ログ収集基盤ソフト
Fluentd超入門

オープンソースカンファレンス
2016/02/26 Tokyo Spring

SRA OSS, Inc. 日本支社
盛　宣陽

copyright © 2016 SRA OSS, Inc. Japan

アジェンダ

- Fluentd入門
- 可視化ツールの紹介
- 事例

copyright © 2016 SRA OSS, Inc. Japan

Fluentdとは

- OS、ミドルウェア、アプリケーション、機器から出力されるログを効率よく収集するソフトウェア
  - ログの転送と集約
    - 散らばっているログの集約
  - ログの加工
    - 調査しやすいようにログレコードの構造化や時刻の整形
  - フィルタリング機能
    - ログレコードの取捨選択
  - バッファリング機能
    - ログの取りこぼし防止
  - 他ツールと連携してログの可視化
    - 可視化ツールと連携するプラグインが豊富

copyright © 2016 SRA OSS, Inc. Japan

- URL

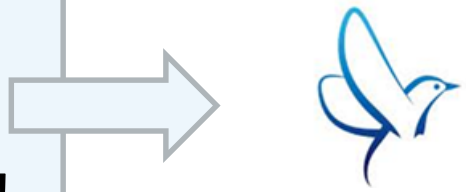
  http://fluentd.org/
- 開発
  - 米国Treasure Data社の開発者
  - コミュニティベース
  - 言語：Ruby + C言語
- ライセンス
  - Apache License Version 2.0

copyright © 2016 SRA OSS, Inc. Japan

イメージ

データ入力：Input

アプリケーションのログ
Webサーバのログ
データベースのログ
Syslog
http入力
Unixドメインソケット入力
コマンド実行結果



フィルタ・バッファ・ルーティング

データ出力：Output

DB

ファイル

コマンド

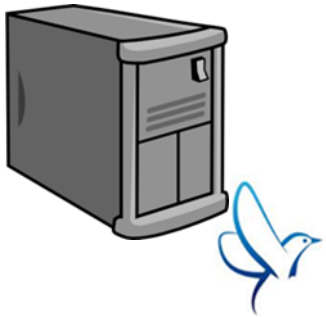
copyright © 2016 SRA OSS, Inc. Japan

物理構成イメージ

データ入力

データ転送

syslog



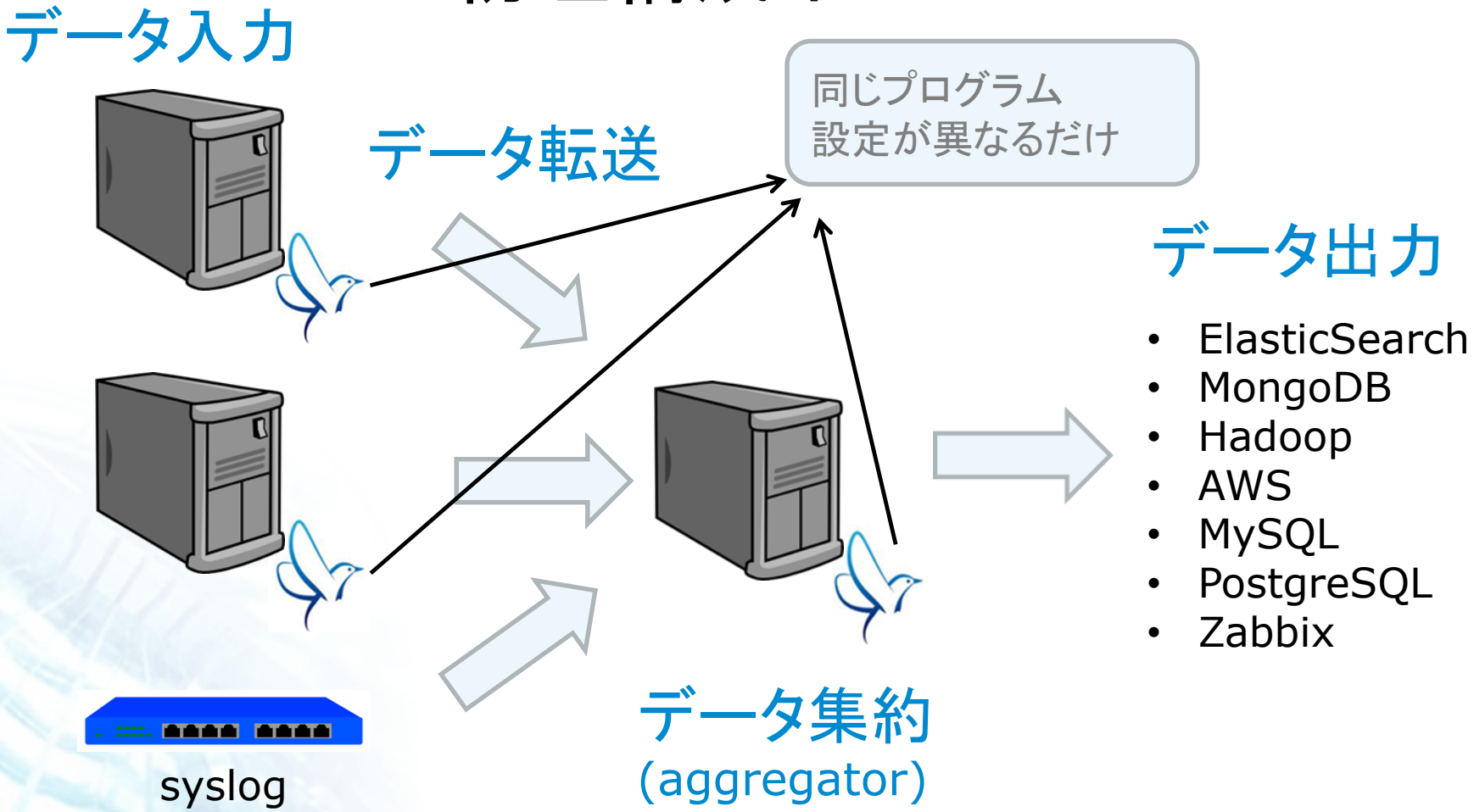
データ集約
(aggregator)

