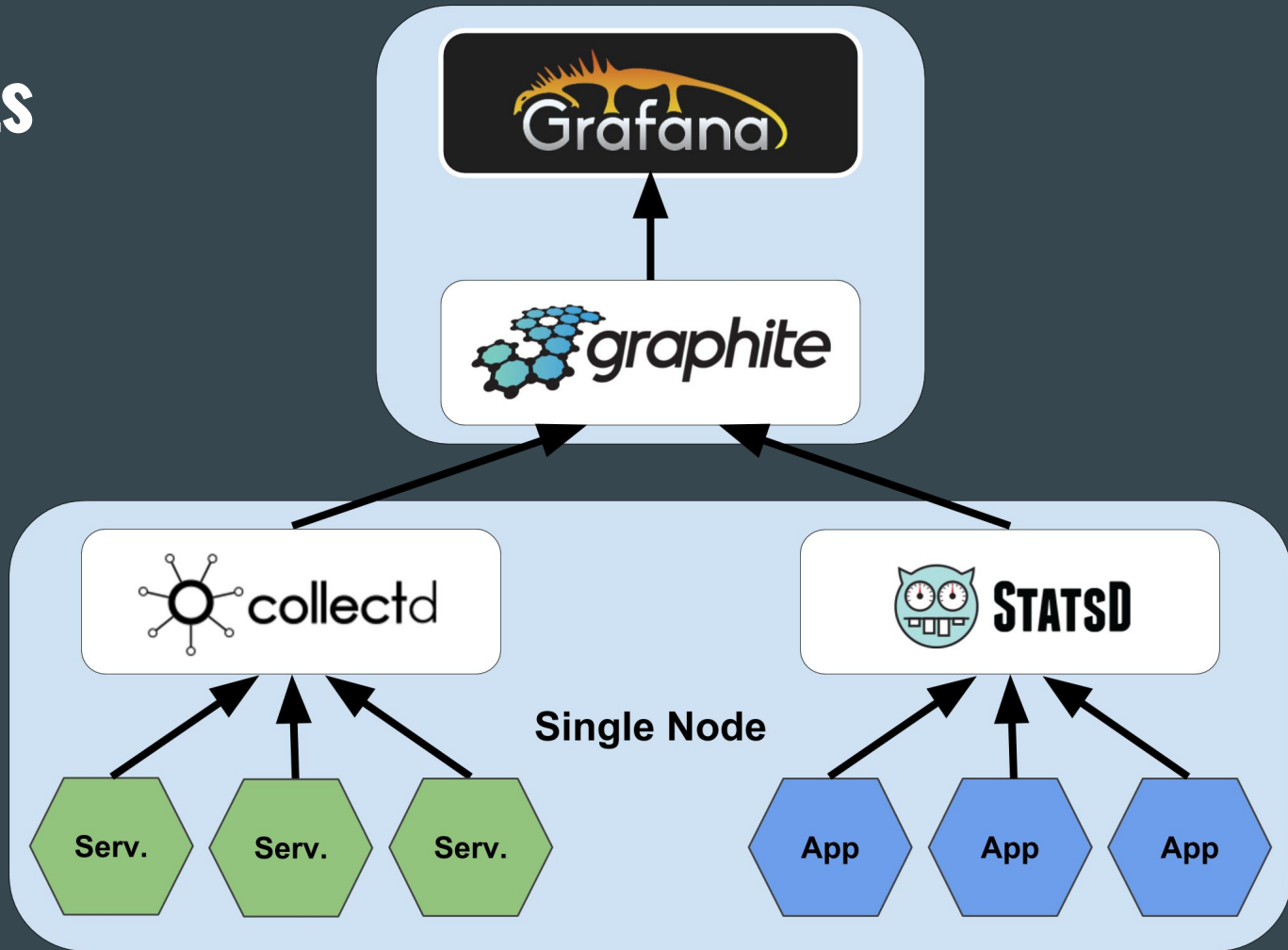
– ?

– ?



(notifications + incident management)

Measurements


- Statsd
- Collectd
- Graphite
- Grafana



Decompose it




- **Logging**
 - ELK stack 
- **Measurements**
 - Statsd + CollectD + Graphite + Grafana 
- **Fault detection**
 - ?
- **Alerting**
 - ?
(notifications + incident management)

Alerting: Notifications + Incident management

- Alerting as a Service: 
 - on-call schedules, rotations and escalations
 - guaranteed alert delivery,
to the right person with the right information



Decompose it

- **Logging**
 - ELK stack 
- **Measurements**
 - Statsd + CollectD + Graphite + Grafana 
- **Fault detection**
 - ?
- **Alerting**
 - PagerDuty 
(notifications + incident management)

Configuration management - puppet

- **Automate everything** (manual work is error-prone)
- **Infrastructure as code** (and **monitoring as code**)
- Simple DSL (+ hiera for data)
- Convergence to desired state (runs every 30 minutes)
- **Modules** from community (**Puppet Forge**)

