

Zabbixの可能性を広げる!

～データ収集ゲートウェイの事例紹介～

NTTコムソリューションズ株式会社

マネジメントソリューション本部
プラットフォームソリューション部

田中 武信

田中 武信

所属

NTTコムソリューションズ株式会社
マネジメントソリューション本部
プラットフォームソリューション部

経歴

2000～08年 ネットベンチャー、大学等の運用現場を転々
2008年～ NTTコムソリューションズに所属
2011年～ Zabbixの設計・構築・製品企画などを担当
2018年 Zabbix Summit、Zabbix Conference Japanで登壇

What is LLDP?

- LLDP Standardization as IEEE 802.1AB in 2004
- Periodically send LLDP packets with multicast
- Information on neighbor devices can be read by SNMP (LLDP-MIB)

Neighbors Information
- Device A
- Device B
- Device C

Network Switch

Device A

Device B

Device C

LLDP

LLDP

LLDP

ZABBIX SUMMIT 18

Monitoring and Visualization of LLDP information in Zabbix

Takeshi Tanaka
Systems Engineer at
NTT Com Solutions,
Japan

NTT コムソリューションズ株式会社

(<https://www.nttcsol.com/>)

- NTTコミュニケーションズの100%子会社
- 2008年より、Zabbix社と提携したZabbix関連事業を開始
- ZABICOMソリューションの導入/運用/製品供給などのサービスを提供



NTT Com Solutions
an NTT Communications Company

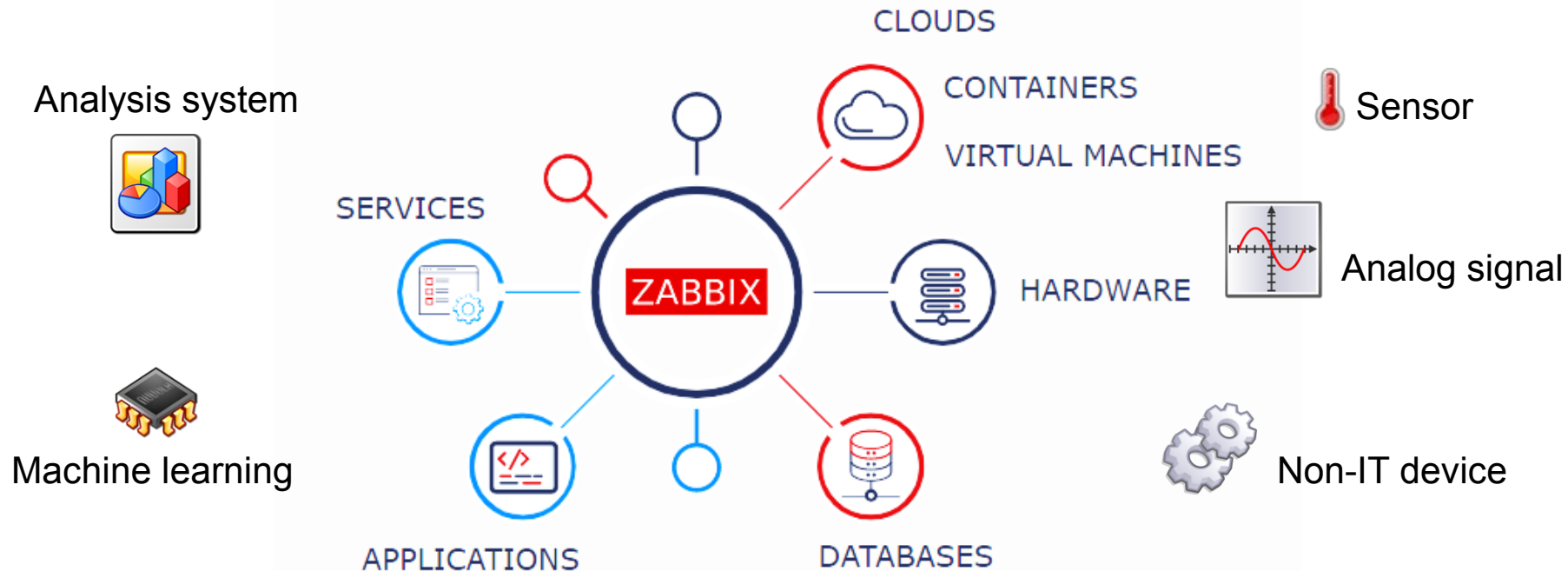


<https://www.zabicom.com>

Zabbixの適用範囲

Scope of Zabbix

Zabbixは、IT（情報通信技術）を構成する要素を監視対象としてきました。
近年、IT以外からの情報収集や他システムとの連携が求められるケースが出てきました。



出典 : <https://www.zabbix.com/jp/features>

新領域に適用できるか？

Is Zabbix applicable to the new scope?