データ出力

- ElasticSearch
- MongoDB
- Hadoop
- AWS
- MySQL
- PostgreSQL
- Zabbix

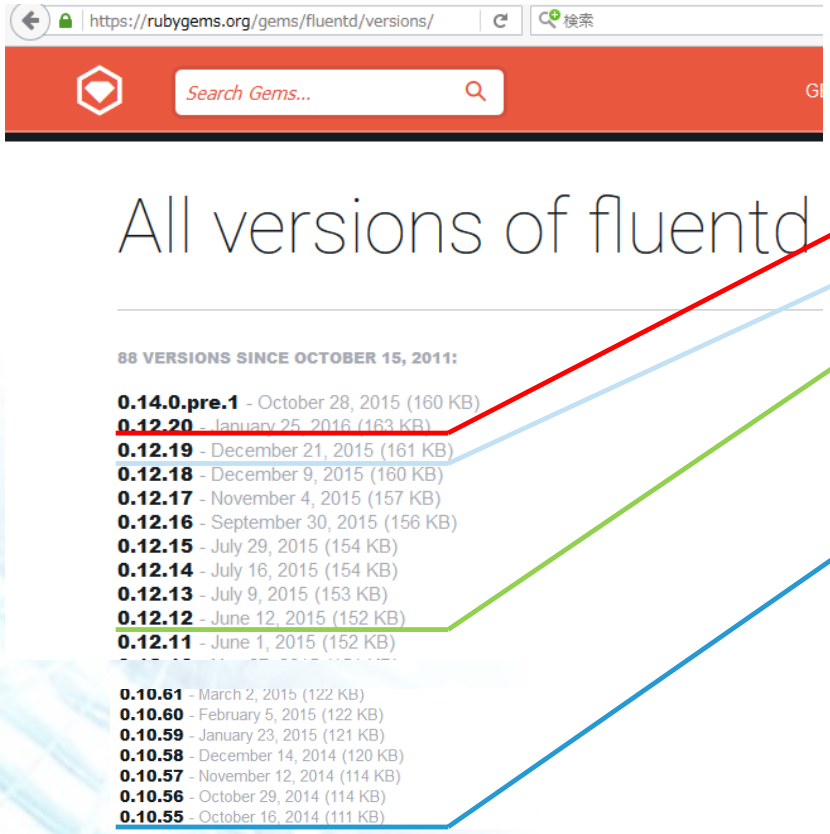
copyright © 2016 SRA OSS, Inc. Japan

物理構成イメージ



copyright © 2016 SRA OSS, Inc. Japan

Fluentdとtd-agentについて



- td-agentはTreasure Data社が提供するfluentdのパッケージ版
- 最新の安定版はtd-agent2
- REHL/CentOS/Amazon Linux Ubuntu/MacOS X

0.xx(偶数)がメージャバージョン

copyright © 2016 SRA OSS, Inc. Japan

Windows対応

- flunetd 0.14~
  - 安定版は未リリース
  - 0.14.0.pre.1 - 2015/10/29
    - Windows support
    - Sub-second event time support

現状は
nxlogでwindowsのログ(ファイルのレコード、イベントログ)を取得して
nxlogからflunetdに転送

copyright © 2016 SRA OSS, Inc. Japan

fluentdの特徴

- 簡単なインストール
- ログの構造化
- 豊富な機能(プラグイン機能)

copyright © 2016 SRA OSS, Inc. Japan

簡単なインストール

- コマンド一発　例)RHEL(CentOS)

```
curl -L https://toolbelt.treasuredata.com/sh/install-redhat-td-agent2.sh | sh
```

注　fluentdの安定版のパッケージであるtd-agent2がインストールされます
　　稼働に必要なRuby 2.1.xや基本的なプラグインもインストールされます

- 起動・停止

```
service td-agent start
service td-agent stop
```

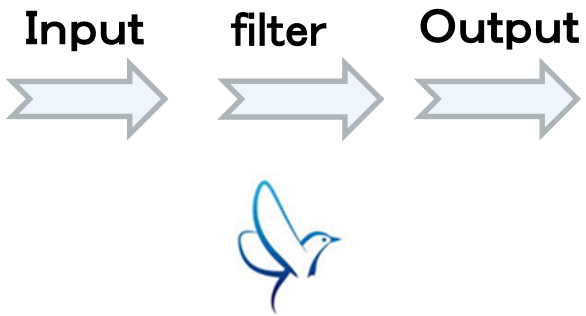
- プロセス

```
/opt/td-agent/embedded/bin/ruby /usr/sbin/td-agent
--log /var/log/td-agent/td-agent.log --use-v1-config --group td-agent --daemon /var/run/td-agent/td-agent.pid
```

copyright © 2016 SRA OSS, Inc. Japan

Flunetdの設定とプラグイン

- Flunetdの設定＝利用したい機能(プラグイン)の設定をする
- 4種類のプラグインを駆使してログ収集・加工・蓄積を行う
  - Input プラグイン
    <source> ~ </source>
  - Filterプラグイン
    <filter タグ名> ~ </filter>
  - Outputプラグイン
    <match タグ名> ~ </match>
  - バッファプラグイン
    Outputプラグインでバッファ機能を継承して利用されている