Infrastructure as Code

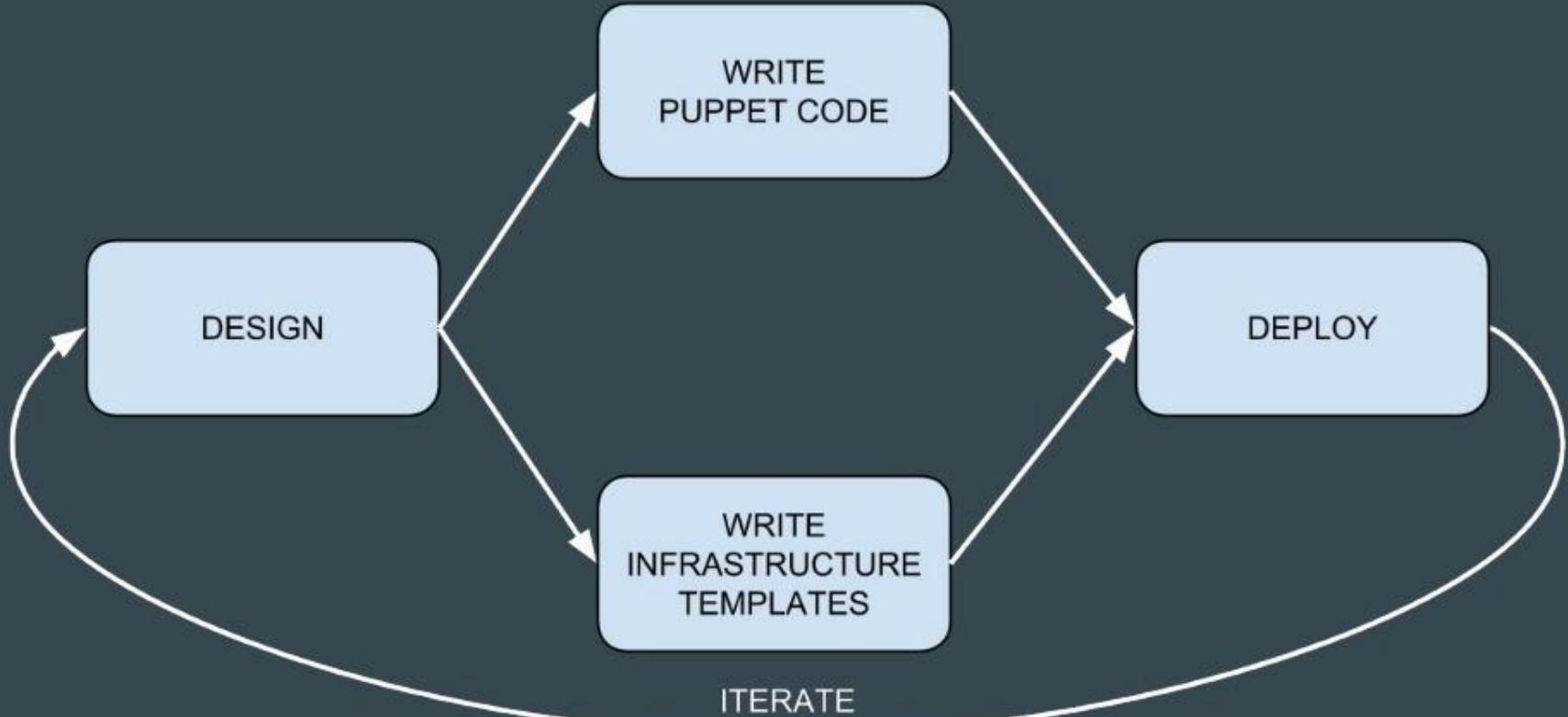
- Apply & **revert** changes quickly: move faster
- Delivering software: faster & reliably

