

Monitoring

**Claude Falguière
Valtech Paris**



Content

- DevOps is more than tooling

Individuals and interactions over processes and tools

- Make you love Data
- Motivations for providing and collecting data
- Monitoring user stories and practices
- Getting started and open source tooling



Claude Falguiere

- DevOps Coach
- Java, Performance



- Devoxx4Kids
- Paris JUG, Devoxx France, Duchess

<http://cfalguiere.wordpress.com>



Monitoring

What would you do if you knew

- that database is broken
- that number of hits doubles every 2 month
- that users struggle to find the order form
- why the app is slow
- what users want to buy



Model

Questions

Hypothesis

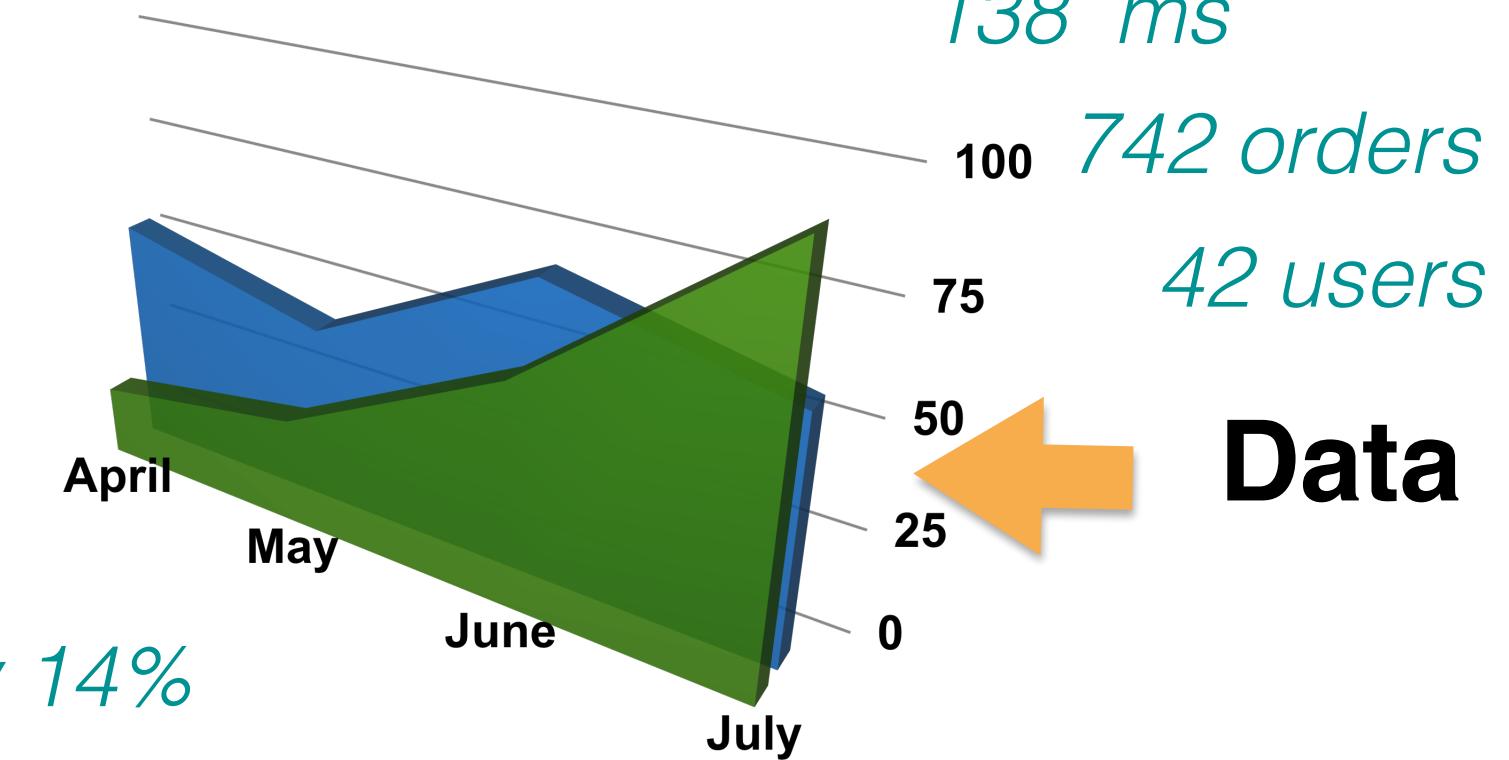
Facts

Sales increased by 14%

Estimated orders next month 934

*Average number of requests is 5 times
the number of users*

Model



Data

138 ms
742 orders
42 users

Galaxy Rotation Problem

Spiral galaxies spin too fast

Expected mass should be ten times the observed mass - calculated from the visible objects - to prevent galaxies from flying apart





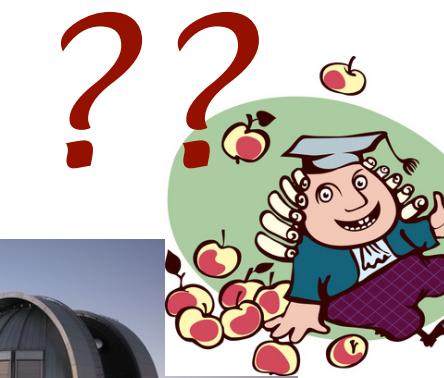
Discovery of Dark Matter



Assumes readings are wrong



If readings are true, is model wrong ?