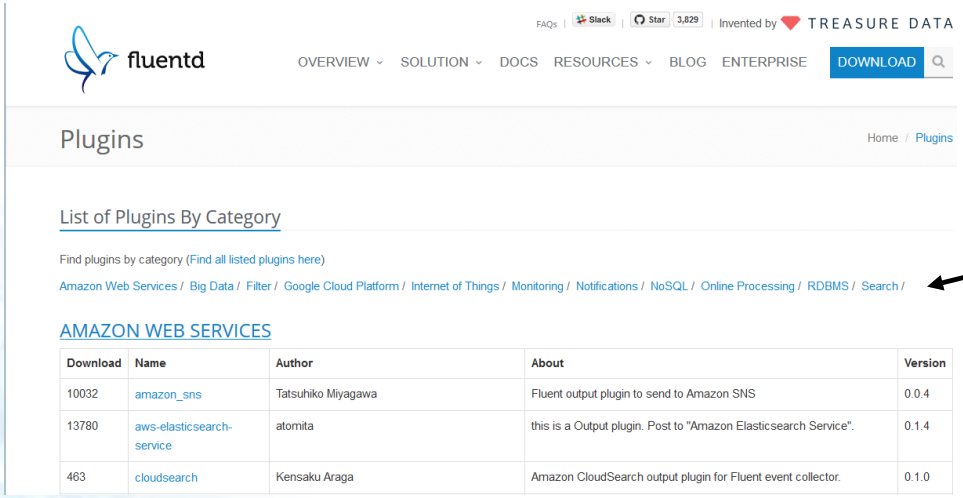


(開発者向け) 他にParser プラグイン、Formatterプラグインあり

copyright © 2016 SRA OSS, Inc. Japan

プラグインの検索

- 利用したいプラグイン＝機能を検索
  - http://www.fluentd.org/plugins



カテゴリ別に検索

- WEB上で検索(日本語で書かれた情報が豊富)

copyright © 2016 SRA OSS, Inc. Japan

プラグインの追加方法

- プラグインの入手方法
  ruby gemでインストール
  例） elasticsearchプラグインの取得

```
/opt/td-agent/embedded/bin/fluent-gem install fluent-plugin-elasticsearch
```

- プラグイン数 552種類
  2016年2月時点
  http://fluentd.org/plugins/allあるもの

copyright © 2016 SRA OSS, Inc. Japan

ルーティングとタグ

- ルーティングとは
  - 一つのログレコードに対してどのような処理を行いながらどのように最終目的地に届けられるか
  - 処理に対して利用するプラグインを定義
- タグとは
  - ログレコードの識別子
    - ログの種類、ミドルウェア名、ホスト名などで定義
    - 例) access.httpd.hostA (ホストAのhttpdのアクセスログ)
  - タグ名でログレコードが利用するプラグイン(加工・転送・保存処理)を使い分けてルーティングの制御を行う

copyright © 2016 SRA OSS, Inc. Japan

ルーティングとタグのイメージ

httpd access.logの例
目的:加工を行って他のサーバ上のDBに保存したい

Inputプラグイン tailでログ取得
タグの設定 access.httpd.host名

⇩

タグ名 access.httpd.host名に対して filterプラグイン grepでログ選別

⇩

タグ名 access.httpd.host名に対して Outputプラグイン xxx でログ加工
タグ名の変更 xxx.access.httpd.host名

⇩

タグ名 **(すべてのログ)に 対してOutputプラグイン forward でログの転送

ログの転送先(aggregator)では

Inputプラグイン forwardでログ取得

⇩

タグ名 xxx.access.httpd.に対してOutputプラグイン yyy でログ加工**
タグ名の変更 yyy.access.httpd

⇩

タグ名 yyy.access.httpd.に対してOutputプラグイン zzz でログ加工**
タグ名の変更 access.httpd

⇩

タグ名 access.httpdに対してOutputプラグイン xyz でDBに保存

copyright © 2016 SRA OSS, Inc. Japan

再: ルーティングとタグ

Flunetdはタグ名でルーティングの制御を行うことができる

しかし

- タグ設計が難しい・・・・
- 設定ファイルが読みにくくなる・・
- タグ名が次の二つの識別子を兼ねていることが問題
  - ログレコードの識別子
  - ルーティングの制御識別子