BASIC IaC WORKFLOW

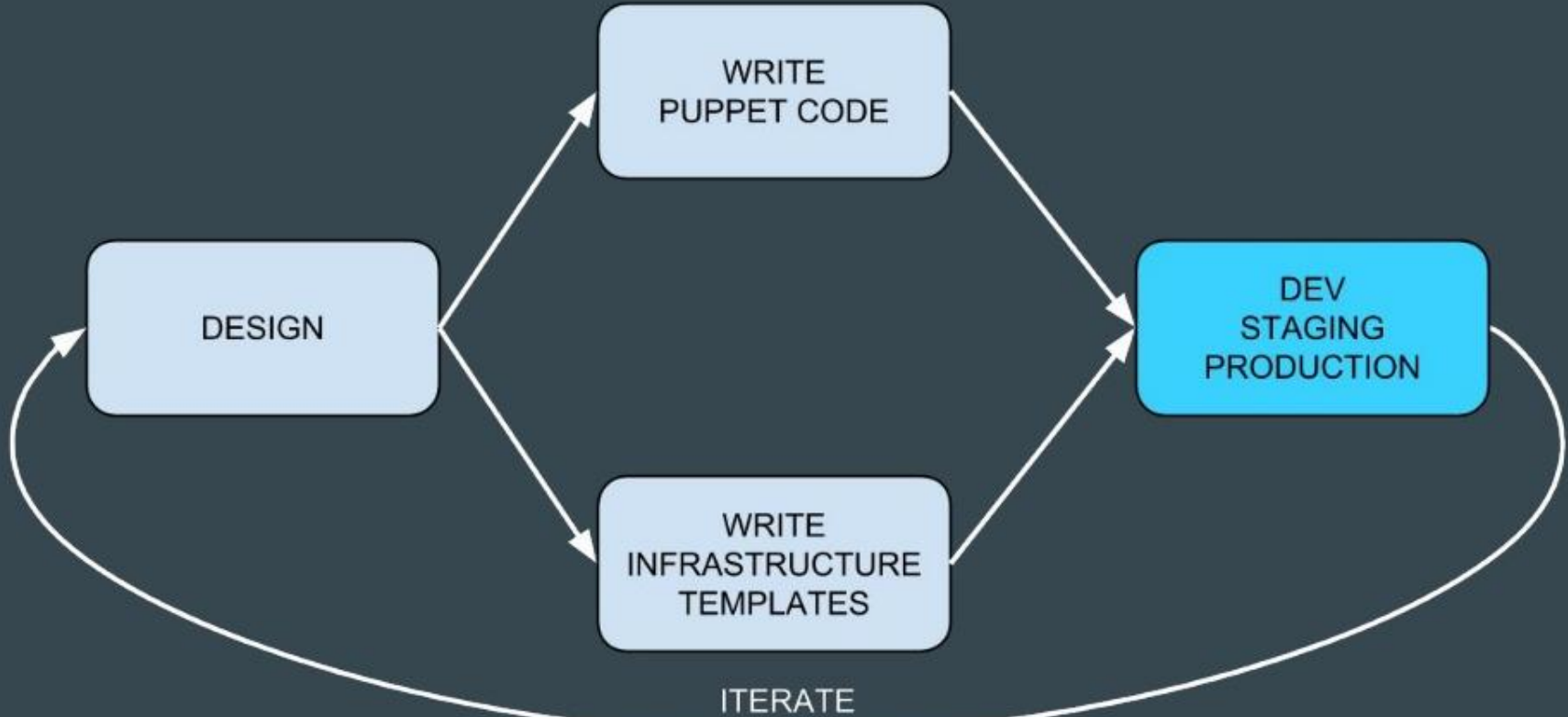


It's all **software**.

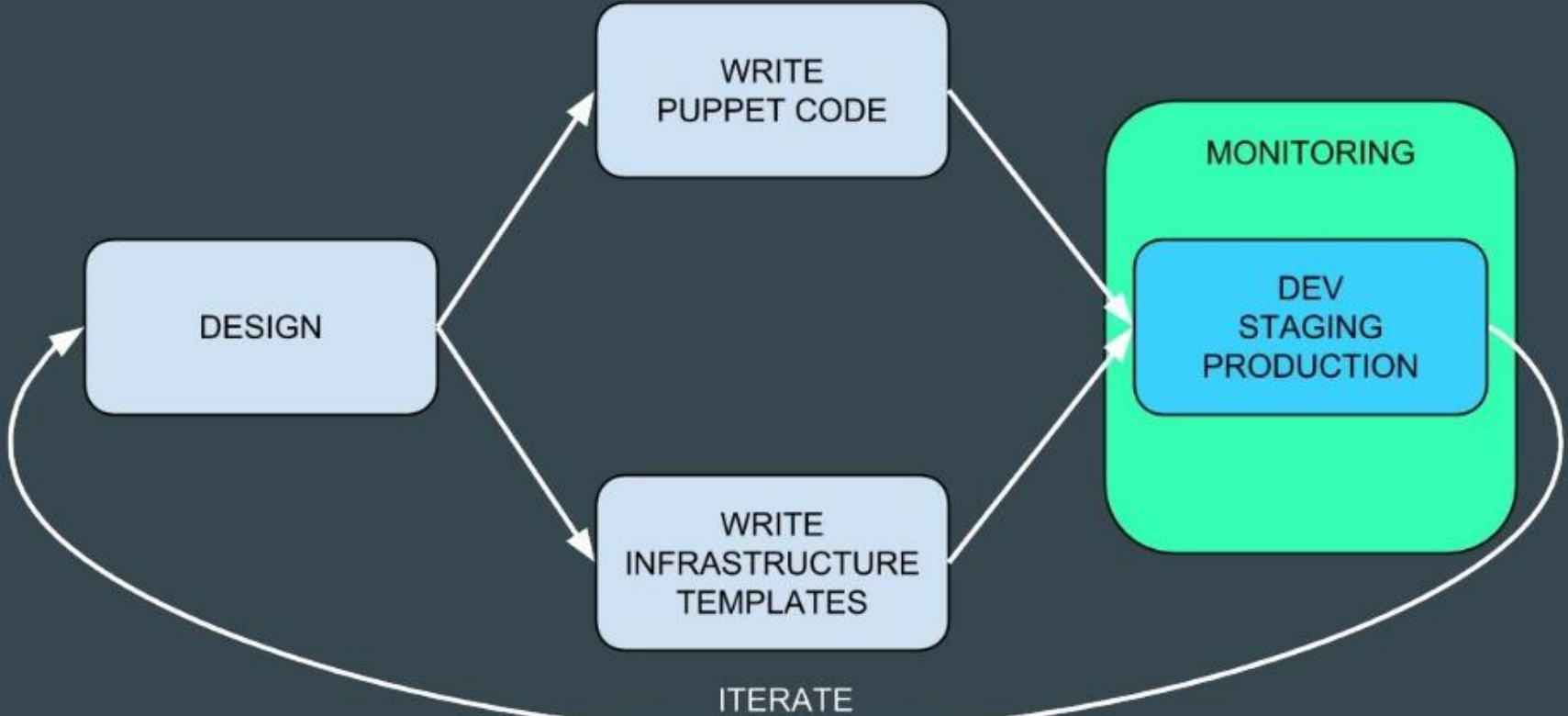
BASIC IaC WORKFLOW



BASIC IaC WORKFLOW



BASIC IaC WORKFLOW



BASIC IaC WORKFLOW



Improve the **feedback loop**.