Jan Oort Fritz Zwicky



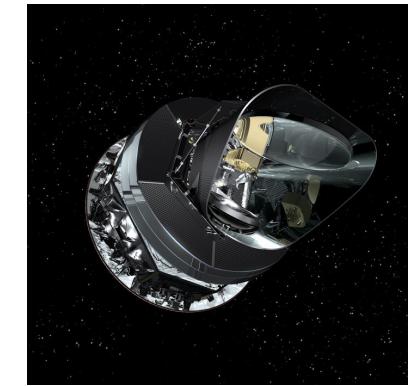
Hypothesis of a missing mass

1932 - 1933

1960 - 1970

2010 - 2013

Plank Satellite



Mass calculated from gravitational effects and evidence of Dark Matter

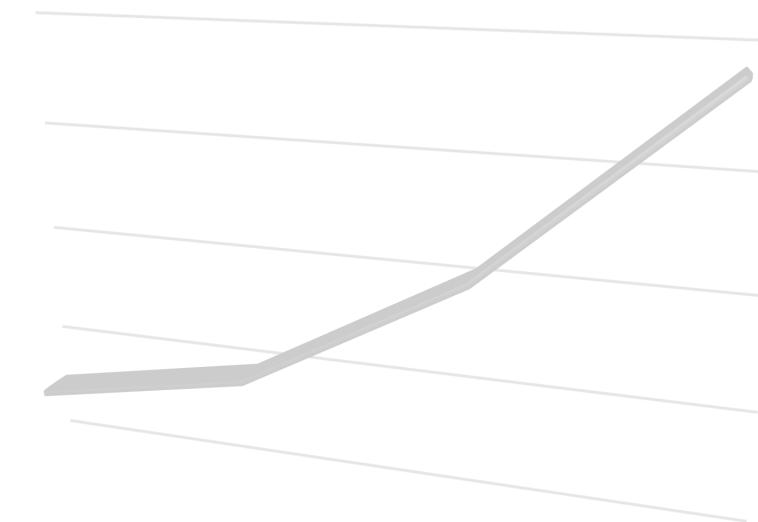
Vera Rubin



Dark matter estimated to 84.5% of the total matter in the universe

Measure everything

Lean Startup



**Make decisions
based on facts**

Big Data

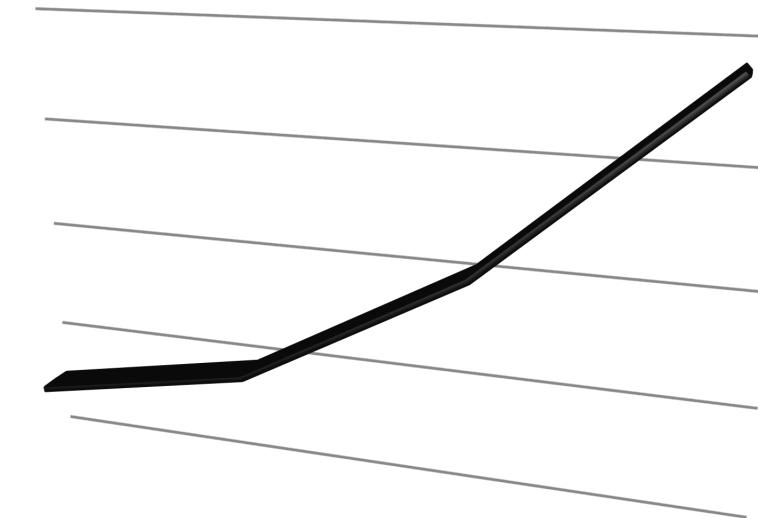


DevOps



Measure everything

Lean Startup



**Make decisions
based on facts**

Big Data



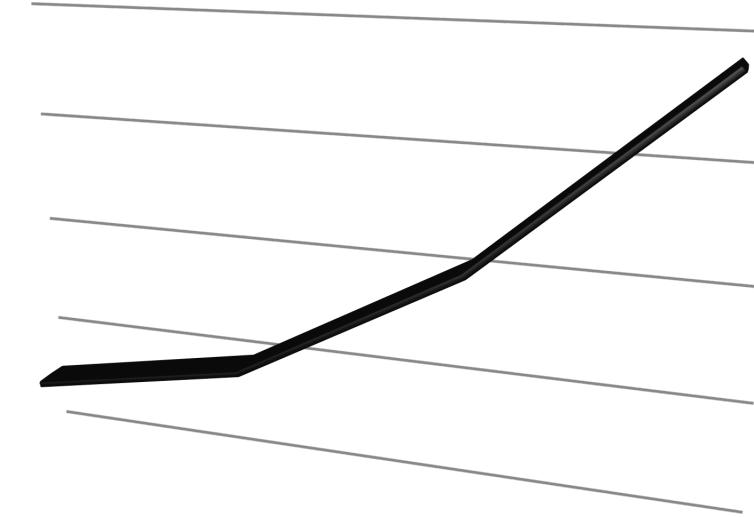
DevOps



TM
X
O
V
D
E
W

Measure everything

Lean Startup



**Make decisions
based on facts**

Big Data

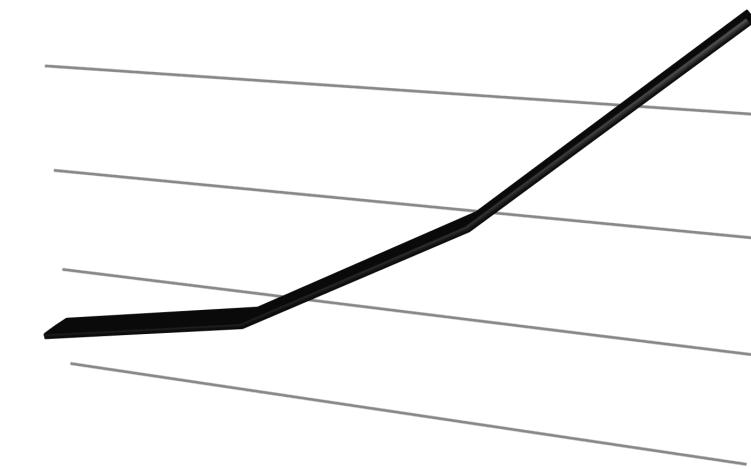


DevOps



Measure everything

Lean Startup

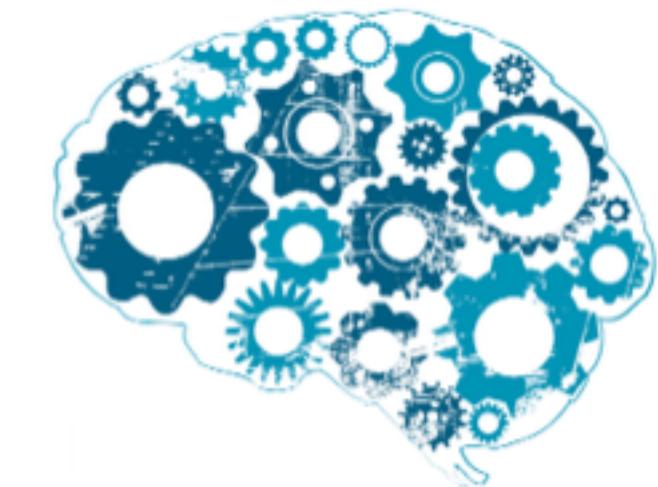


**Make decisions
based on facts**

Big Data



DevOps





What would you do if you knew

that database is broken

that number of hits doubles every 2 month

that users struggle to find the order form

why the app is slow

what users want to buy

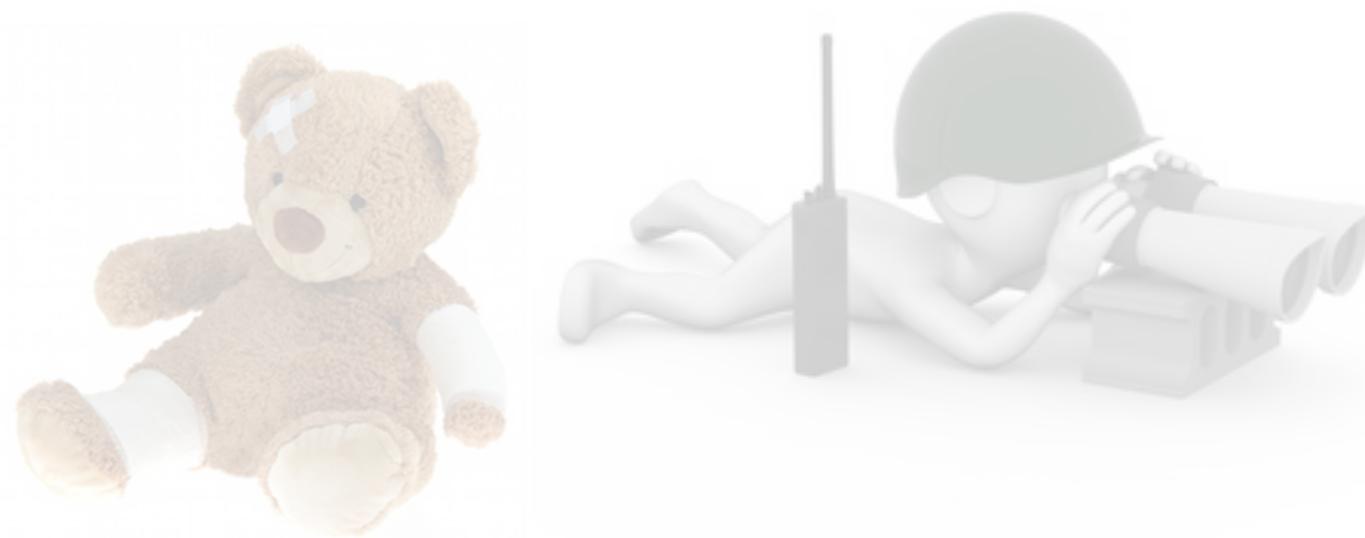


Motivations and user stories

SLA observance
Alerting



Diagnosis / Post-Mortem
Capacity Planning
Improvement



Alerting

Storage, Visualization



Motivations and user stories

SLA observance
Alerting



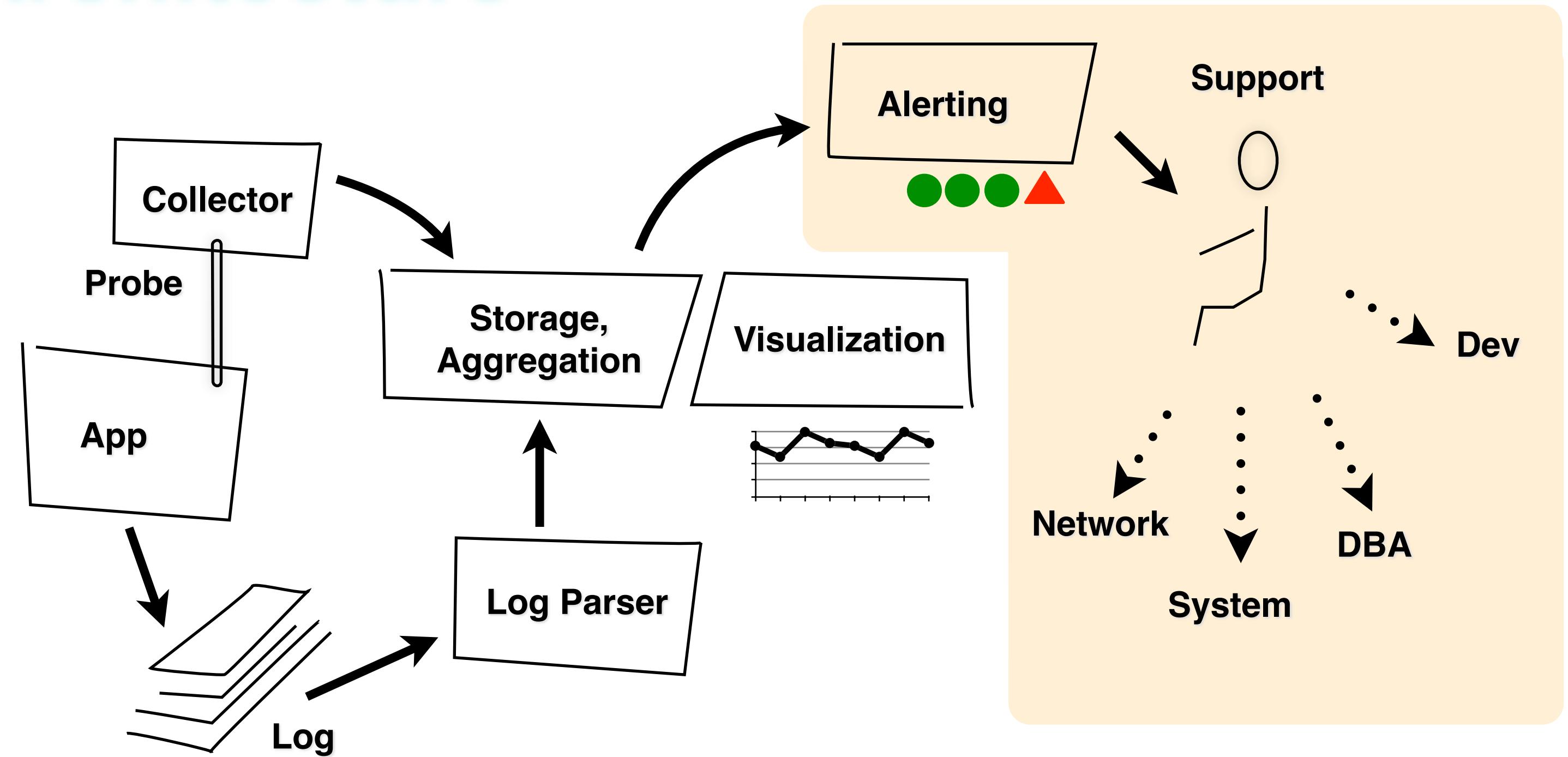
Alerting

Diagnosis / Post-Mortem
Capacity Planning
Improvement

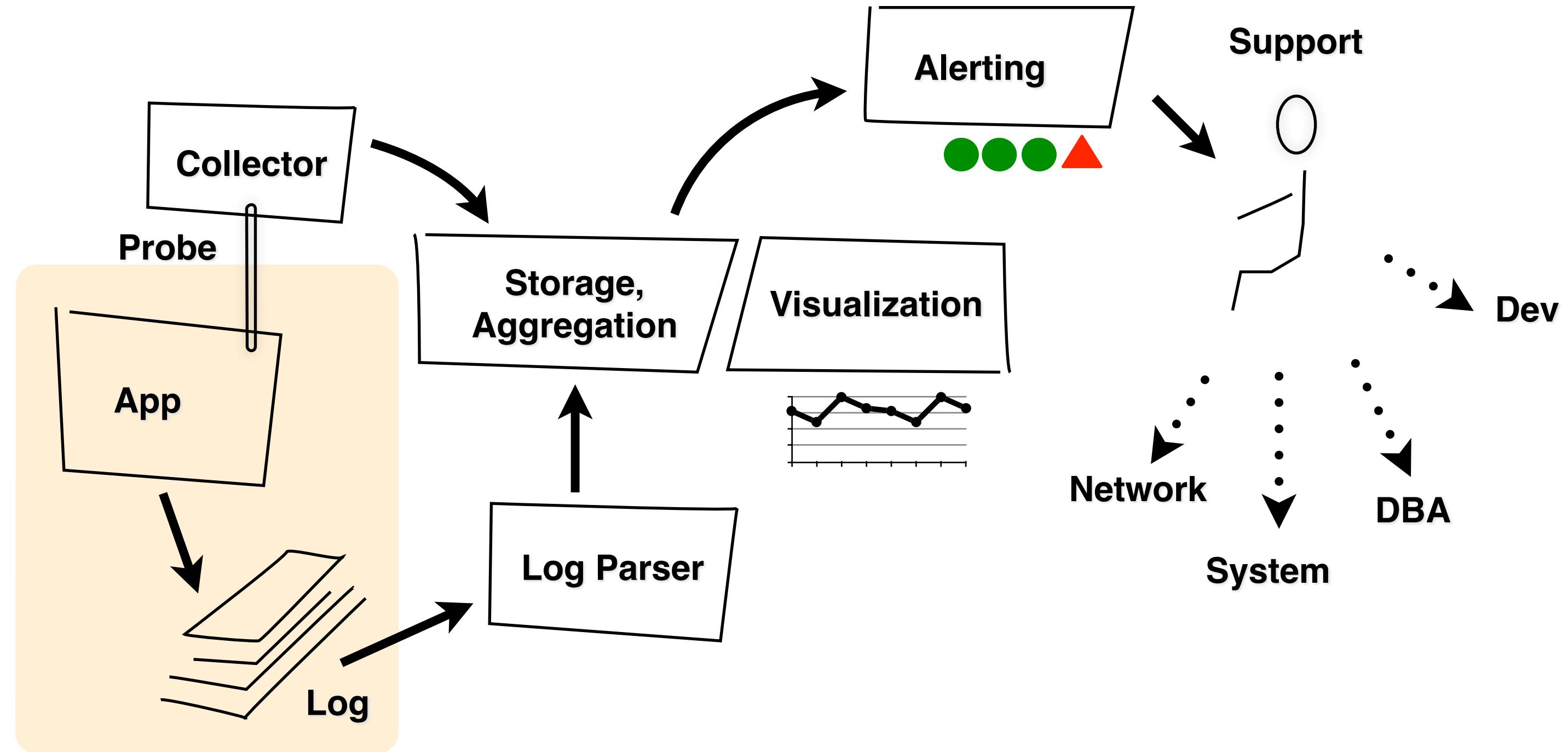


Storage, Visualization

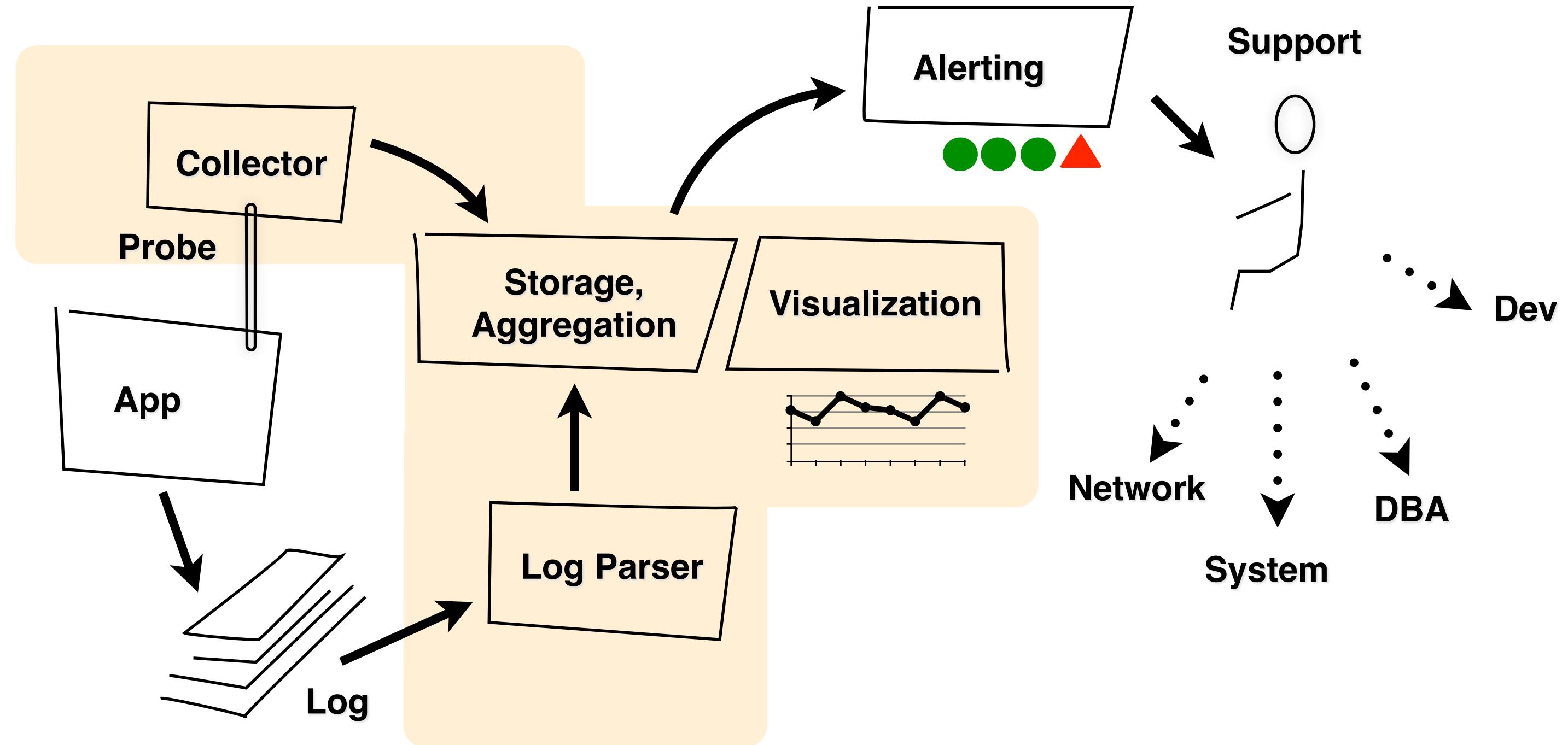
Architecture



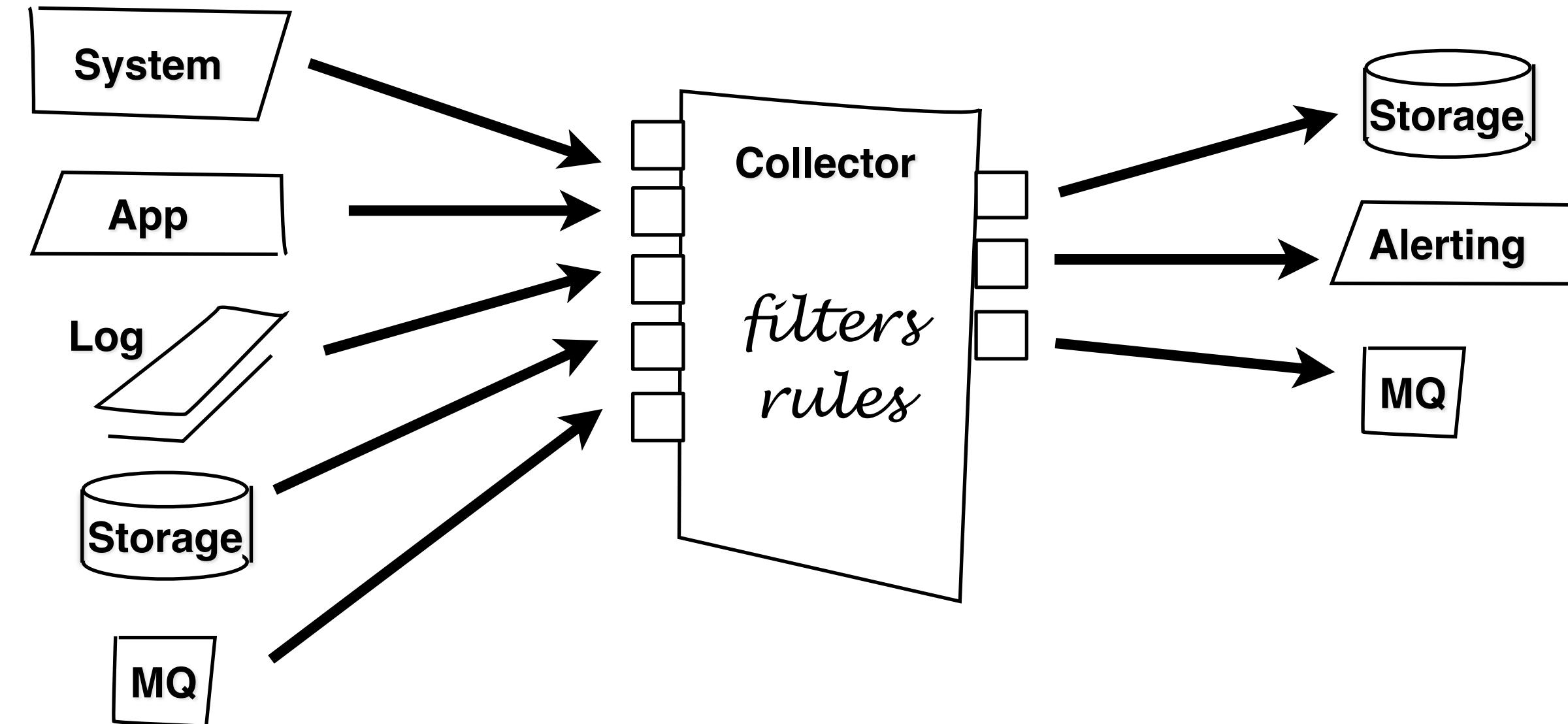
Architecture



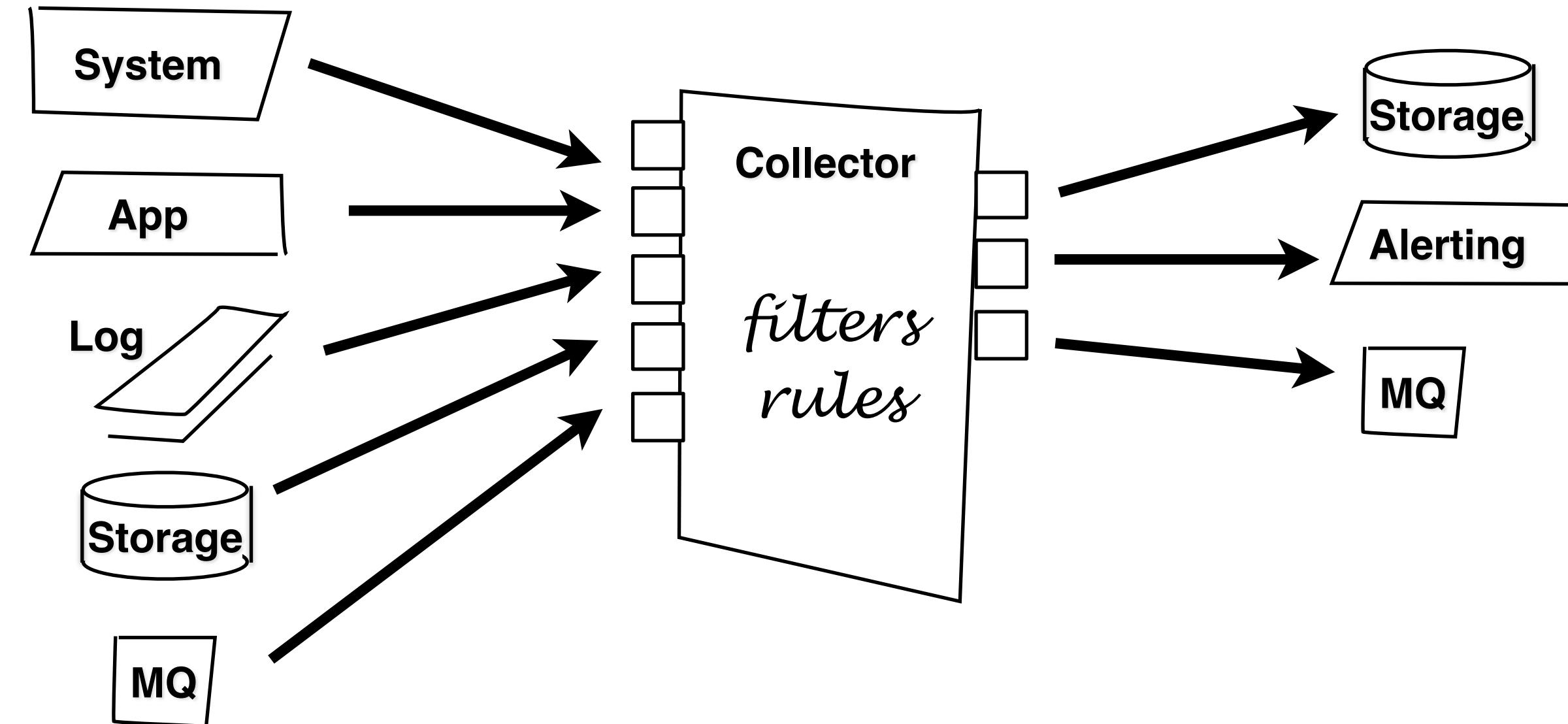
Architecture



TM XX Collector

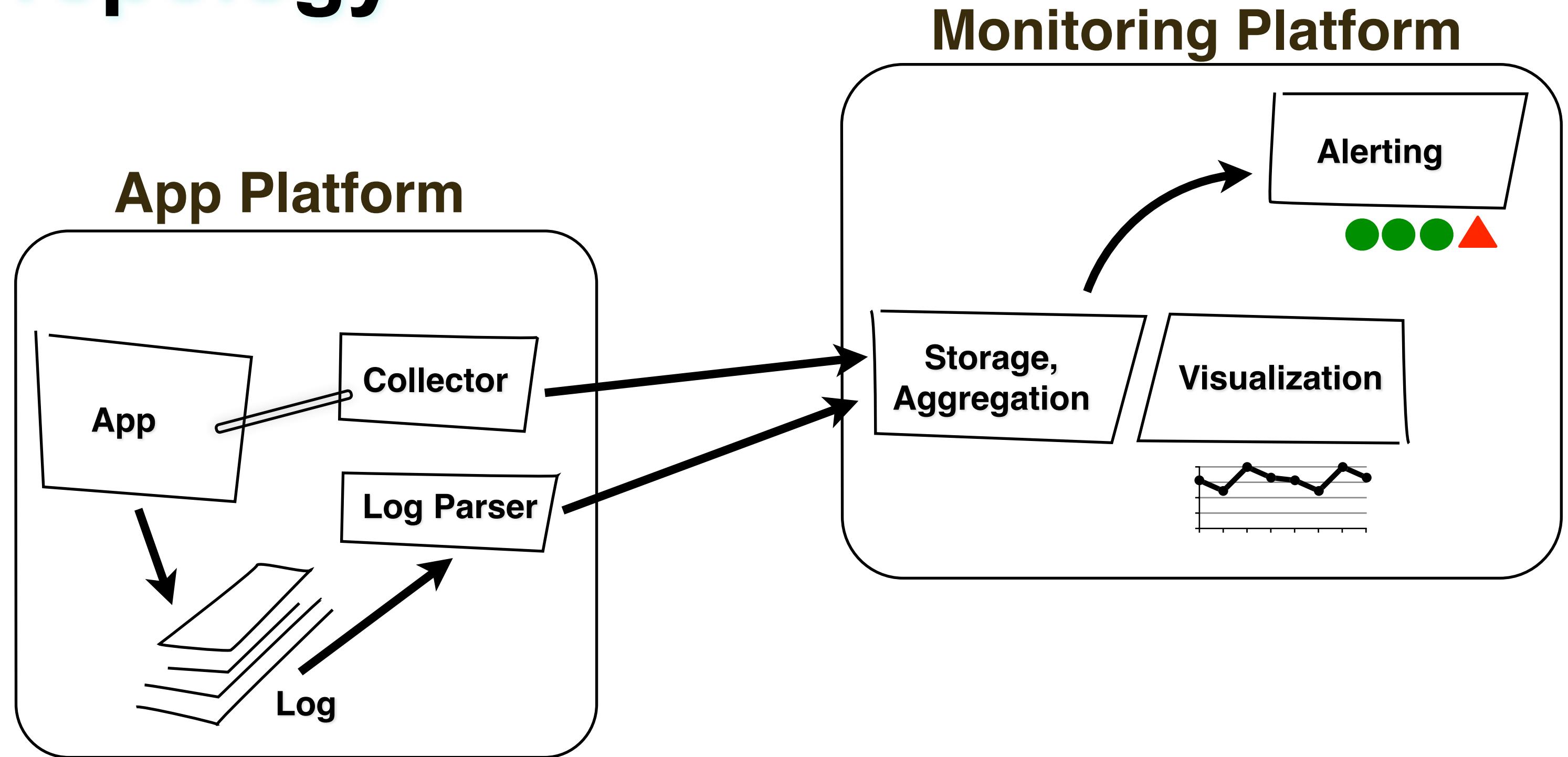