ということで

flunetd v0.12(td-agent2)ではルーティングを制御するためのラベルというものが利用できるようになりました

copyright © 2016 SRA OSS, Inc. Japan

ルーティングとラベルのイメージ

httpd access.logの例
目的:加工を行って他のサーバ上のDBに保存したい

Inputプラグイン tailでログ取得
タグの設定 access.httpd.host名
ラベルの設定 route1

⇩

ラベル名 route1に対して
filterプラグイン grepでログ選別

⇩

ラベル名 route1に対して
Outputプラグイン xxx でログ加工

⇩

タグ名 全てのログ(**)に対してOutputプラグイン forward でログの転送

ログの転送先(aggregator)では

Inputプラグイン forwardでログ取得

⇩

タグ名 access.httpd.に対してOutputプラグイン yyy でログ加工**
ラベル名 route2の定義

⇩

ラベル名 route2 に対してOutputプラグイン zzz でログ加工

⇩

タグ名 access.httpd.*に対してOutputプラグイン zzz でDBに保存

タグ名は変更なし

copyright © 2016 SRA OSS, Inc. Japan

ラベル利用時の注意事項

- 全てのプラグインでラベルが利用できるとは限らない
  - Flunetd 0.10 (td-agent)の時に開発されたプラグインでメンテナンスされていないプラグインはラベルの利用ができません
    - ソースコード上では、
      - Engine.emitを利用しているプラグインはラベルは使えません。
      - router.emitを利用しているプラグインはラベルが使えます。

copyright © 2016 SRA OSS, Inc. Japan

ログの構造化①

- ログの構造化とは
  - ログに構造を持たせて後々検索しやすいように加工すること

- Flunetdでは時刻フォーマットの統一とログの内容をJSON化することで構造化を持たせることができます

```
value1  value2  value3
=>
{“key1":“value1”,“key2":“value2",“key3",“value3“}
```

JSONで柔軟に定義

copyright © 2016 SRA OSS, Inc. Japan

ログの構造化②

- 入力:Input (Apacheのアクセスログ)

```
192.168.8.48 - - [20/Nov/2013:19:05:36 +0900] "GET /10.html
HTTP/1.0" 200 3 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64;
rv:25.0) Gecko/20100101 Firefox/25.0"
```

- 出力:Output

```
2013-11-20T19:05:36+09:00          ・・・・・時刻データ (注 apacheログと比較:時刻
                                              フォーマットの加工)
Host.Apache.access                 ・・・・・タグ: 利用者が定義(データ識別子)

 {"host":"192.168.8.48","user":"-
","method":"GET","path":"/10.html","code":"200","size":"3","referer":"-
","agent":"Mozilla/5.0 (Windows NT 6.1; WOW64; rv:25.0)
Gecko/20100101 Firefox/25.0"}      ・・・・・レコード JSON形式のログ内容
```

copyright © 2016 SRA OSS, Inc. Japan

ログの構造化③

- ログの構造化方法
  - 正規表現で指定してログに構造をもたせることができます
- tailプラグインで解説
  - tailコマンドのようにログファイルの行を取得する
  - 設定ファイル(/etc/td-agent/td-agentd.conf)

```
<source>      #Inputプラグインは<source>ディレクティブで指定
   @type tail 利用プラグインの指定
   tag        タグ名
   path       ログファイルパス
   pos_file   指定ファイルに最後に読み込んだ位置を記録します。
   format     ログのフォーマット '/'で囲まれた正規表現を指定
   time_format  時間フィールドのフォーマット(strftime形式)
</source>
```

copyright © 2016 SRA OSS, Inc. Japan

簡単な例 Input tail プラグイン

- ログ生成スクリプト　1秒ごとにログを出す

```
#!/bin/bash
export LANG=C
while :
do
        echo `date` ERROR テスト >> /tmp/test.log
        sleep 1
done
```

- ログ出力サンプル(□スペース)

```
Mon□Mar□18□00:56:03□JST□2014□ERROR□テスト
```

- fluentdの形式(時刻、タグ名、ログ内容)で取る

```
時刻 タグ名{"ERRORLEVEL":"ERROR" ,"MESSAGE":"テスト"}
```

copyright © 2016 SRA OSS, Inc. Japan

簡単な例 Input tail プラグイン 設定

- 設定ファイルの中身（□スペース）

```
<source>
  @type      tail
  tag        localhost.tail
  path       /tmp/test.log
  pos_file /tmp/test.pos
  format□/^(?<time>[^ ]*□[^ ]*□[^ ]*□[^ ]*□[^ ]*
   □[^ ]*)□(?<ERRORLEVEL>[^ ]*)□(?<MESSAGE>.*)$/
  time_format□%a□%b□%d□%H:%M:%S□%Z□%Y
</source>
```

Flunetdが構造化されたログを取得する設定
時刻 タグ名{“ERRORLEVEL”:”ERROR” ,”MESSAGE”:”テスト”}

copyright © 2016 SRA OSS, Inc. Japan

簡単な例 Output file プラグイン

- 取得したログの出力
  - fileプラグイン
    - fluentdのログ形式でファイルに出力
  - 設定ファイル(/etc/td-agent/td-agentd.conf)

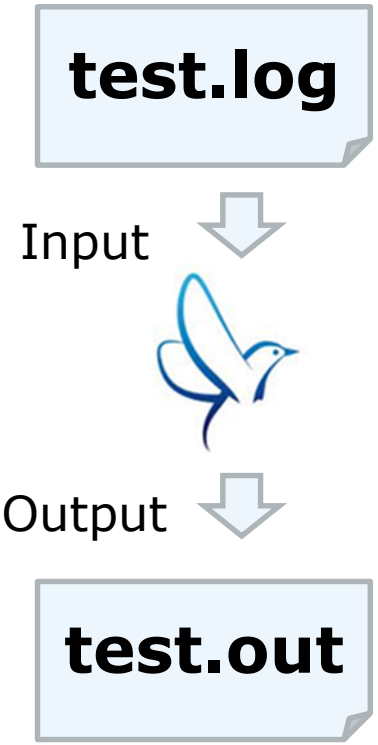
```
<match タグ名>      #outputプラグインで指定するディレクティブ
                    #inputプラグインに指定したタグ名を記述する
  @type file          #fileプラグインの設定
  path                出力ログファイルパス
  time_slice_format   出力ファイル名＋付与される名前
                          (デフォルト%Y%m%d)
  time_wait   出力ファイルがローテーションする前に待つ時間
                          (デフォルト10分)
  <バッファの設定>  後述
</match>
```