Provides **continuous testing**.

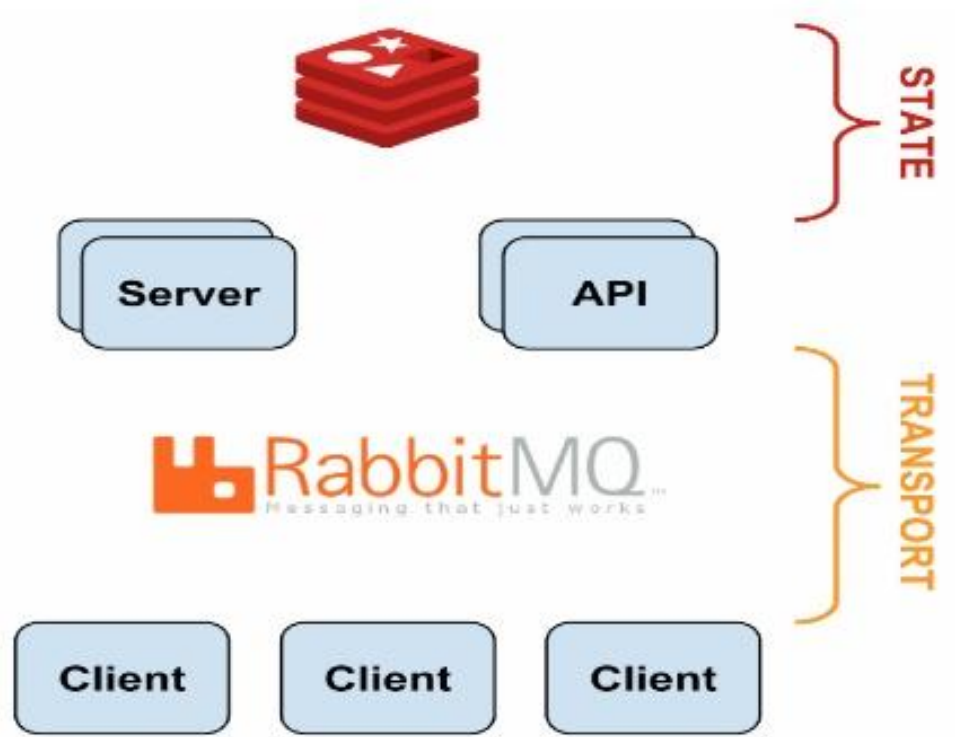
Fault detection - Sensus

- **Composed of simple components** (client/server/api/...)
- **Designed for automation** (configs in json)
- **API-driven** (REST)
- **Compatible with Nagios checks** (return 0/1/2)

Fault detection - Sensus

- **stand-alone checks** (configured and scheduled on client)
- **plug-in architecture** (checks/handlers/...)
- **Scalable** (horizontal scaling)

Sensu Architecture



Sensu **servers**

- Receive check results from **RabbitMQ**
- Create and process events
 - Filters
 - Mutators
 - **Handlers**
- Use **Redis** to persist:
 - **client registry**
 - **check results**
 - check execution history
 - current event data

Sensu **servers**

- **Daemons:**

- sensu-server
- sensu-api

- **Config files:**

- /etc/sensu/conf.d/api.json
- /etc/sensu/conf.d/rabbitmq.json
- /etc/sensu/conf.d/redis.json
- /etc/sensu/conf.d/handlers/*.json



sensu/sensu

by: [Sensu](#) [Project URL](#) [Report issues](#) [RSS Feed](#)

A module to install the Sensu monitoring framework

Latest version is compatible with:

- Puppet Enterprise 2016.4.x, 2016.2.x, 2016.1.x, 2015.3.x, 2015.2.x, 3.8.x, 3.7.x, 3.3.x, 3.2.x, 3.1.x
- Puppet >=3.3.0 <5.0.0
- [RedHat](#), [Windows](#), [Ubuntu](#), [CentOS](#)

Tags: [sensu](#), [monitoring](#)

To use this module, add this declaration to your Puppetfile:

```
mod 'sensu-sensu', '2.2.0'
```

[Learn more about managing modules with a Puppetfile](#)
[download latest tar.gz](#)

To manually install this module with puppet module tool:

```
puppet module install sensu-sensu --version 2.2.0
```

 **1,156,642**

Latest version: 44,915

APPROVED

Quality Score



5.0

No change with last release

[details](#)

Community Rating



4.7

Based on 20 questions answered

[details](#)

Tell us about this module

How helpful are the docs?



not at all

helpful

How easy to use?



hard

easy

Does what it promises?



Y

N

Works without changes?



Y

N

Used in production?