TM XX Collector



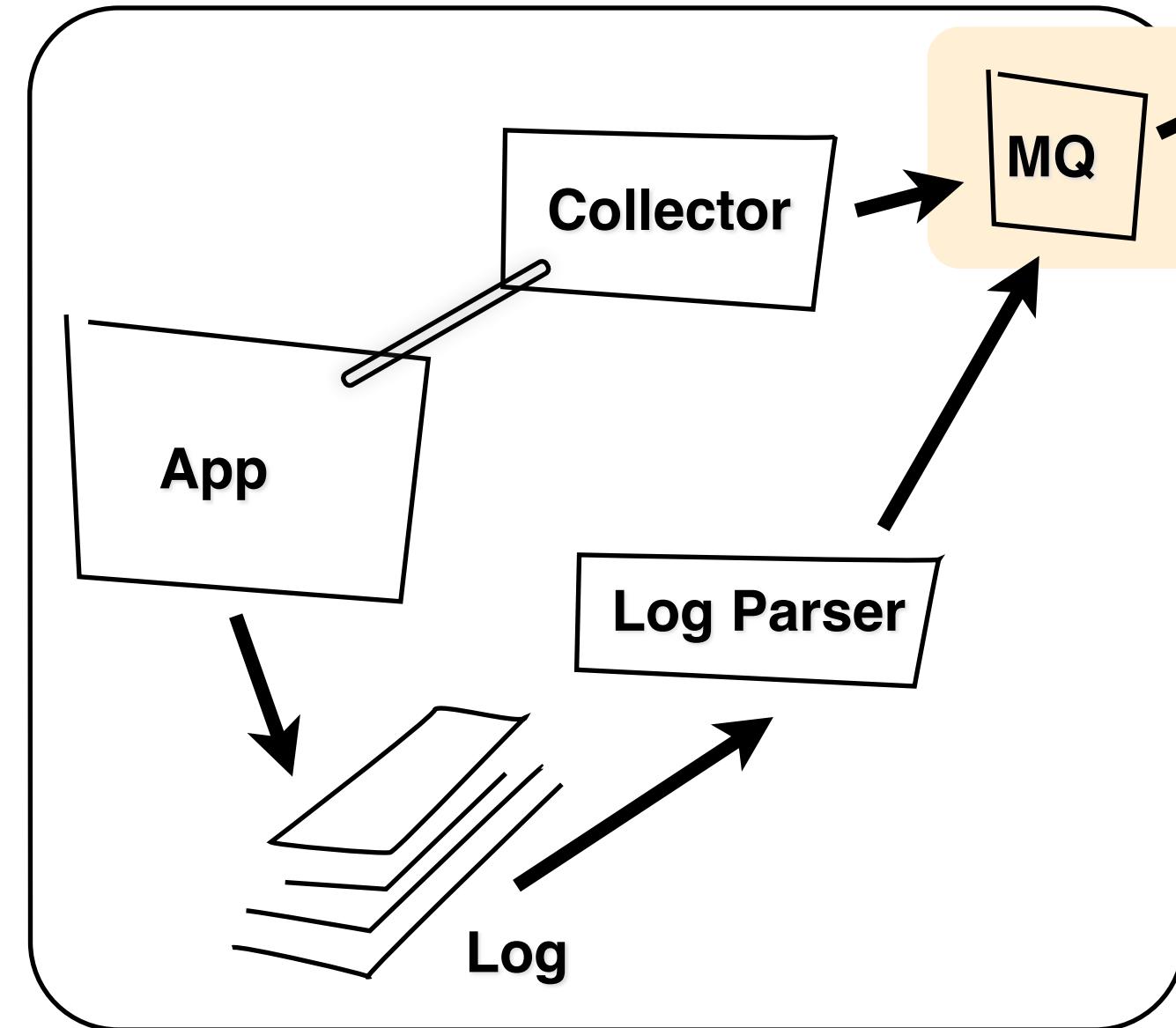
TM

Topology

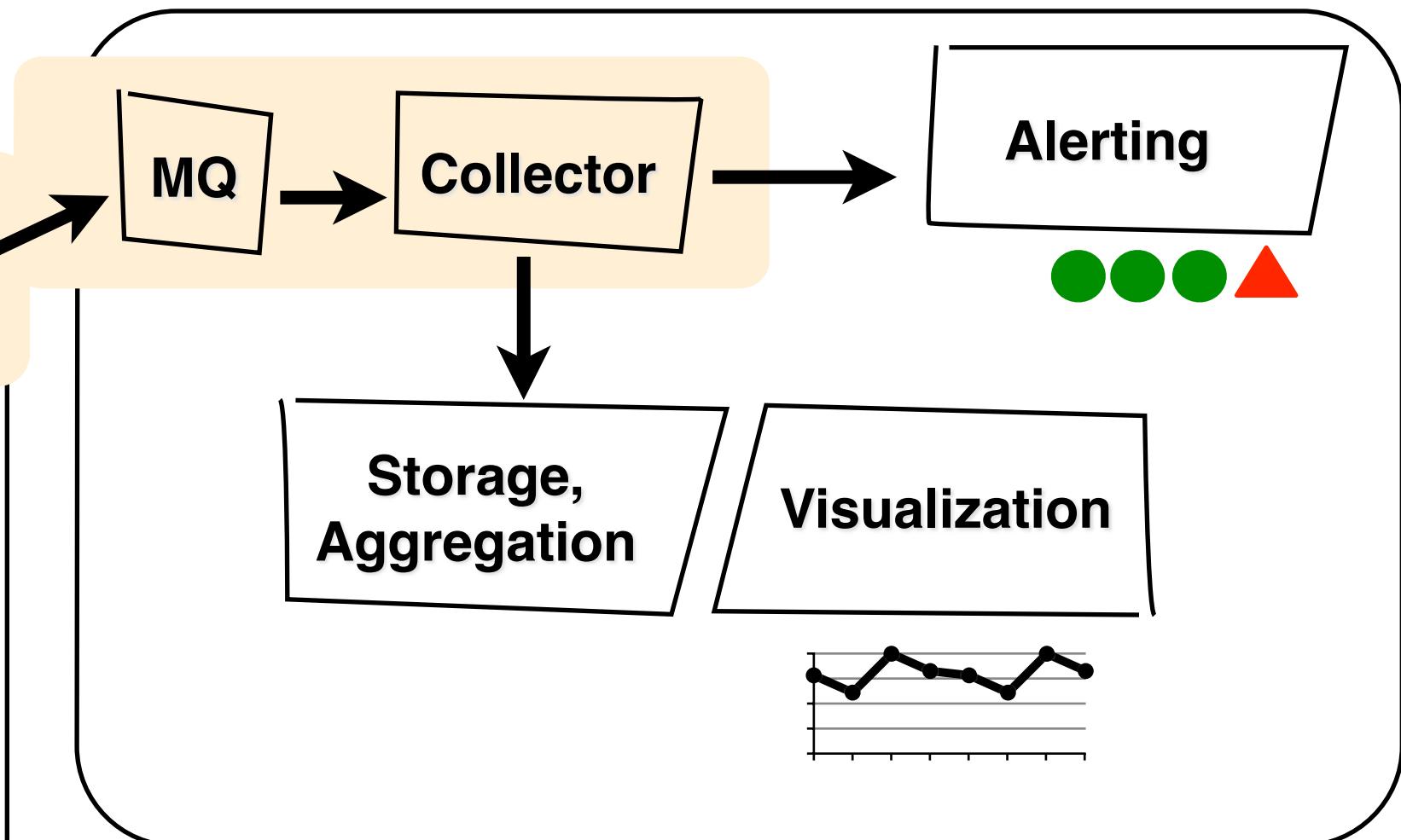


Resilience

App Platform



Monitoring Platform



What would you do if you knew that database is broken





Error detection and alerting

- Log filtering
- Event firing
- Context
 - is it critical ?
 - which feature does it impact ?
 - how deep is the impact ?



Is this a log ?

```
Exception in thread "main" com.mysql.jdbc.exceptions.jdbc4.MySQLSyntaxErrorException:  
Access denied for user 'shopapp'@'shprdb1' to database 'shop'  
at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)  
at sun.reflect.NativeConstructorAccessorImpl.newInstance(Unknown Source)  
at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(Unknown Source)  
at java.lang.reflect.Constructor.newInstance(Unknown Source)  
at com.mysql.jdbc.Util.handleNewInstance(Util.java:411)  
at com.mysql.jdbc.Util.getInstance(Util.java:386)  
at com.mysql.jdbc.SQLException.createSQLException(SQLException.java:1054)  
at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:4237)  
at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:4169)  
at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:928)  
at com.mysql.jdbc.MysqlIO.proceedHandshakeWithPluggableAuthentication(MysqlIO.java:1750)  
at com.mysql.jdbc.MysqlIO.doHandshake(MysqlIO.java:1290)  
at com.mysql.jdbc.ConnectionImpl.coreConnect(ConnectionImpl.java:2493)  
at com.mysql.jdbc.ConnectionImpl.connectOneTryOnly(ConnectionImpl.java:2526)  
at com.mysql.jdbc.ConnectionImpl.createNewIO(ConnectionImpl.java:2311)  
at com.mysql.jdbc.ConnectionImpl.<init>(ConnectionImpl.java:834)  
at com.mysql.jdbc.JDBC4Connection.<init>(JDBC4Connection.java:47)  