Machine learning



Analysis system

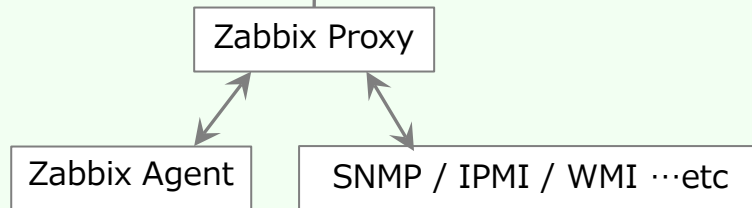
データ渡す仕組み
(Data passing mechanism)



データ溜める仕組み
(Data storage mechanism)



データ集める仕組み
(Data collection mechanism)



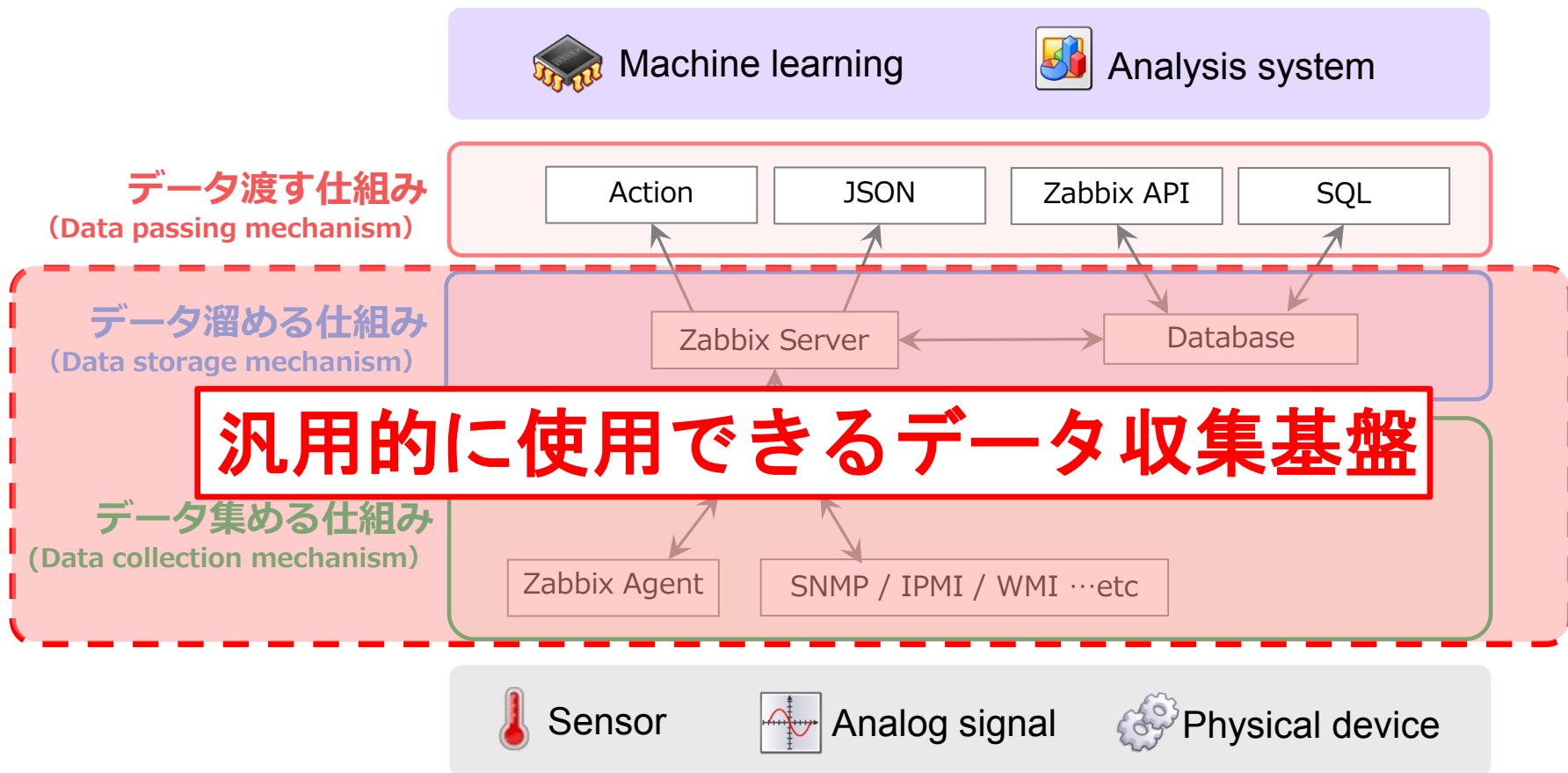
Sensor



Analog signal



Non-IT device

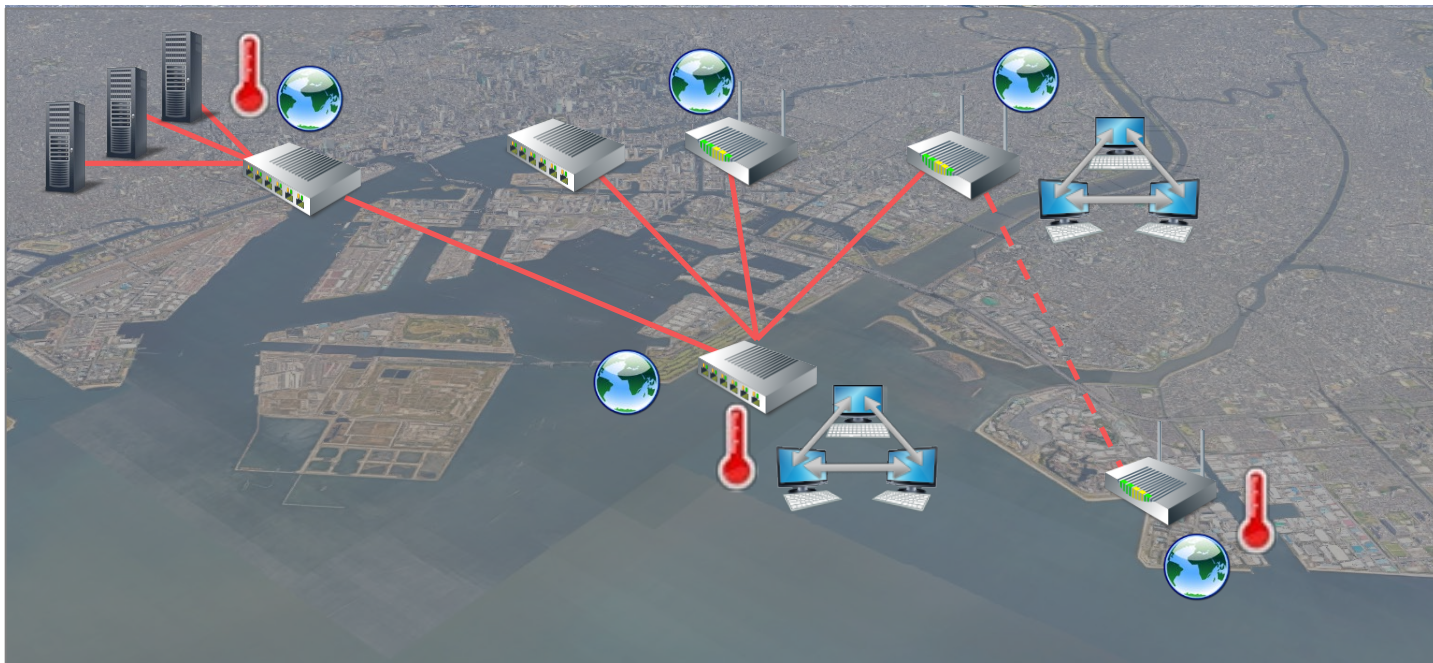


データ収集基盤としてのZabbix

Zabbix as a data collection platform

目的：とあるイベントにおける機器/環境の監視

対象：ネットワーク機器、サーバ・パソコン、環境情報、位置情報



※画像はイメージです

① IT機器からのデータ収集
(Data collection from IT devices)

⇒ Zabbixの機能範囲内

- ・ IPネットワーク（第三層以上）の情報収集
- ・ 他システムからのデータ受信(SNMPTrap等)

② 環境情報の収集
(Collection of environmental information)

⇒ 対応機器を介した情報の収集

- ・ 位置情報の収集
- ・ センサー機器からのデータ収集

③ 複数の物理拠点に分散
(Monitor multiple locations)