Y

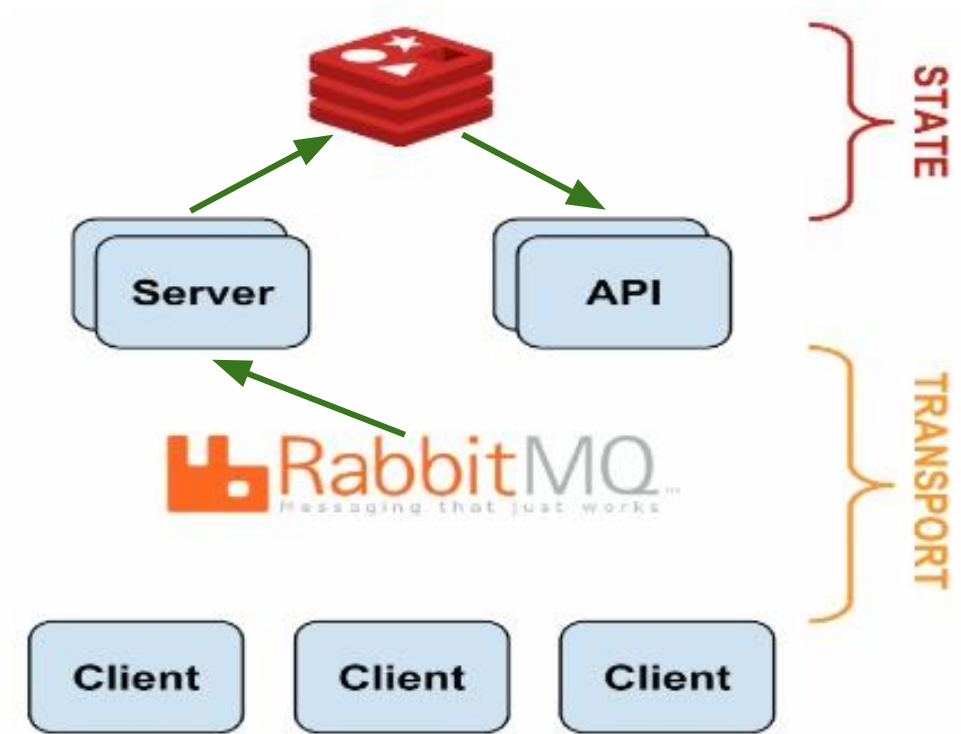
N

Configure a Sensu **server** - puppet manifest

```
class { '::sensu':  
  server      => true,  
  api         => true,  
  rabbitmq_password => 's3cr3t',  
  use_embedded_ruby => true,  
  version     => '0.26.5-2',  
}
```

```
class {'::redis': }  
class {'::rabbitmq': }
```

Sensu Architecture



Sensu **clients**

- monitoring agents providing:
 - dynamic **self-registration**
 - local check **execution scheduler**
 - **execution platform** for monitoring:
 - local system
 - local services
 - external resources
 - **input socket** for accepting input from external services

Sensu **clients**

- Daemons:
 - **sensu-client**
- Config files:
 - `/etc/sensu/conf.d/client.json`
 - `/etc/sensu/conf.d/rabbitmq.json`
 - `/etc/sensu/conf.d/checks/*.json`

Configure a Sensu **client** - puppet manifest

```
class { '::sensu':  
  rabbitmq_host      => 'sensu.foo.com',  
  rabbitmq_password => 's3cr3t',  
  use_embedded_ruby => true,  
  version            => '0.26.5-2',  
}
```

Configure a Sensu **client** - embedded Ruby

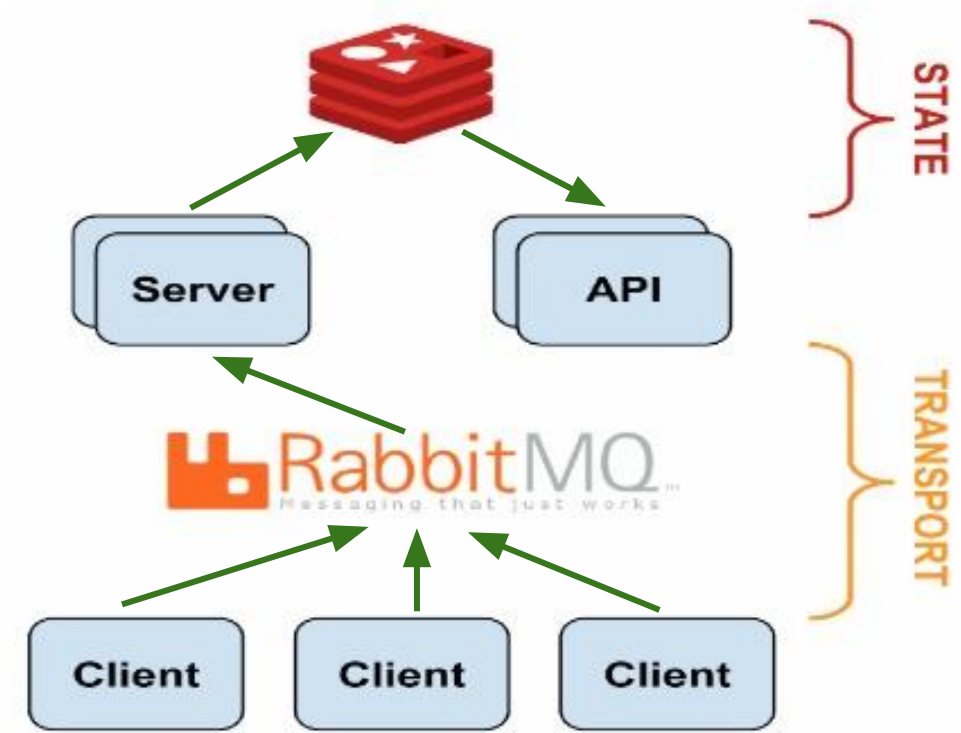
```
# ruby -v
```

```
ruby 2.0.0p598 (2014-11-13)
```

```
# /opt/sensu/embedded/bin/ruby -v
```

```
ruby 2.3.0p0 (2015-12-25)
```