at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)  
at sun.reflect.NativeConstructorAccessorImpl.newInstance(Unknown Source)  
at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(Unknown Source)  
at java.lang.reflect.Constructor.newInstance(Unknown Source)  
at com.mysql.jdbc.Util.handleNewInstance(Util.java:411)  
at com.mysql.jdbc.ConnectionImpl.getInstance(ConnectionImpl.java:416)  
at com.mysql.jdbc.NonRegisteringDriver.connect(NonRegisteringDriver.java:347)  
at java.sql.DriverManager.getConnection(Unknown Source)
```

Log example

2013-12-17 05:53:16,208 ERROR [Order Creation Service](456713)
[shpras2](web1234) Could not create order id=456713 - Cause:
Can't connect to database 'shop' - MySqlMessage: Access denied
for user 'shopapp'@'shprdb1' to database 'shop'



2013-12-17 05:53:16,208
ERROR
[Order Creation Service]
(456713)
[shpras2]
(web1234)
Could not create order id=456713
Cause: Can't connect to database 'shop'
MySqlMessage: Access denied for user 'shopapp'@'shprdb1' to database 'shop'



Severity

Timestamp
2013-12-17 05:53:16,208

Severity
ERROR

[Order Creation Service]
(456713)
[shpras2]
(web1234)

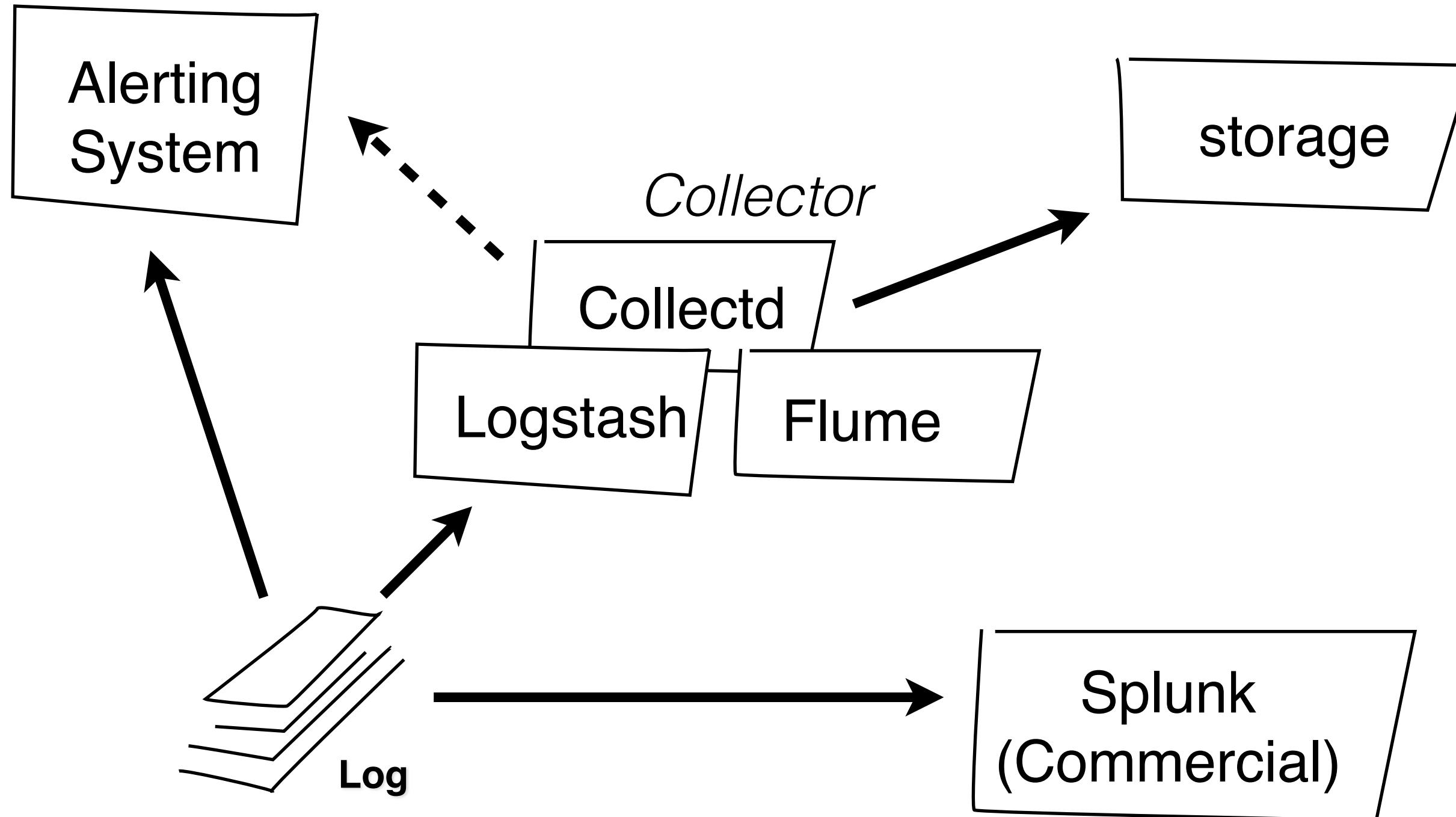
Could not create order id=456713
Cause: Can't connect to database 'shop'
MySqlMessage: Access denied for user
'shopapp'@'shprdb1' to database 'shop'

Meaningful information



Context (technical and business)

Log Collectors





Logstash

```
input {  
  file {  
    path => "/app/logs/apache/*.log"  
    type => "apachelog"  
  }  
}  
  
filter {  
  if [type] == "apachelog" {  
    grok {  
      pattern => "%{COMBINEDAPACHELOG}"  
    }  
  }  
}  
  
output {  
  elasticsearch { host => localhost }  
  stdout { }  
}
```