⇒ データセンター外にZabbixを設置

- ・ 屋内外を問わず動作できる仕組み
- ・ 劣悪な環境下での運用

⇒ この要件を満たすため、各拠点にZabbix Proxyを設置する方針としました

Zabbix Proxyの利点と課題

Advantages and problems of Zabbix Proxy

◆構成上の差異

① 通信セッション数の最小化

Minimizing communication sessions

② パッシブ監視 / コマンド実行

Passive monitoring / command execution

③ 通信圧縮

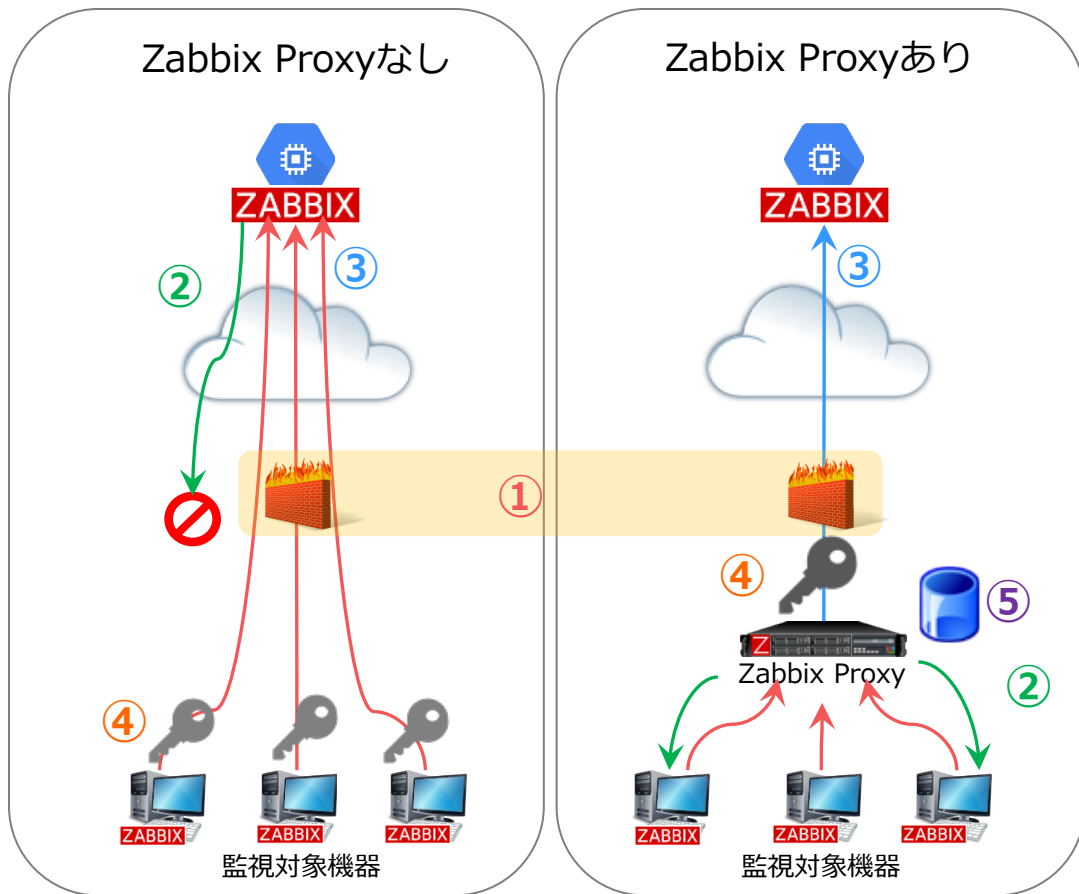
Compression

④ 通信暗号化

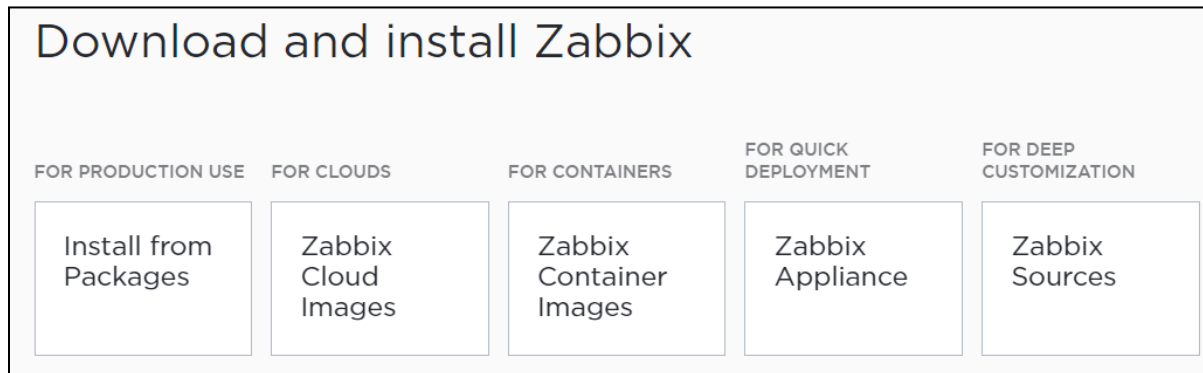
Encryption

⑤ オフラインバッファ

Offline buffer



Zabbix Proxyはパッケージ / クラウドイメージ / コンテナ / 仮想マシンなどの形で提供



これらを使用するには仮想化環境やサーバ機などが必要になる

⇒ 拠点ごとにサーバ機を設置することが困難

◆Zabbix Proxy Appliance (ZP-1400)

- ・ Zabbix Japanが提供するアプライアンス製品

CPU : ARMプロセッサ (32bit)

NIC : 10/100/1000BASE-T 2ポート
PoE受電対応

◆課題点

- ・ Linuxで動作しているが、OSの**カスタマイズ不可**
- ・ 無線ネットワークが**未搭載**
- ・ ストレージは未搭載
⇒ 再起動すると**オフラインバッファが消える**



<https://www.zabbix.com/jp/zp1400>

要件を充足できないため、新たに「データ収集ゲートウェイ」を作りました

データ収集ゲートウェイの実装と運用

Implementation and operation of data collection gateway

◆概要

- ・産業用のコンパクトサーバ
- ・ Intel Atom系列のプロセッサを搭載
- ・ インターフェイスを多数搭載
 - > GbE NIC 2 Port
 - > USB 2.0 4 Port
 - > HDMI 1 Port
 - > RS232C 3 Port
- ・ 拡張用オプションにより機能付加が可能
- ・ ファンレス設計
- ・ ACアダプタにより駆動



◆概要

- OS は Ubuntu を搭載
- Zabbix Proxy 4.0 を搭載
- 専用管理UI は Zabbix Appliance 準拠
- **ファームウェアは Zabbix Japan社製**

The screenshot shows the Zabbix Enterprise Appliance web interface. The top navigation bar includes links for 'システム管理', 'サポート', and 'カスタマーポータル'. Below the navigation bar, there are tabs for 'システム情報', 'システム設定', 'ネットワーク', 'Zabbixプロキシ', 'VPNクライアント', 'パスワード', and 'サブスクリプション'. The main content area is divided into three sections: 'システム情報', 'システムステータス', and 'プロセスステータス'.