Sensu Architecture



Sensu checks

- Commands executed by the Sensu client:
 - monitor condition
 - collect measurements
- Exit status code indicates state
 - 0 – “OK”
 - 1 – “WARNING”
 - 2 – “CRITICAL”
 - other – “UNKNOWN” or custom status



This organization Search

[Pull requests](#) [Issues](#) [Gist](#)



Sensu Community Plugins

The official home for the Sensu community plugins.

 [irc.freenode.net #sensu](#)  <http://sensu-plugins.io>

 **Repositories**

 **People** 6

Type: **All** ▾

Language: **All** ▾

1 result for repositories matching **http**

 Clear filter

sensu-plugins-http

This plugin provides native HTTP instrumentation for monitoring and metrics collection, including: response code, JSON response, HTTP last modified, SSL expiry, and metrics via `curl`.



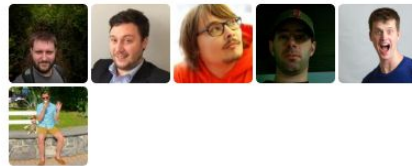
 Ruby  12  52 Updated 6 days ago

Top languages

 Ruby  Python  CoffeeScript
 HTML  Erlang

People

6 >



Configure a **check** - puppet manifest

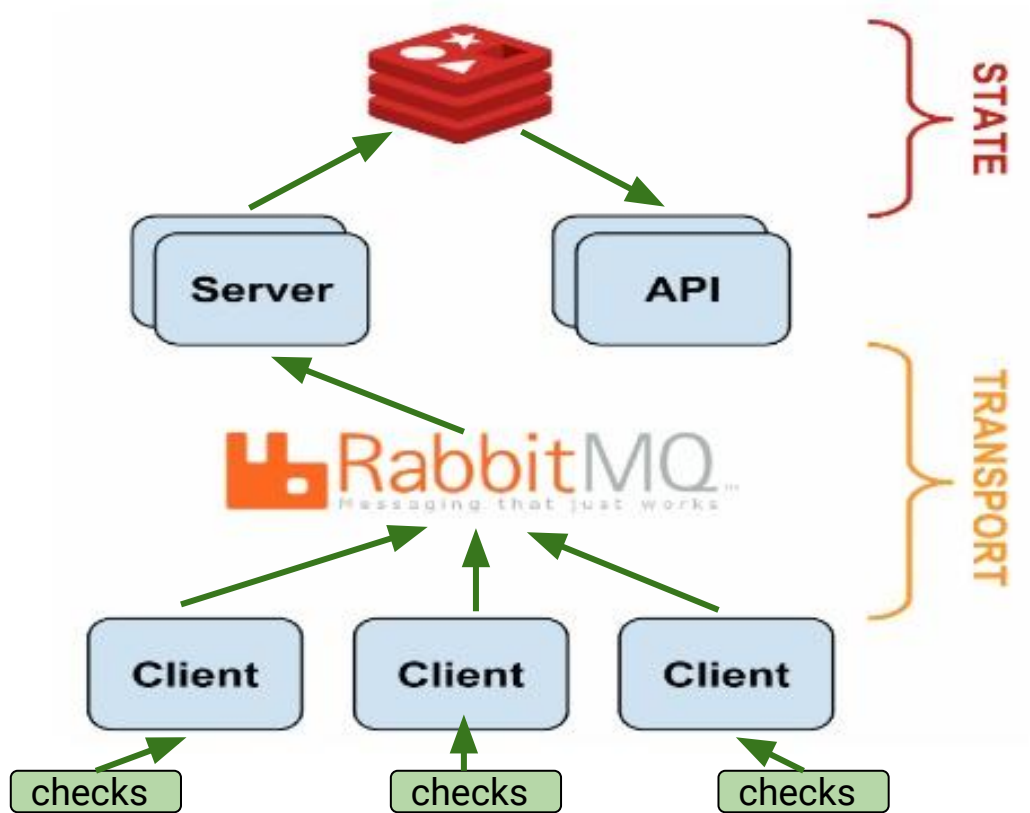
```
sensu::plugin { 'sensu-plugins-http':  
  type          => 'package',  
  pkg_provider => 'sensu_gem',  
}
```

```
sensu::check { 'api_http_response':  
  command      => 'check-http.rb --url http://127.0.0.1/health',  
  handlers     => ['pagerduty'],  
  interval     => 30,  
}
```