copyright © 2016 SRA OSS, Inc. Japan

簡単な例 Output file プラグイン 設定

- 設定ファイルの中身（□スペース）InputとOutput

```
<source>
  @type      tail
  tag        localhost.tail
  path       /tmp/test.log
  pos_file /tmp/test.pos
  format□/^(?<time>[^ ]*□[^ ]*□[^ ]*□[^ ]*□[^ ]*
   □[^ ]*)□(?<ERRORLEVEL>[^ ]*)□(?<MESSAGE>.*)$/
  time_format□%a□%b□%d□%H:%M:%S□%Z□%Y
</source>
<match localhost.tail>
  @type file
  path /tmp/test.out
</match>
```



copyright © 2016 SRA OSS, Inc. Japan

オリジナルのログと構造化されたログ

- Inputのログ形式

```
Mon□Mar□18□00:56:03□JST□2014□ERROR□テスト
```

- Outputファイルのログ形式

```
2014-03-18T00:56:03+09:00
localhost.tail
{"ERRORLEVEL":"ERROR","MESSAGE":"テスト"}
```

copyright © 2016 SRA OSS, Inc. Japan

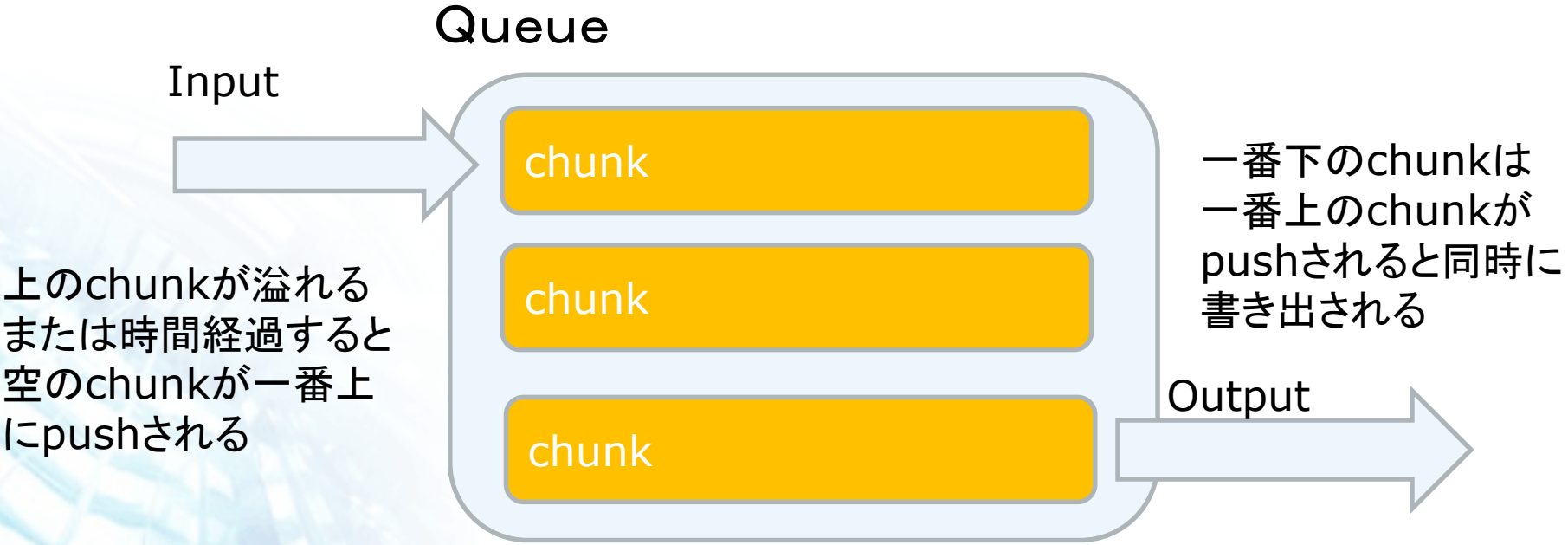
必ず利用する機能 バッファリング①

- バッファプラグイン
  - 性能と信頼性向上
    - まとめて出力
    - 受け取ったデータを保管して出力先の一時的な障害に対応
  - バッファリングするOuputプラグインとセットで利用
    (ファイル・DB出力や転送処理)
  - オンメモリとファイルの指定が可能

copyright © 2016 SRA OSS, Inc. Japan

必ず利用する機能 バッファリング②

- バッファの構造 (Queueとchunk)
  - chunkのサイズとQueue内のchunk数は設定可能



copyright © 2016 SRA OSS, Inc. Japan

必ず利用する機能 バッファリング③

設定例

```
<source>
  @type      tail
  tag        localhost.tail
  path       /tmp/test.log
  pos_file /tmp/test.pos
  format□/^(?<time>[^ ]*□[^ ]*□[^ ]*□[^ ]*□[^ ]*
   □[^ ]*)□(?<ERRORLEVEL>[^ ]*)□(?<MESSAGE>.*)$/
  time_format□%a□%b□%d□%H:%M:%S□%Z□%Y
</source>
<match localhost.tail>
  @type file
  path /tmp/test.out
  buffer_type  memory
  buffer_queue_limit 64
  buffer_chunk_limit 8m
  flush_interval  60s
  retry_wait 1s
  retry_limit 10
</match>
```