システム情報		システムステータス	
ホスト名	ZP-7400-02	ロードアベレージ	0.33
システム時刻	2019-11-13 11:12:16	空きメモリ	1.5 GB
起動ファームウェア		システム領域使用率(/)	15.6 %
システムファームウェア		ストレージ使用率(/boot, /opt)	18.1 % / 5.2 %
サブスクリプション		ディスク状態(消費率)	sda(不明)

プロセスステータス	
Cron	起動中
DNS Cache	起動中

◆追加機能

- ・ 位置情報 : スタンドアロン型GPSレシーバ
- ・ 通信機能 : SIM対応 (LTE / W-CDMA)
: 無線LAN対応 (IEEE802.11ac対応)
- ・ データ保持機能 : SSDを内蔵



位置情報の取得と活用

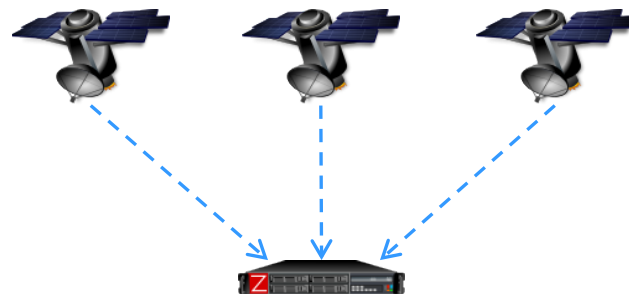
Collection and utilization of location information

◆位置情報の収集方法

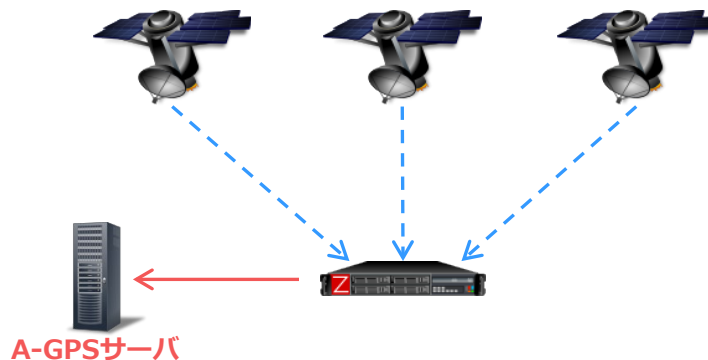
- GPS (Global Positioning System) を使用
- GPSの種類
 - > スタンドアロン型GPS
 - 衛星からの情報のみで位置情報を取得
 - 環境条件により、位置の特定に時間がかかる
 - > アシスト型GPS (A-GPS)
 - A-GPSサーバより衛星の軌道情報を取得
 - 衛星の情報と合わせて、迅速に位置情報を取得
 - A-GPSサーバと通信ができる必要がある

⇒ **スタンドアロン型GPSを採用**

スタンドアロン型GPS



アシスト型GPS



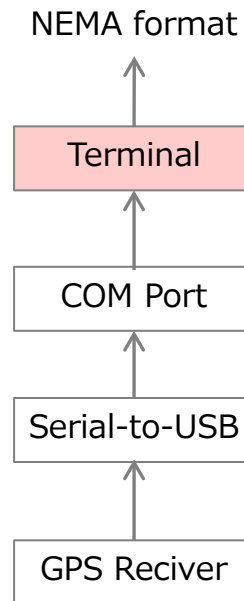
◆デバイス接続方法

- GPSレシーバはシリアル接続（COMポート）方式が主流

◆データの取得方法

- NEMAフォーマットのデータが取得できる
- 取得単位の概念はなく、データが流れ続ける仕様

```
$GPRMC,V,,,,,,,,,N*53
$GPGGA,,,,,0,00,,M,0.0,M,,0000*48
$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGSV,3,1,12,01,00,000,,02,00$GPGSV,3,1,09,02,25.3,303.8,19.8,03,9.8,53.4,25.7,05,10.5,2
$GPGSV,3,2,09,12,14.8,296.7,27.1,19,73.8,194.1,19.7,23,34.5,56.3,32.3,06,62.6,345.9,*57
$GPGSV,3,3,09,17,59.8,173.0,*47
$GLGSV,3,1,09,,,,,28.1,66,28.8,30.9,40.1,,,,,27.2,82,71.0,16.9,*73
$GLGSV,3,2,09,88,,,,,83,26.7,326.3,,81,40.8,120.9,,67,84.4,0.0,*57
$GLGSV,3,3,09,68,35.2,215.2,*52
$GPGGA,051531.0,3541.889534,N,13945.706254,E,1,04,2.1,80.1,M,39.0,M,,*6B
$PQXF1,051531.0,3541.889534,N,13945.706254,E,80.1,62.08,57.92,4.93*5E
$GNGNS,051531.0,3541.889534,N,13945.706254,E,AN,04,2.1,80.1,39.0,,*50,000,,03,00,000,,
```



◆ **gpsd** (<https://gpsd.gitlab.io/gpsd/>)