Configure a **check** - sensu_gem

```
# /opt/sensu/embedded/bin/gem dependency sensu-plugins-http
Gem sensu-plugins-http-1.0.0
  aws-sdk (~> 2.3)
  rest-client (~> 2.0)
  sensu-plugin (~> 1.2)
```

Sensu Architecture



Sensu **handlers**

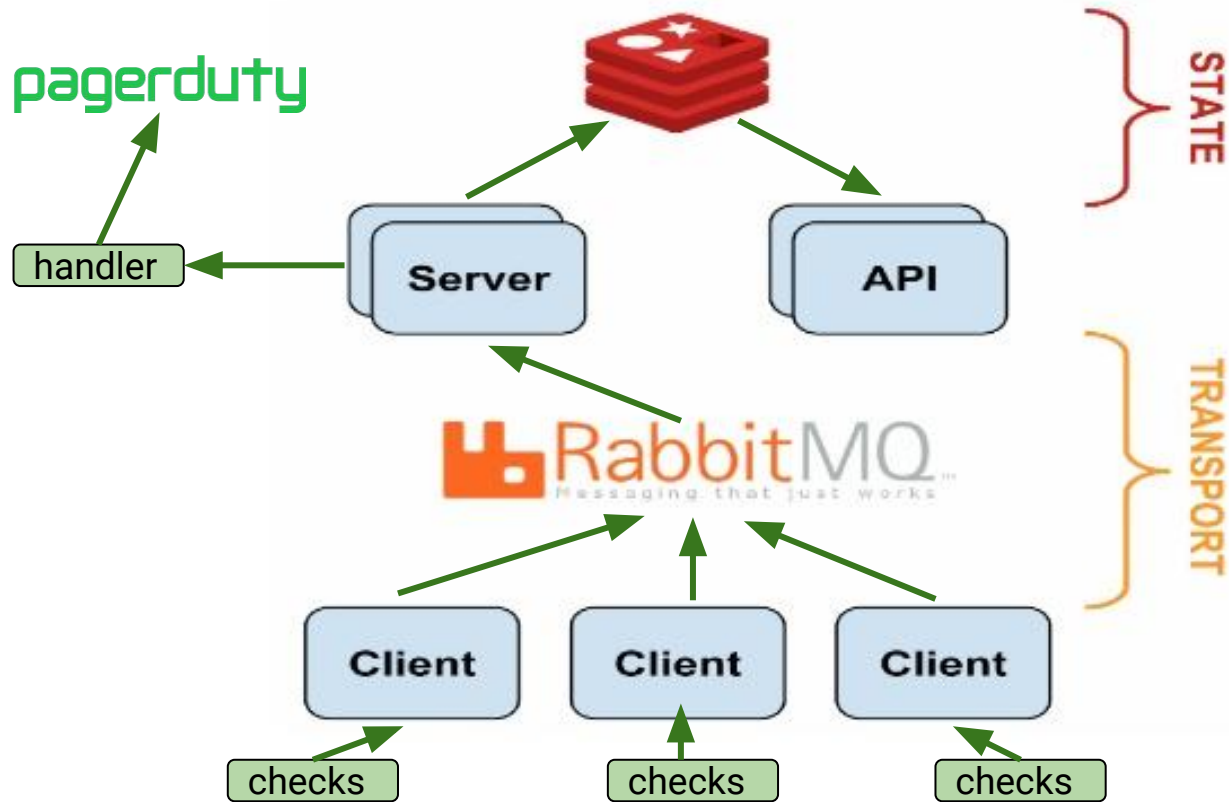
- actions executed by the Sensu **server** on **events**
 - send an email alert
 - **create or resolve an incident** (e.g. in PagerDuty)
 - store metrics in a time-series database (e.g. Graphite).

Configure a Sensu **handler** - puppet manifest

```
sensu::plugin { 'sensu-plugins-pagerduty':  
  type          => 'package',  
  pkg_provider => 'sensu_gem',  
}
```

```
sensu::handler { 'pagerduty':  
  command => 'handler-pagerduty.rb',  
  config  => {  
    'api_key' => 's3cr3t_key',  
  },  
}
```

Sensu Architecture



GUI : Uchiwa

- Multiple Sensus Instances
- Real-Time
- Silence Checks and Clients
- Simple Filtering

EVENTS >

ALL DATACENTERS ▾

BULK ACTIONS ▾

HIDE ▾

ALL CHECKS ▾

ALL STATUS ▾

4 OF 4 ▾

Search

<input type="checkbox"/>	Source ⇅	Check ⇅	Output ⇅	⚙️ ⇅	☁️ ⇅	⌚ ⇅	🔄 ⇅
<input type="checkbox"/>	🔊 [blurred]	🔊 check-disk-usage	CheckDisk CRITICAL: /mnt 100.0% bytes usage (3 GiB... 64		sensu_grey	a few seconds ago	an hour ago
<input type="checkbox"/>	🔊 [blurred]	🔊 check-puppet-last-run	PuppetLastRun CRITICAL: Puppet last run 25h 35m 6... 621		sensu_grey	a few seconds ago	a day ago
<input type="checkbox"/>	🔊 [blurred]	🔊 check-puppet-last-run	PuppetLastRun CRITICAL: Puppet last run 25h 30m 0... 615		sensu_grey	a few seconds ago	a day ago
<input type="checkbox"/>	🔊 [blurred]	🔊 check-disk-usage-officehours	CheckDisk CRITICAL: /mnt 100.0% bytes usage (3 GiB... 12		sensu_grey	a minute ago	12 minutes ago