outputプラグイン単位でバッファの設定

copyright © 2016 SRA OSS, Inc. Japan

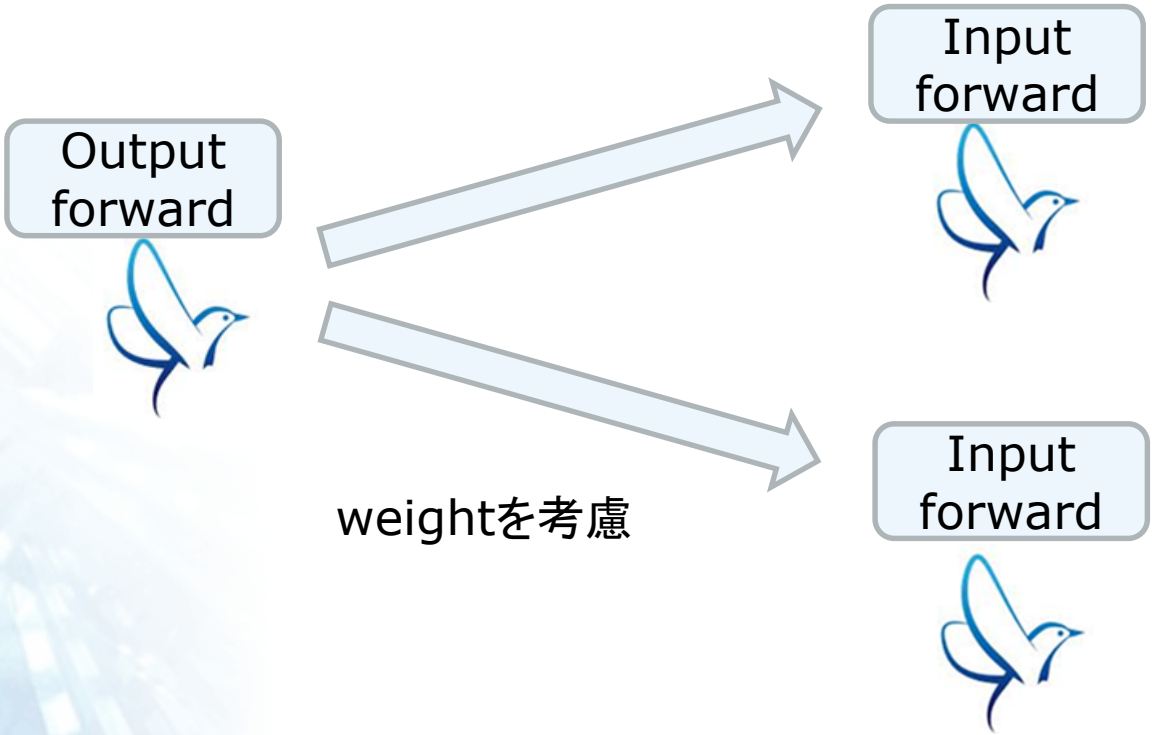
必ず利用する機能　ログ転送①

- forwardプラグインによるログ転送

copyright © 2016 SRA OSS, Inc. Japan

必ず利用する機能　ログ転送②

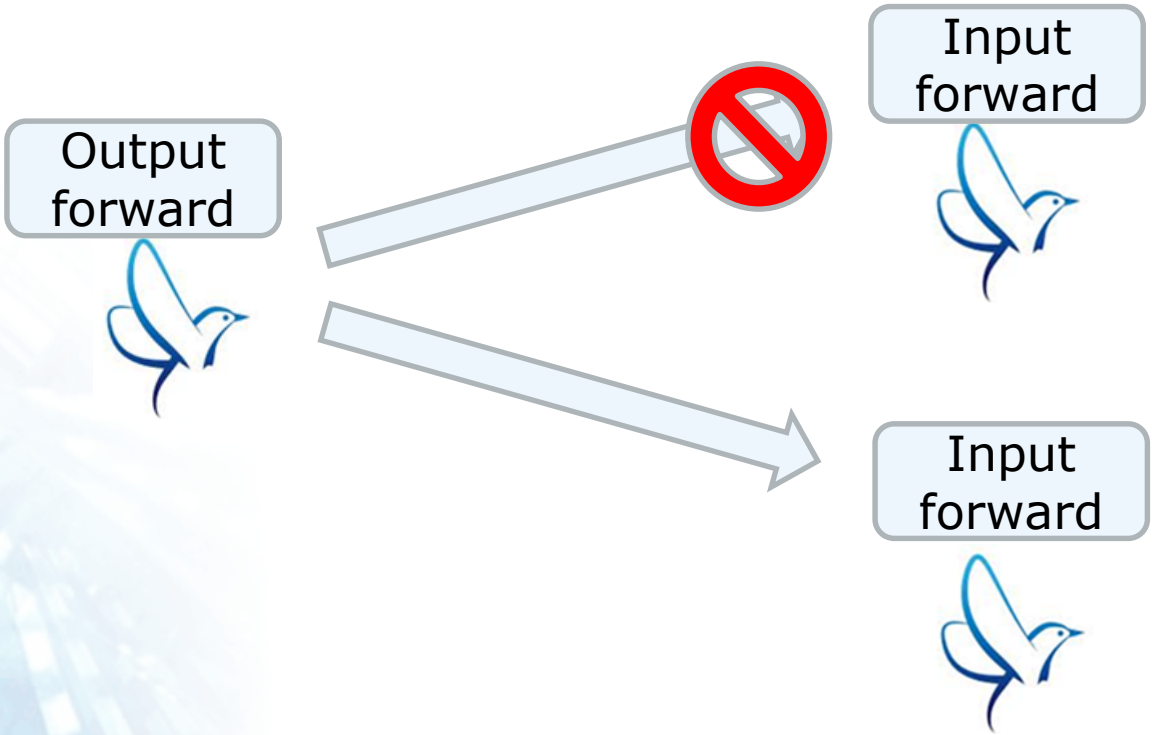
- forwardプラグインによる負荷分散



copyright © 2016 SRA OSS, Inc. Japan

必ず利用する機能　ログ転送③

- forwardプラグインによる冗長化構成



copyright © 2016 SRA OSS, Inc. Japan

必ず利用する機能　ログ転送④

設定例

```
<match pattern>
   @type forward
...
<server>
   name myserver1
   host 192.168.1.3
   weight 60
</server>
<server>
   name myserver2
   host 192.168.1.4