Logstash

```
input {  
  file {  
    path => "/app/logs/appserver/monitor*.log"  
    type => "applog"  
  }  
}  
  
filter {  
  if [type] == "applog" {  
    grok {  
      pattern => "%{TIMESTAMP_ISO8601:ts} %{WORD:severity} ...  
    }  
  }  
}  
  
output {  
  elasticsearch { host => localhost }  
  stdout { }  
}
```

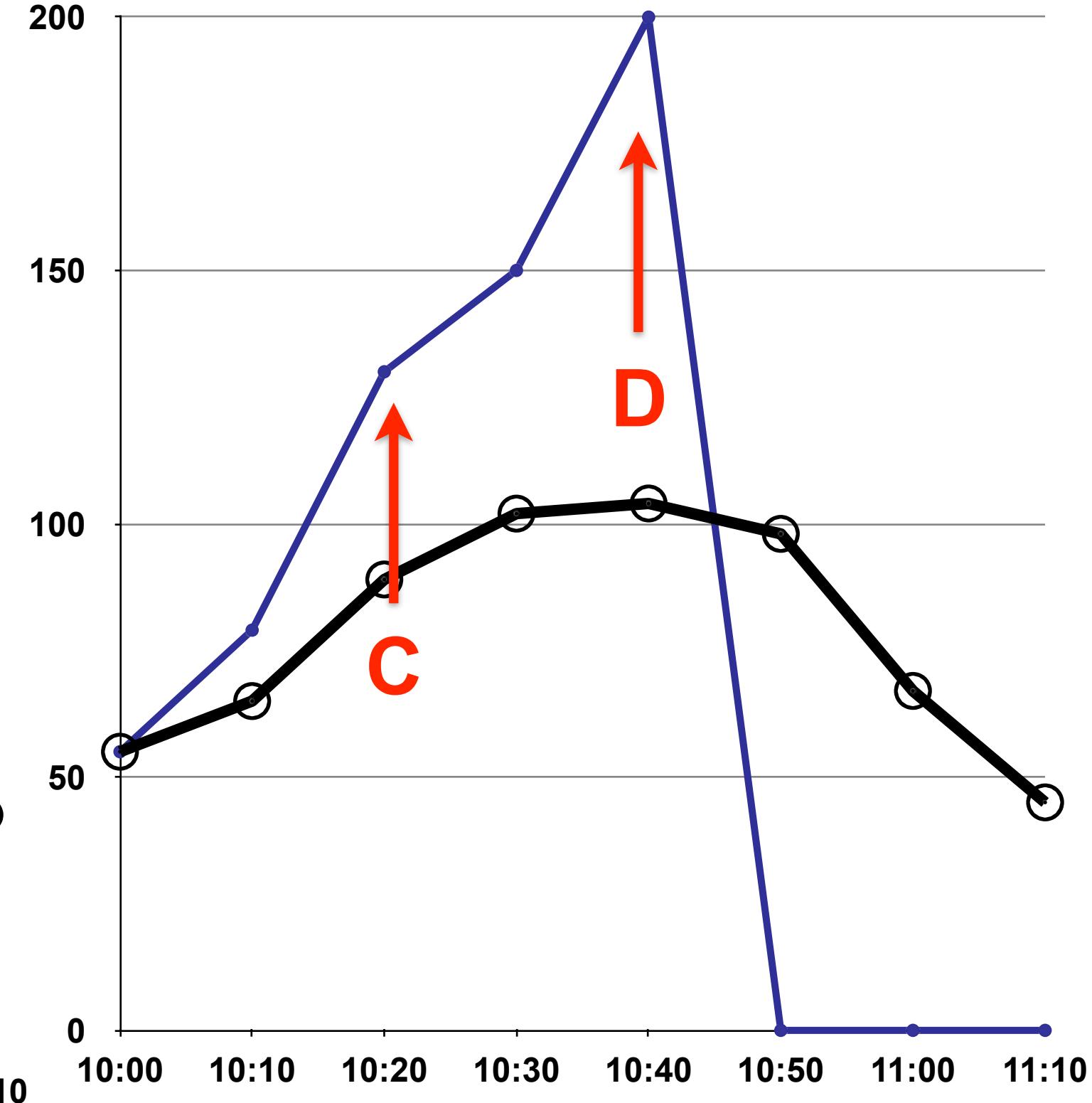
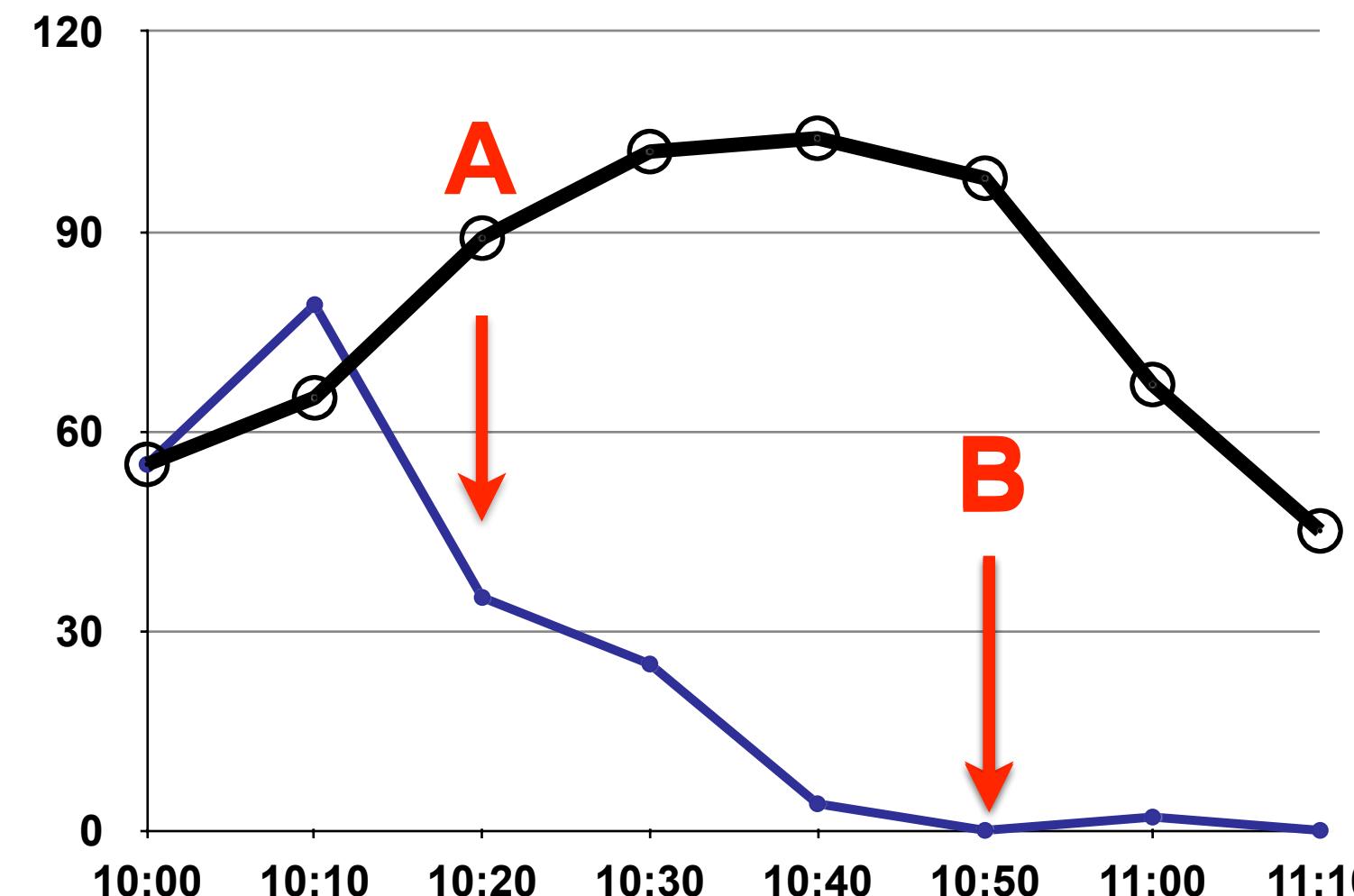


Rate check

- Frequency of an error increases
- Activity falls (e.g. Frequency of orders)
- Alerting based on threshold

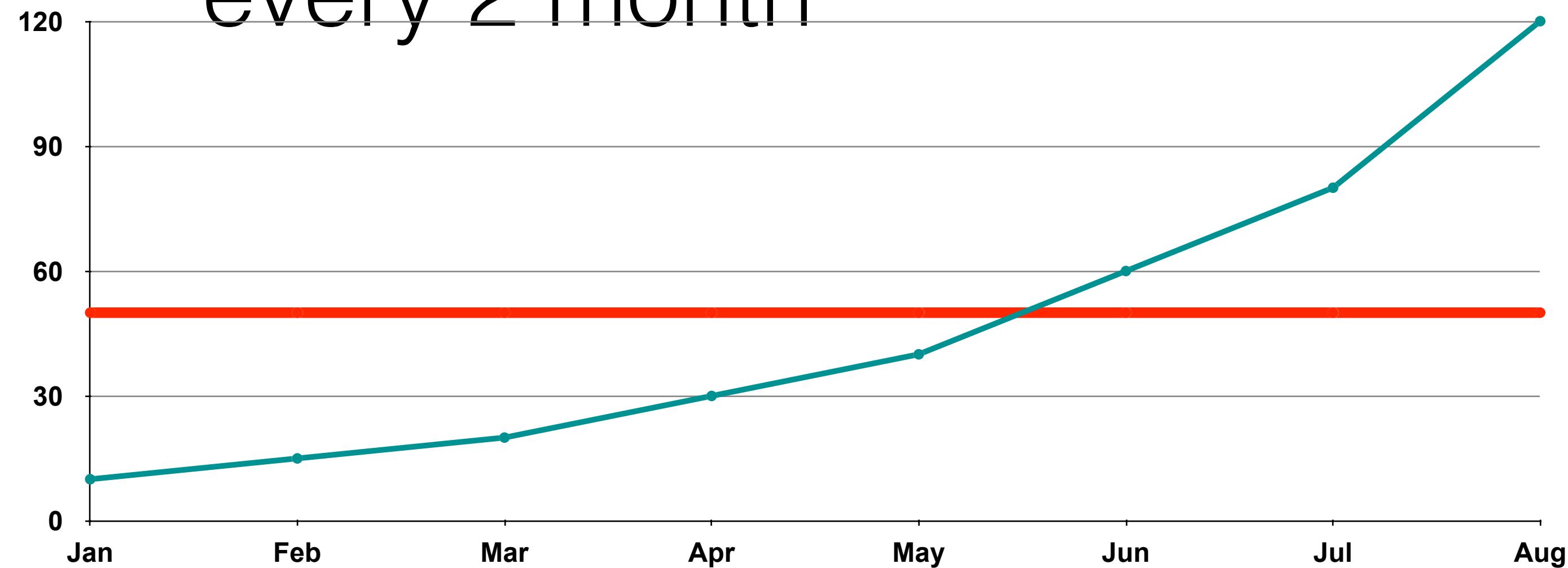


Baselining





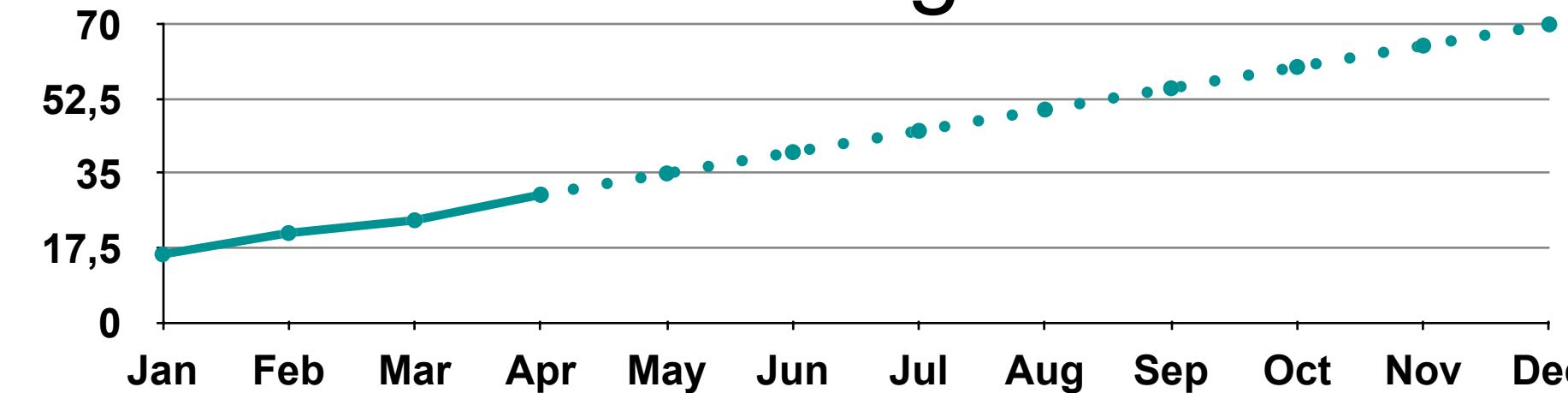
What would you do if you knew that
number of hits doubles
every 2 month