[illegible]

[redacted] 🔊 ✕

🕒 a few seconds ago

☁️ sensu_grey

_id sensu_grey/[redacted]
address [redacted]

keepalive {
 "handlers": [
 "default",
 "hipchat",
],
 "thresholds": {
 "critical": 300,
 "warning": 180
 },
}
name [redacted]

🔴 CHECK-PUPPET-LAST-RUN 🔊 ✕ ✓

alert true
command check-puppet-last-run.rb --summary-file /var/lib/puppet/state/last_run_summary.yaml --warn-age 4500 --crit-age 8100
duration 0.475
executed 2017-01-25 10:53:26
handlers default, hipchat,
interval 60
issued 2017-01-25 10:53:26
name check-puppet-last-run
occurrences 5
output PuppetLastRun CRITICAL: Puppet last run 25h 36m 6s ago
refresh 3600
runbook [Runbook%3A+base](#)
standalone true
status 2
subdue {
 "days": {
 "friday": [
 {
 "begin": "00:00",
 "end": "07:00"
 },
],
 },
}

Package plugins in RPMs: sensu-plugin-builder

- default method uses:
 - `pkg_provider => 'sensu_gem'`
 - `/opt/sensu/embedded/bin/gem install`
- if you don't want `gem install` on production?
- let's build **RPMs** with:
 - [jordansissel/fpm](#)
 - a **docker** container with all build requirements

Package plugins in RPMs: sensu-plugin-builder

- **sensu-plugin-builder**
 - `pkg_provider => 'yum'`
 - hub.docker.com/r/ttarczynski/sensu_plugin_builder

Usage:

```
# docker run -v /tmp:/tmp/ ttarczynski/sensu_plugin_builder \  
  ./build_sensu_plugins.sh /tmp/ sensu-plugins-http
```

Defining checks with **sensu_check_wrapper**

- Common params for checks:
 - occurrences (alert after 5 occurrences)
 - refresh_every (re-alert after 60m)
 - handlers (handle with: hipchat, pagerduty)
 - subdue (don't schedule this check after 3PM)
 - ...
- Hiera lookups for:
 - check
 - check-group
 - default

Defining checks with **sensu_check_wrapper**

- Required params:
 - command
 - **runbook** (on-call documentation)

Defining checks with `sensu_check_wrapper`

Manifest:

```
::sensu_check_wrapper {check-haproxy-redis:  
  command => "check-haproxy.rb --stats localhost -w 33 -c 33 -s 'redis'",  
  runbook => 'http://wiki/Runbook_Haproxy',  
}
```

Hiera:

```
sensu_check_wrapper::occurrences: 5  
sensu_check_wrapper::refresh_every: '60m'  
sensu_check_wrapper::handlers:  
  - 'hipchat'  
  - 'pagerduty'  
Sensu_check_wrapper::aggregate::check-haproxy-redis: 'check-haproxy-redis'
```

Defining checks with **sensu_check_wrapper**

- We've open sourced it:
 - forge.puppet.com/ttarczynski/sensu_check_wrapper
 - forked from: [Yelp/puppet-monitoring_check](https://github.com/Yelp/puppet-monitoring_check)

Avoiding too many alerts: **check-aggregate**

- For HA setups:
 - single-node failure isn't critical
 - alert above $n\%$ (or n) nodes failed

Avoiding too many alerts: **check-aggregate**

- Sample config

```
::sensu_check_wrapper {'aggregate_check-haproxy-redis':  
  command => "check-aggregate.rb --api '${sensu_api_url}' \  
    --check 'check-haproxy-redis' \  
    --warning_count 3 --critical_count 2 --output",  
  runbook => 'http://wiki/Runbook%3A+Haproxy',  
}
```

AGGREGATES > CHECK-HAPROXY-REDIS > OK

CHECK-HAPROXY-REDIS

sensu_grey



1

Check Member



16

Client Member



RESULTS (16)

16

Ok



0

Critical



0

Warning



0

Unknown



✓ CHECK-HAPROXY-REDIS

output
clients

CheckHAProxy OK: UP: 33% of 3 /redis/ services

sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey
sensu_grey

output
client

Resolving on request of the API

sensu_grey

Materials

- [Damian Adamowicz](#): ["Essence of the Cloud"](#)
- [James Turnbull](#): ["Monitoring As A Service"](#)
- [James Turnbull](#): ["The Art of Monitoring"](#)
- [Sean Porter](#): ["Watching the Puppet Show"](#)
- [Tomas Doran](#): ["Sensu and Sensibility"](#)
- [Jos Boumans](#): ["How to measure everything"](#)
- sensuapp.org/docs/
- github.com/sensu-plugins
- github.com/ttarczynski/sensu_plugin_builder
- github.com/ttarczynski/sensu_check_wrapper

Questions?

Tomasz Tarczyński @ttarczynski