   weight 60 standby #待機系
</server>
</match>
```

分かりやすい設定

copyright © 2016 SRA OSS, Inc. Japan

ログの可視化ツール

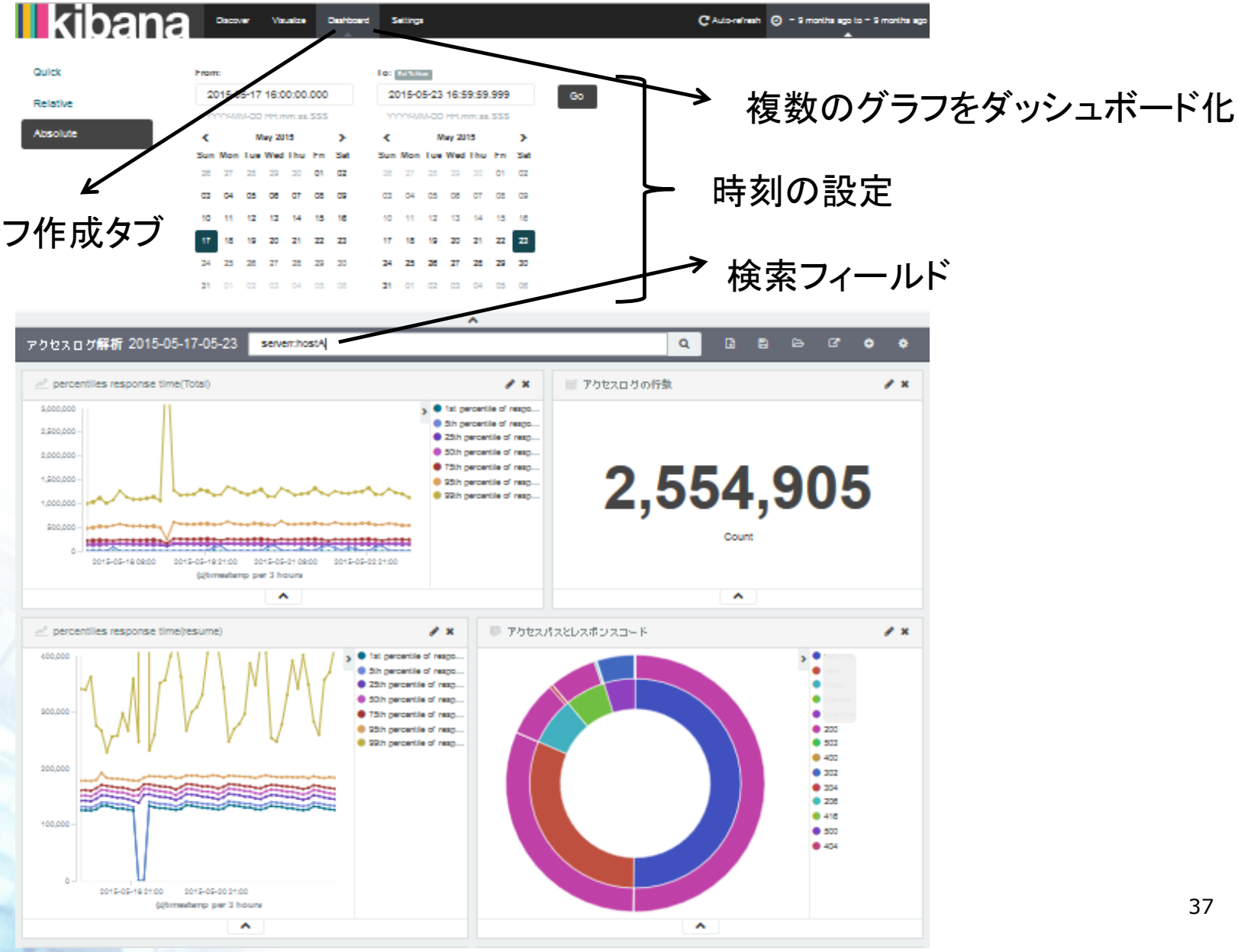
- Elasticsearchとkibana
- GrowthForecast
- Zabbix

copyright © 2016 SRA OSS, Inc. Japan

Elasticsearchとkibana

- Elastic社によって開発
  - DB部分がElasticsearch
    - 商用版とOSS版がある
    - 高いスケーラビリティ
    - REST-APIで操作
  - Viewer部分がkibana
    - Elasticsearchのデータを元に柔軟にグラフ作成が可能
    - 既存のデータから見たい要素を取り出して確認しながらグラフ定義
    - 複数のグラフをまとめてダッシュボード化
    - 直感的な操作
    - データが肥大化する傾向 →短期的な詳細ログ解析に向いている
- fluent-plugin-elasticsearchを利用してflunetdと連携