- Linux / FreeBSDで動作するGPSデータ取得用デーモン
- 専用APIからJSON形式の情報を返す機能を提供

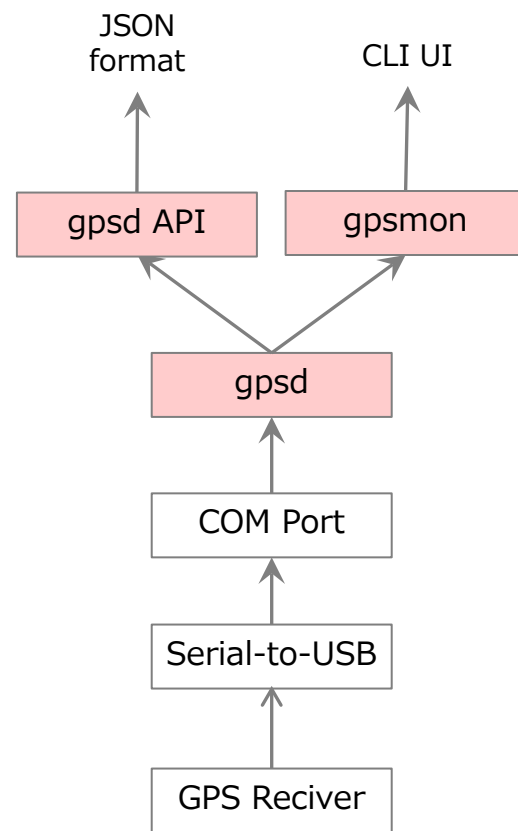
```

tcp://localhost:2947          SIRF>
----- X ----- Y ----- Z ----- North ----- East ----- Alt -----
Pos: -3951696  3359494  3699648  m      35.68209° 139.63081°  47  m
Vel:   0.0    0.0    0.0  m/s      0.0    -0.0    0.0  climb m/s
Time: 2019-11-10T16:09:47.020Z Leap: ??   Heading: -0.0°  0.0  speed m/s
Fix: 0 =                                HDOP: 0.0  M1: 00  M2: 00

----- Packet type 2 (0x02) -----
Measured Tracker
Ch PRN  Az  El  Stat  C/N ? SF
0  7 304 35 00bd 36.6
1  8 301 63 00ad 26.1
2  9 252  4 00ad 23.9
3 27  28 57 0000  0.0
4 11 226 29 0000  0.0
5 26 111 22 0000  0.0
6  1 201 18 0000  0.0
7 20  58 12 0000  0.0
8 30 321  8 0000  0.0
9 22 312 50 0000  0.0
10 16 139 37 0000  0.0
11 129 169 48 003d 29.8
----- Packet Type 4 (0x04) -----

----- Packet type 6 (0x06) -----
Firmware Version
----- Packet Type 6 (0x06) -----
CPU Throughput
Max: 0.000 Lat: 0.000 Time: 0.000 MS: 0
----- Packet type 9 (0x09) -----
Clock Status
SVs:  Drift:      Bias:
GPS Time:      PPS:
----- Packet type 7 (0x07) -----
Visible List
----- Packet type 13 (0x0D) -----
DGPS Status
----- Packet type 27 (0x1B) -----

```

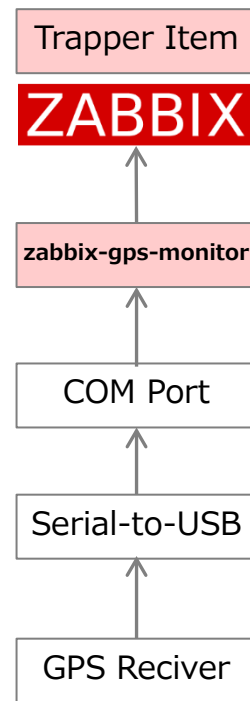


◆実装方法

- COM Portからデータの取得を行う常駐型プログラムを開発
- Zabbix Serverに対してデータを送信する方式
(zabbix_proxy.conf の送信先サーバ情報を使用)
- Zabbix Serverに受信用のトラッパーアイテムを作成

⇒ Zabbixにデータを格納することを前提としたシンプルな構造

⇒ 構造を単純にすることで、不具合発生のリスクを低減



The image shows two overlapping screenshots of the Zabbix web interface. The top screenshot shows the '最新データ' (Latest Data) page for host 'gps-pizero-01', with a red box highlighting the '位置情報 (4アイテム)' (Location Information (4 items)) section. The bottom screenshot shows the 'ホスト' (Host) configuration page for 'gps-pizero-01', with a red box highlighting the 'ホストインベントリ' (Host Inventory) tab and the '緯度 (Latitude)' and '経度 (Longitude)' input fields.