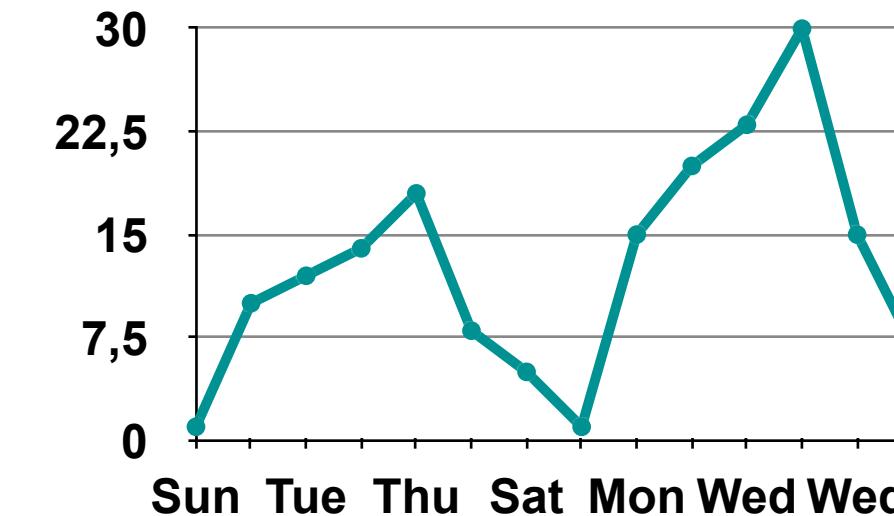


Graphers

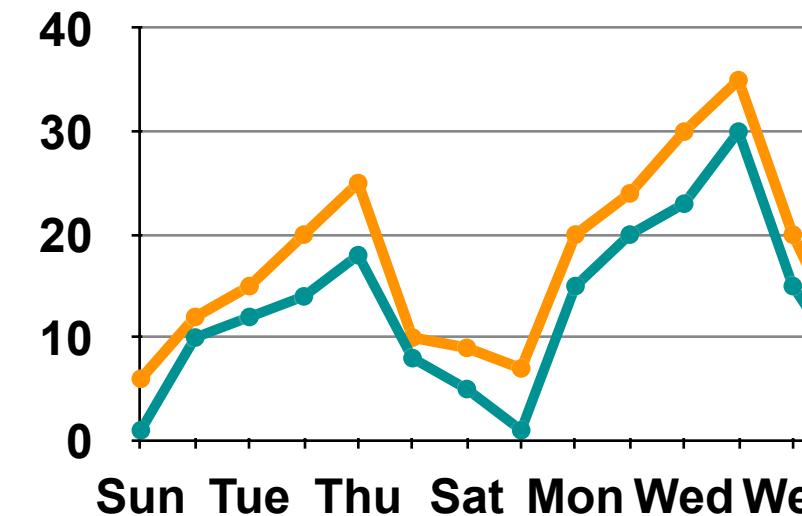
- Foresight



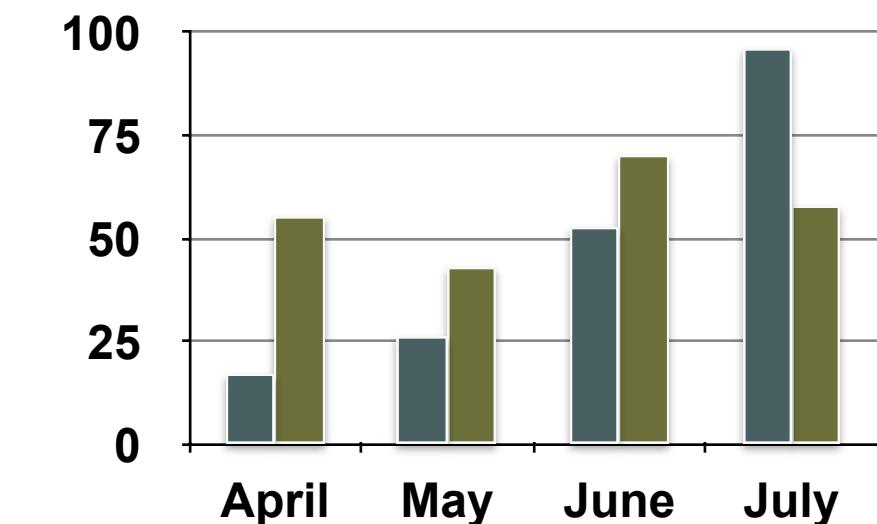
- Cycles



- Correlation

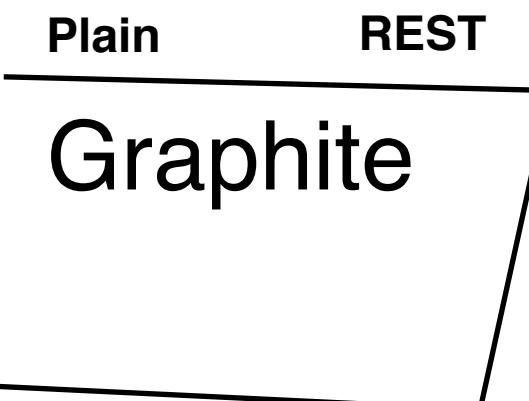


- Distribution



Storage / Visualization

Collectors (Collectd / Statd / Logstash / Flume)