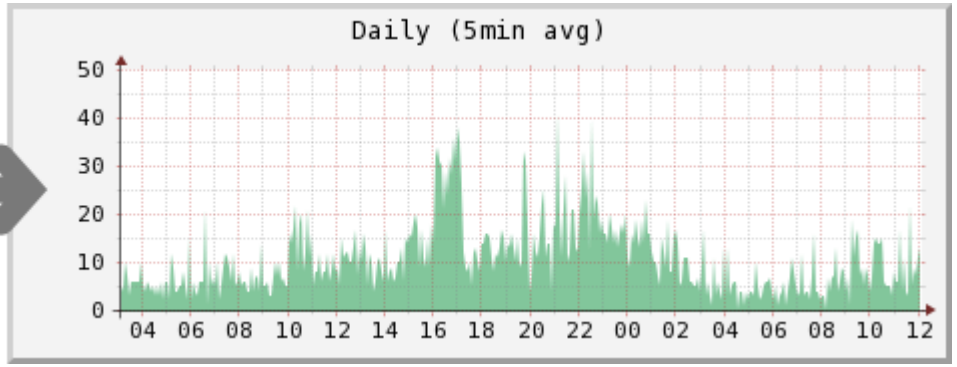
copyright © 2016 SRA OSS, Inc. Japan


kibana
Discover
Visualize
Dashboard
Settings
Auto-refresh
Quick
Relative
Absolute
From:
To:
2015-05-17 16:00:00.000
2015-05-23 16:59:59.999
Go
May 2015
Sun Mon Tue Wed Thu Fri Sat
グラフ作成タブ
複数のグラフをダッシュボード化
時刻の設定
検索フィールド
アクセスログ解析 2015-05-17-05-23
server:hostA
percentiles response time(Total)
アクセスログの行数
2,554,905
Count
percentiles response time(resume)
アクセスパスとレスポンスコード

GrowthForecast

- httpプロトコルを利用してサーバにデータをPOSTするとグラフが生成されるツール
- グラフはRDDtoolで作成される
  - サイズが増えずに長期的な時間軸の描画が可能
    - 解析向けではなく定点観測向け
      (時間軸の変更ができない)

httpでデータをポスト



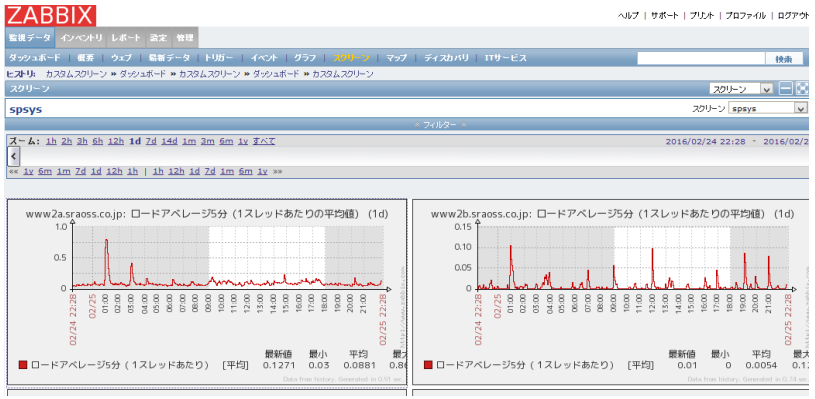
fluent-plugin-growthforecast
でGrowthForecastと連携

copyright © 2016 SRA OSS, Inc. Japan

Zabbix

- 監視ツールZabbixと連携
  - Flunetdで取得、加工したメトリックスをZabbixに送信してグラフ化
  - 通知もZabbixから行える
  - Zabbixではできない複数行ログを生成するログ監視の代替として利用
  - Zabbixのログ収集よりも高速に機能
  - 古いデータは自動的にトレンド化(最大、最小、平均)して保存するため古いデータの可視化もしやすい

fluent-plugin-zabbixを利用して
flunetdとZabbixの連携



copyright © 2016 SRA OSS, Inc. Japan

事例

- 構築事例
- システムコンサルティングでの活用事例
- サポート事例

「公開資料では事例を削除させていただきます」

copyright © 2016 SRA OSS, Inc. Japan

サービスの紹介

- **Fluentd エンタープライズサポートサービス**
  - 開発元の米国トレジャーデータ社と**Fluentd** のサポートで協業しました。(2015/10～)
- OSSまとめてサポートサービス
  - 幅広いOSSソフトウェアに対してまとめてワンストップでサポートします
- Zabbix/Hinemosサポート
  - 開発元のパートナーとしてサポートサービスと提供しております
- OSSシステムコンサルティング・構築
  - OSS全般のコンサルティング
  - OSS全般の構築

PostgreSQLもやっています

詳細はブース[201]で

copyright © 2016 SRA OSS, Inc. Japan

ご清聴ありがとうございました

copyright © 2016 SRA OSS, Inc. Japan