ZABBIX 監視データ インベントリ レポート 設定 管理

ダッシュボード 障害 概要 Web **最新データ** グラフ スクリーン マップ ディスカバリ サービス dev-geo-zbx01

最新データ

▼ ホスト 名前 ▲

▼ gps-pizero-01 位置情報 (4アイテム)

Read Geometry

経度 (Longitude)

緯度 (Latitude)

高度 (Altitude)

ZABBIX 監視データ インベントリ レポート 設定 管理

ホストグループ テンプレート **ホスト** メンテナンス アクション イベント相関関係 ディスカバリ サービス

ホスト

すべてのホスト / gps-pizero-01 有効 ZBX SNMP JMX IPMI アプリケーション 6 アイテム 23 トリガー 1 グラフ ディスカ

ホスト テンプレート IPMI マクロ **ホストインベントリ** 信号化

無効 マニュアル 自動

タイプ

緯度 35.682366 ← 緯度 (Latitude)

経度 139.691706 ← 経度 (Longitude)

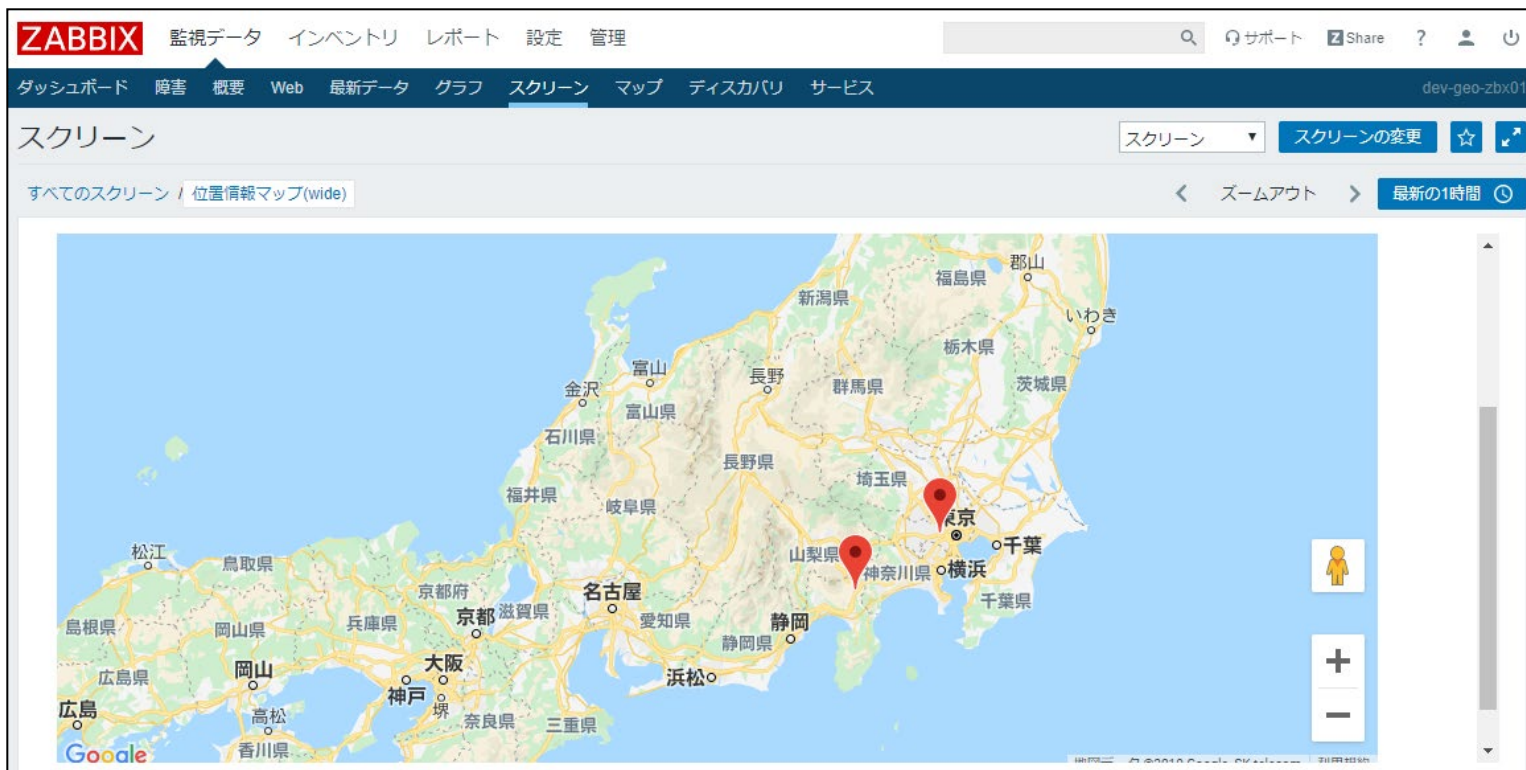
⇒ ホストインベントリに格納することで、最新の位置情報を参照できる

Zabbix APIを使い、位置情報が記録されているホストの一覧を取得

```
{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "output": ["host"],
    "selectInventory": [
    "searchInventory": [
  ],
  "id": "2",
  "auth": "${TOKEN}"
}

"result": [
  {
    "hostid": "10274"
    "host": "DESKTOP-A1CJC3R"
    "inventory": {
      "os": "Windows DESKTOP-A1CJC3R 10.0.17763 Microsoft Windows 10 Home x64"
      "location_lat": "35.29xxxxxxxxxx"
      "location_lon": "138.93xxxxxxxxxx"
    }
  }
  {
    "hostid": "10277"
    "host": "gps-pizero-01"
    "inventory": {
      "os": "Raspbian GNU/Linux 10 (buster)"
      "location_lat": "35.68xxxxxxxxxx"
      "location_lon": "139.63xxxxxxxxxx"
    }
  }
]
```