docker: lopter/collectd-graphite



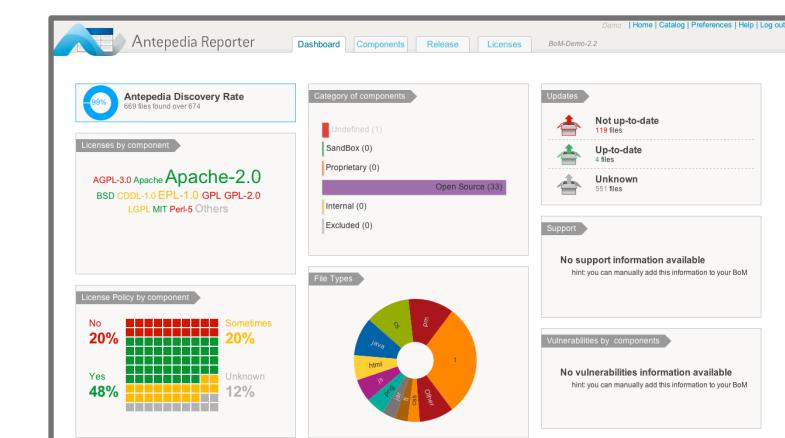


Collect and Share

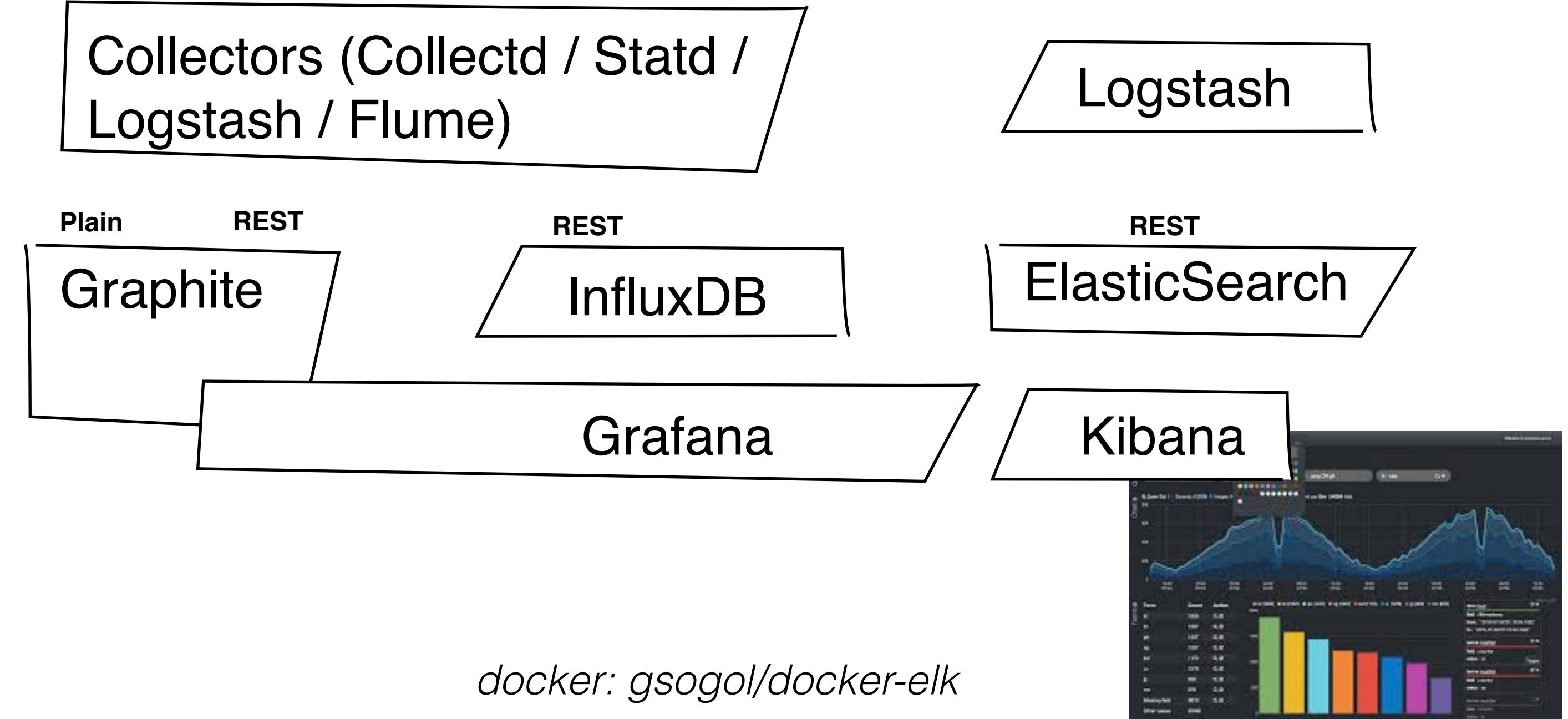
Collect Once and Share

- Support,
- Ops, Dev
- Business

UpToDate Flexible

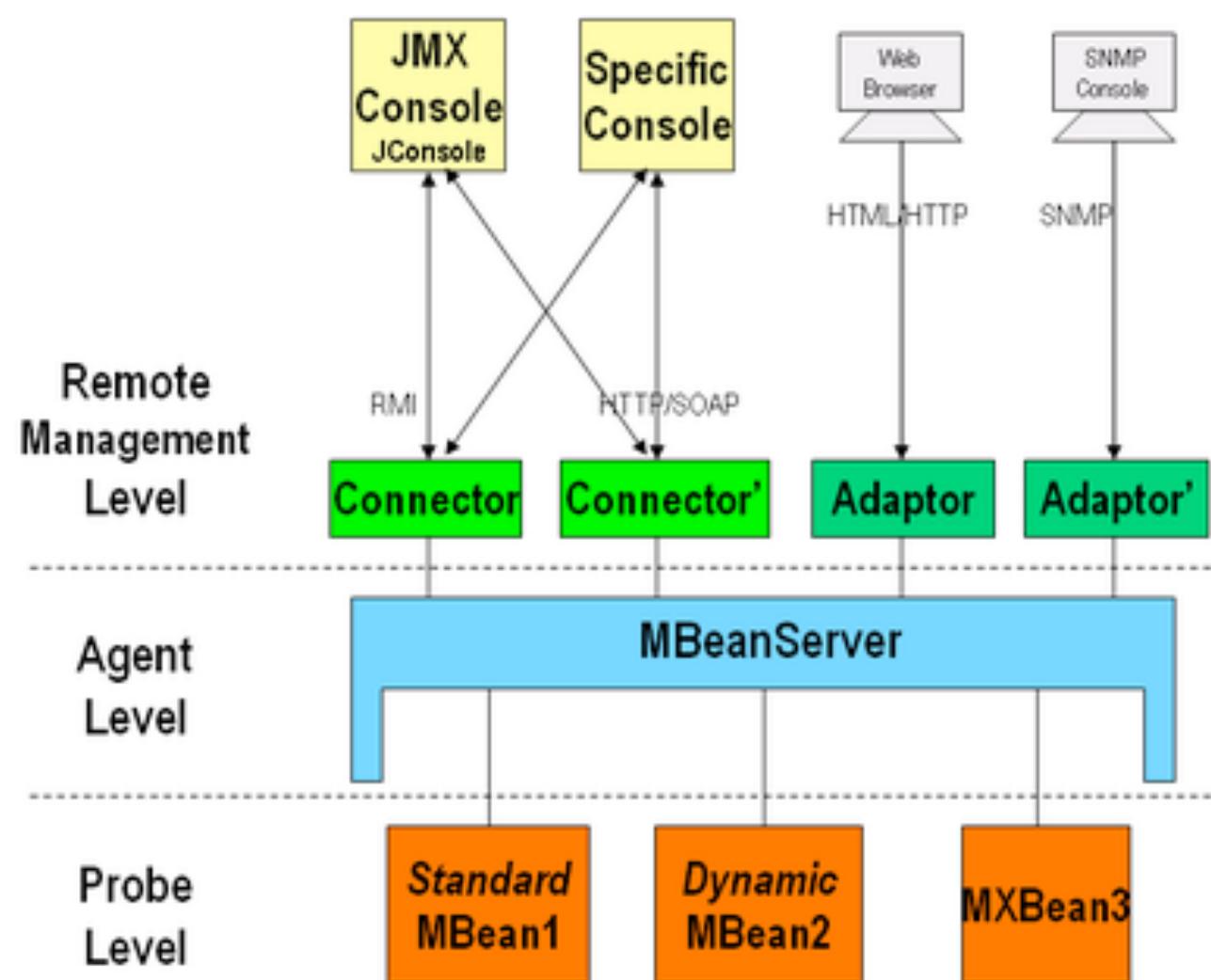


Storage / Visualization



TM

JMX



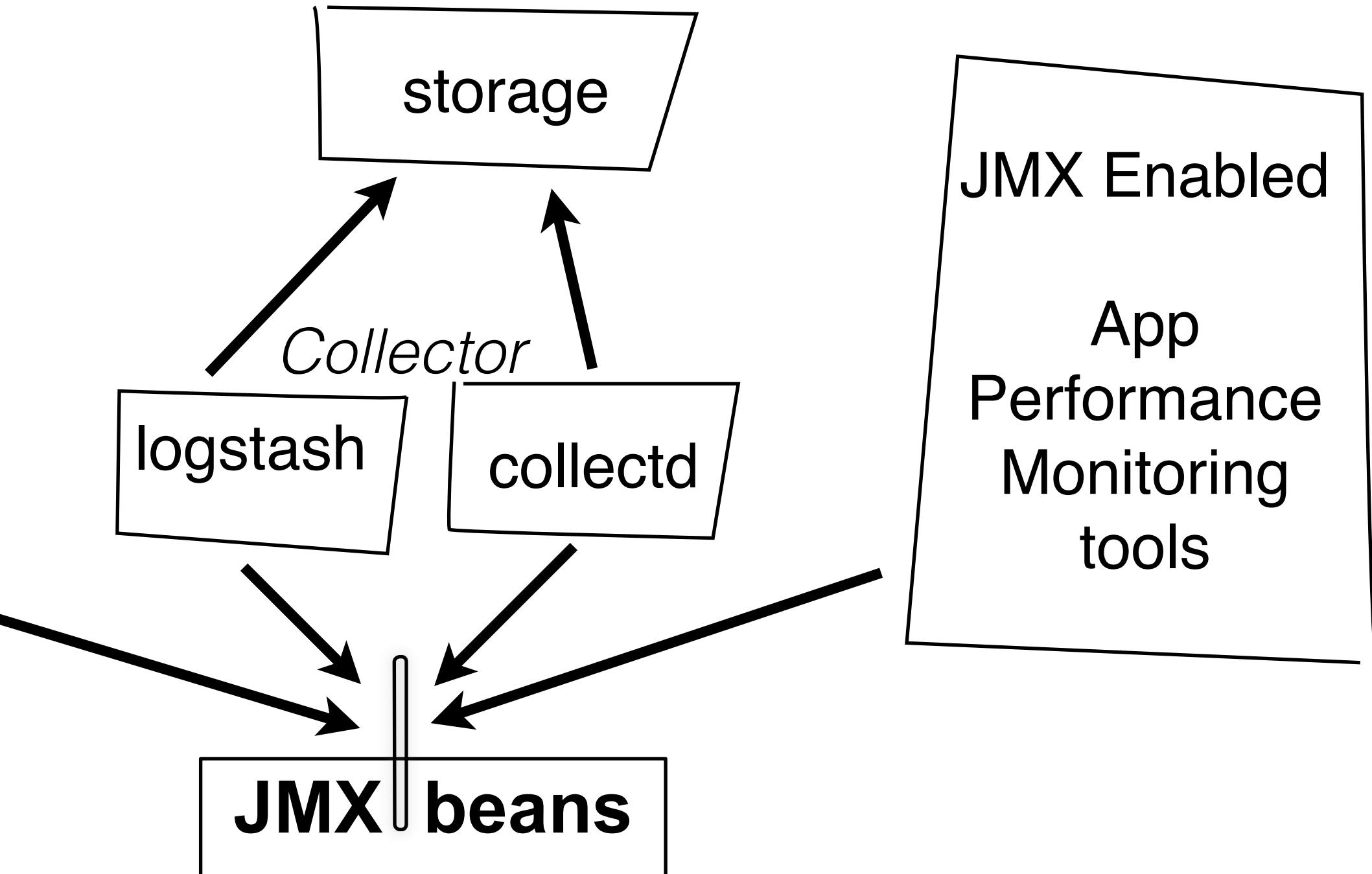
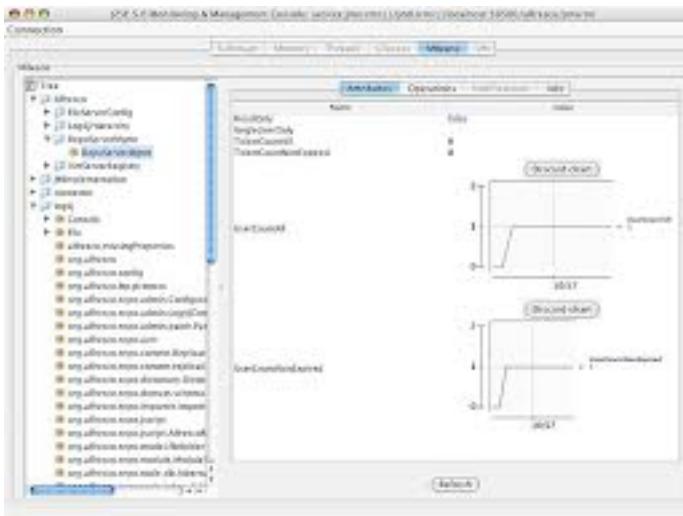
source: *wikipedia*

- MBeans
- Registration
- Servo
- RMI and firewalls
 - -Dcom.sun.management.jmxremote.rmi.port=p
 - -Djava.rmi.server.hostname=n.n.n.n
- Jolokia
- jmxtrans



JMX Collectors

VisualVM
JConsole



JSON Event over REST

```
curl -X POST “...”
```

Timestamp

```
  -d '{"ts": "2013-12-17 05:53:16,208",  
    "type": "metric",  
    "module": "Order Creation Service",  
    "module-id": "456713",  
    "instance": "shpras2",  
    "thread": "web1234",  
    "name": "order-creation",  
    "duration": "12",  
    "unit": "ms"}'
```

} Metric)

} Context (technical and business)

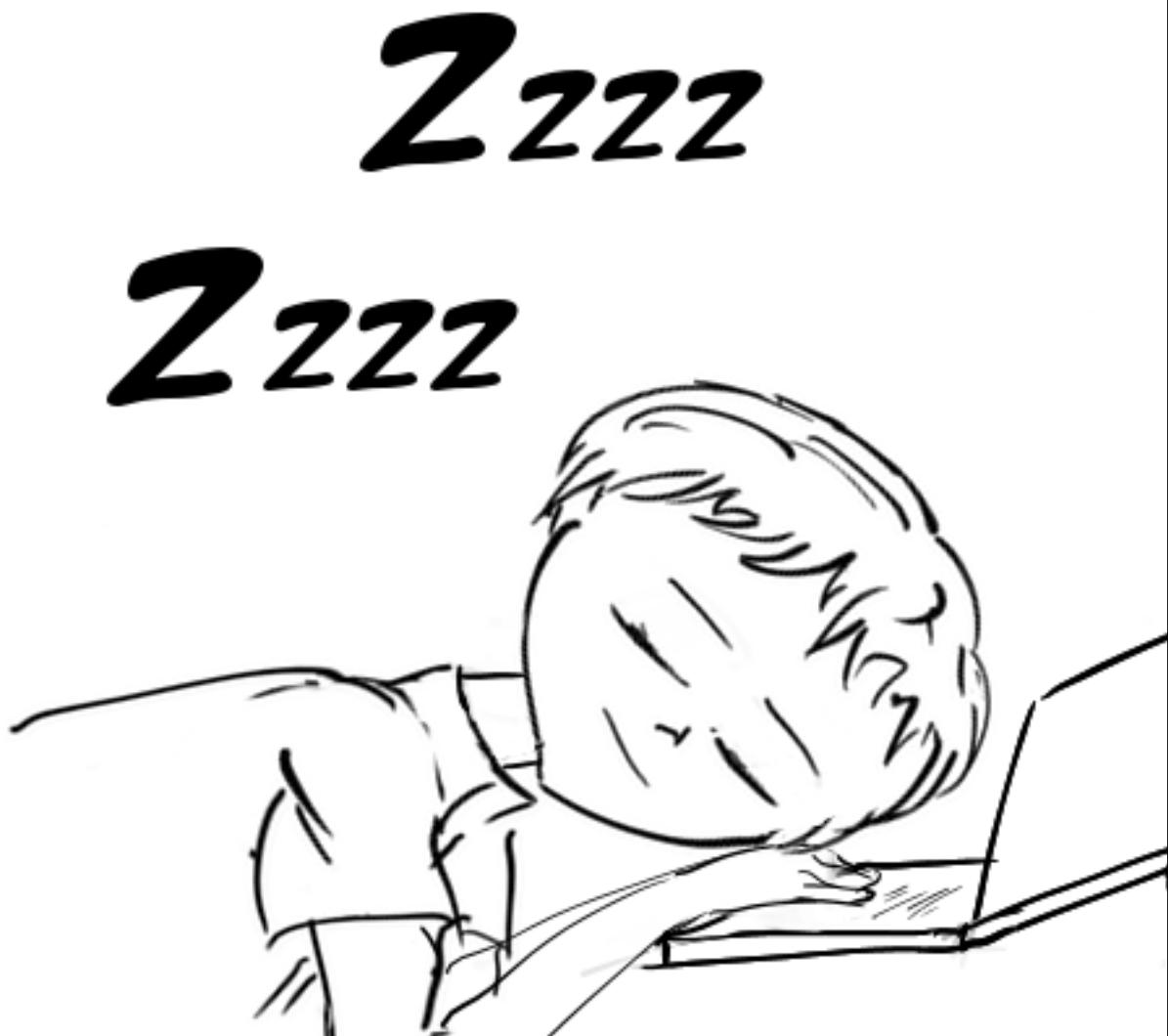


TM

X
X
O
>
S
D

What would you do if you knew

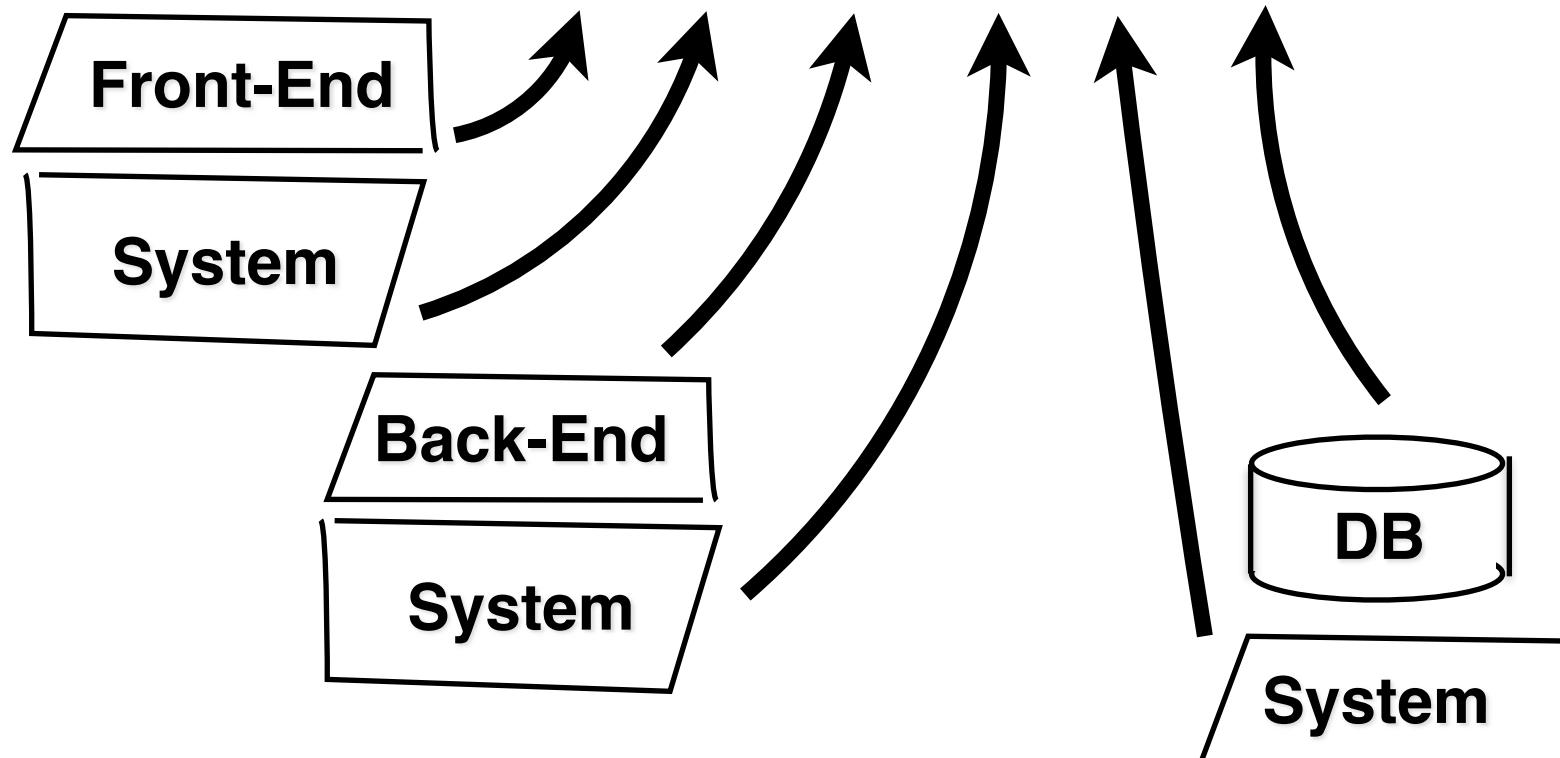
why app is slow





Tuning

cross-check metrics from various sub-systems



- Collectd/Stadt plugins
- Metrics
- Commercial : Plumbr, AppDynamics, New Relics

Where does it spend time ?
Why ?

TM



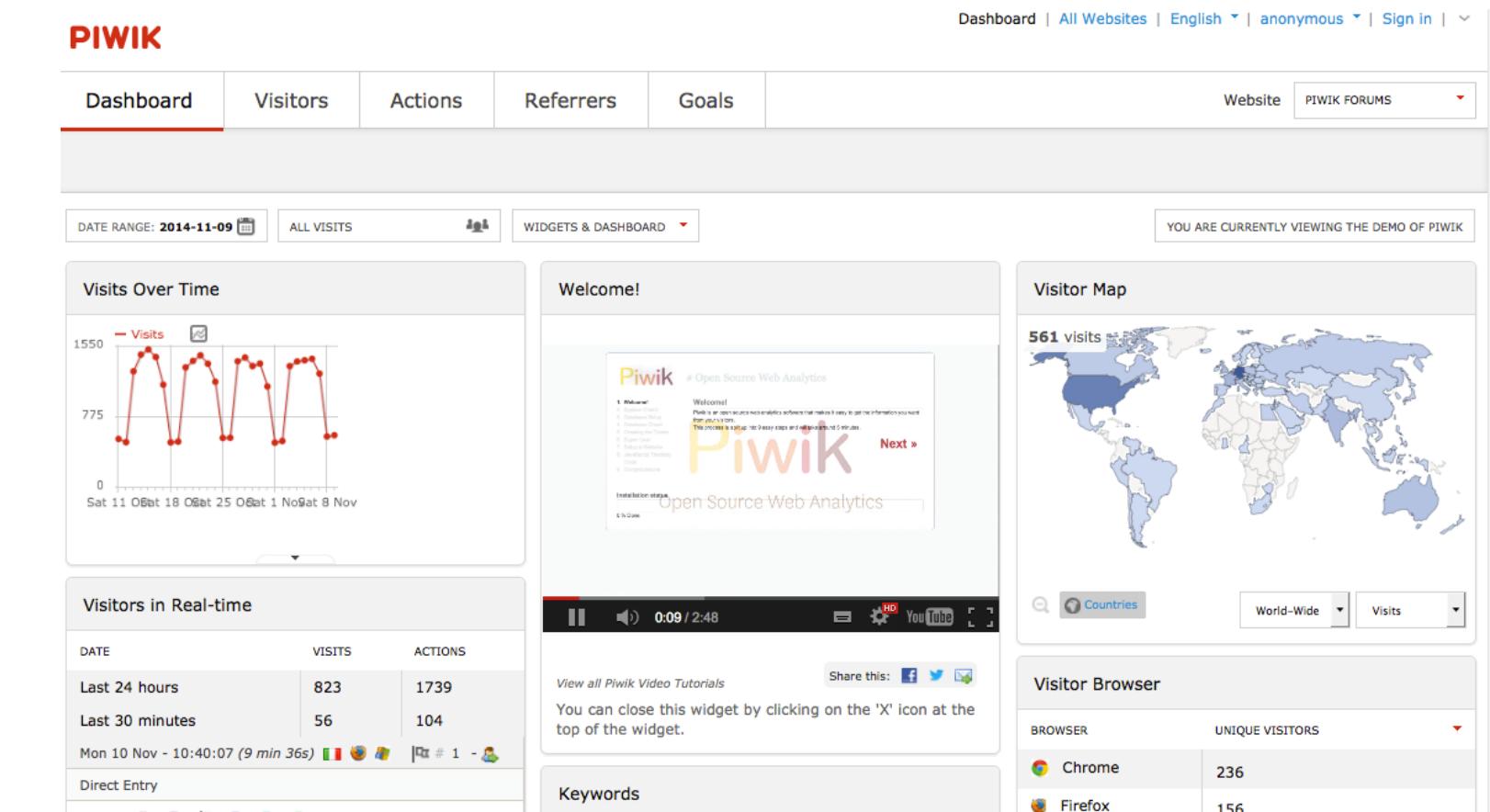
What would you do if you knew that users struggle to find the order form





Web Analytics / User tracking

- Web analytics
 - Page counters
 - Tagging
 - Log parser
- Google Analytics
- Piwik (*docker: cfalguiere/docker-piwik*)
- Reporting APIs



What would you do if you knew what users want to buy



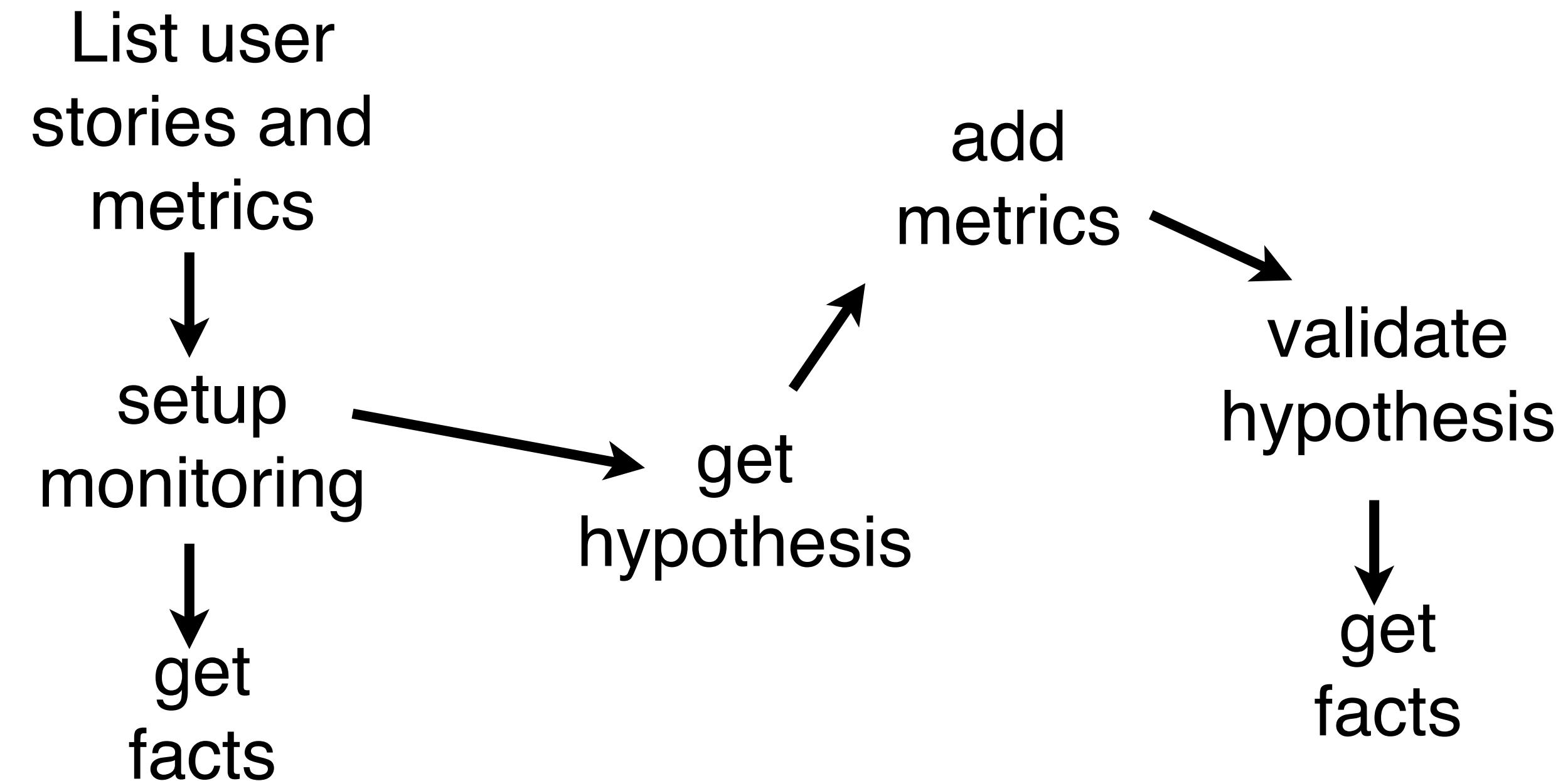
Model vs Big Data

- Expected information
 - Explicit Model
 - List of metrics
- Classification
 - Machine Learning
 - Patterns detection



Highlights valuable metrics and relationships

Getting started





What should I monitor ?

Alerting & Post-Mortem :

Presence check

Activity (how many users, requests, orders ...)

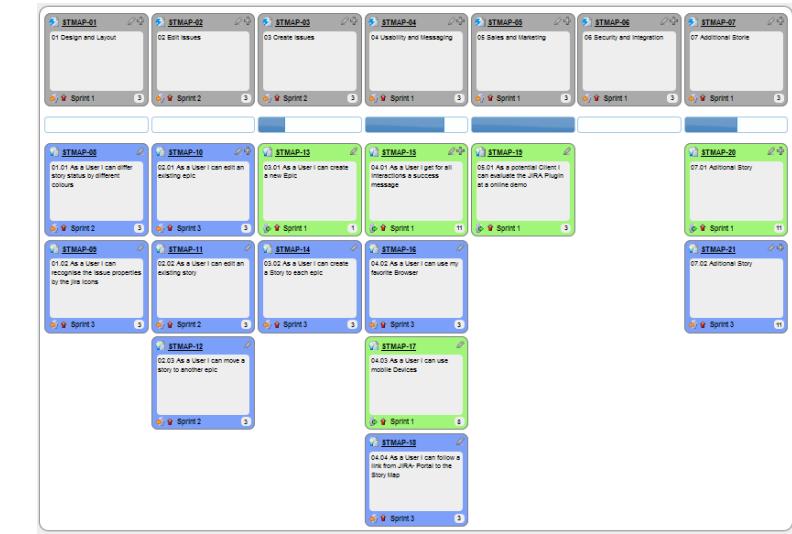
Ressources that are limited in size

Physical : CPU, memory, free disk space, network bandwidth ...

Logical : pools, queues, caches, ...

Errors

Others





What should I monitor ?

Plan & Improve :

Any information which is useful to understand the process

time spent for each major step

things that are done often or requires large datasets

user navigation

context

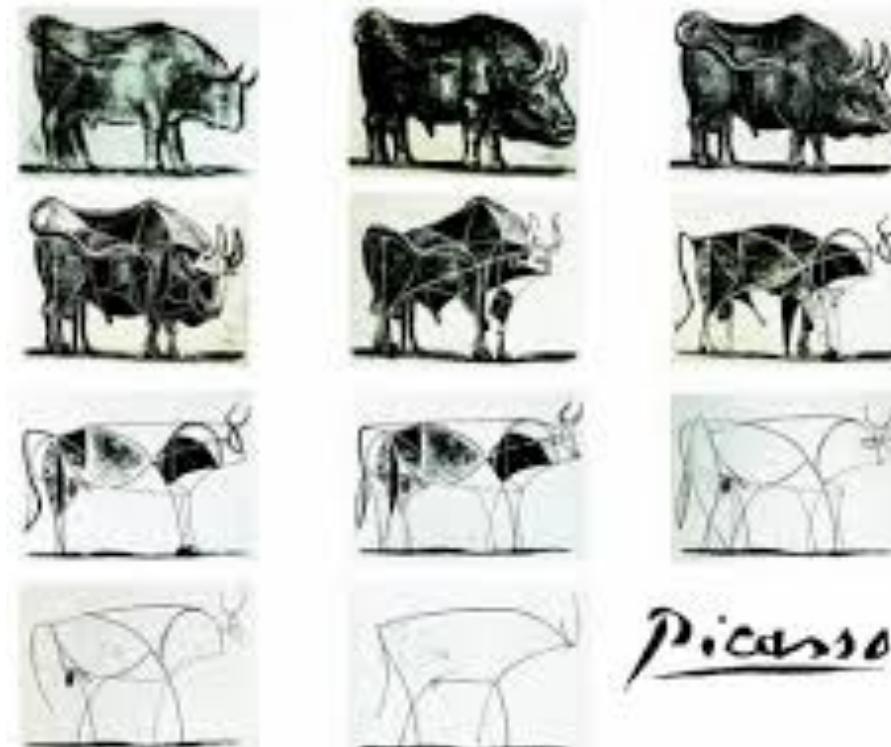
Listen to users and ops





Learn from data

Continuous
Improvement



Pablo Picasso, Bull (plates I - X) 1945

Design for
Failure





Thank You