Zabbixから取り出した位置情報とGoogle Mapを組み合わせると・・・



無線通信の設定と運用

Wireless communication settings and operation

◆デバイス情報

チップ : Qualcomm MDM9215

モデル : SIM7100JC

接続 : mini PCI-e

◆通信規格

LTE通信 : FDD-LTE / TD-LTE

3G通信 : W-CDMA方式



Linuxにおける一般的な設定と同じです。

◆APN設定 (/etc/qmi-network.conf)

```
APN=APN NAME  
APN_USER=USER@DOMAIN  
APN_PASS=PASSWORD
```

◆インターフェイス設定 (/etc/systemd/network/wwan0.network)

```
[Match]  
Name=wwan0  
  
[Network]  
DHCP=ipv4
```

◆デバイス接続方法

- ・無線LAN子機
- ・USB接続

◆通信規格

- ・IEEE802.11ac / n / a / g / b
- ・ARIB STD-T71(5GHz)
- ・ARIB STD-T66(2.4GHz)



Linuxにおける一般的な設定と同じです。

◆ SSID設定（ /etc/wpa_supplicant/wpa_supplicant.conf ）

```
ctrl_interface=/var/run/wpa_supplicant
network={
  ssid="SSID"
  psk="Pre-Shared Key"
  key_mgmt=WPA-PSK
  id_str="wifi01"
}
```

```
ctrl_interface=/var/run/wpa_supplicant
network={
  ssid="SSID#1"
  psk="Pre-Shared Key"
  key_mgmt=WPA-PSK
  priority="10" ← 値の大きい方が優先されます
  id_str="wifi01"
}

network={
  ssid="SSID#2"
  psk="Pre-Shared Key"
  key_mgmt=WPA-PSK
  priority="5"
  id_str="wifi02"
}
```

◆ インターフェイス設定 (/etc/systemd/network/wlan0.network)

```
[Match]
Name=wlan0

[Network]
DHCP=ipv4
```

⇒ 無線LANに自動接続したが、どのIPアドレスが付与されたかが分からず、データ収集ゲートウェイの管理画面にアクセスできない！

IPアドレスが知りたければ、監視すればよい！

アイテム 保存前処理

*名前

タイプ

system.run["ip addr show **{\$IF1}** | grep ¥"inet ¥" | awk '{printf ¥"%s¥n¥",\$2}' | cut -d/ -f 1"]

データ型

ホストマクロ

マクロ	値	
{\$IF1}	wlan0	削除
{\$IF2}	wwan0	削除

Information (Hardware)

gps-pizero-01		Information (Network) (5アイテム)	
<input type="checkbox"/>	Default Gateway	2019/11/14 23:16:14	192.168.1.254
<input type="checkbox"/>	IP Address #A	2019/11/14 23:16:13	192.168.1.131

データ保持機能

Data retention function

◆ストレージデバイス

- ・ mini PCI-e接続のSSD

◆目的

- ・ オフラインバッファ保持

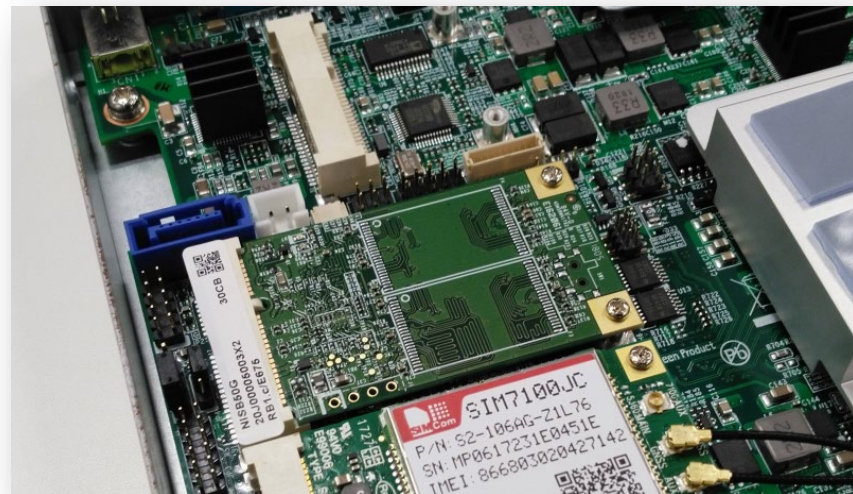
◆想定される事態

・ 通信品質の問題

- > 通信の信頼性が欠ける環境での使用
- > オフラインバッファが大量に溜まる

・ 不意の電源断

- > 発電機から電力供給がなされる環境での使用
- > 配線の断、発電機の停止などを想定



⇒ 今回の事例では、出番はありませんでした・・・

センサーを用いたデータ収集

Data monitoring using sensors

- Zabbixでの使用実績がある「feelers」の製品を使用 (<https://www.feelers.jp/>)
- Zabbixプロトコルにネイティブ対応 (Active Agentとして動作)
- コントローラはEthernet接続でZabbixへデータを送信
- コントローラとセンサーは1-Wireで接続



コントローラ

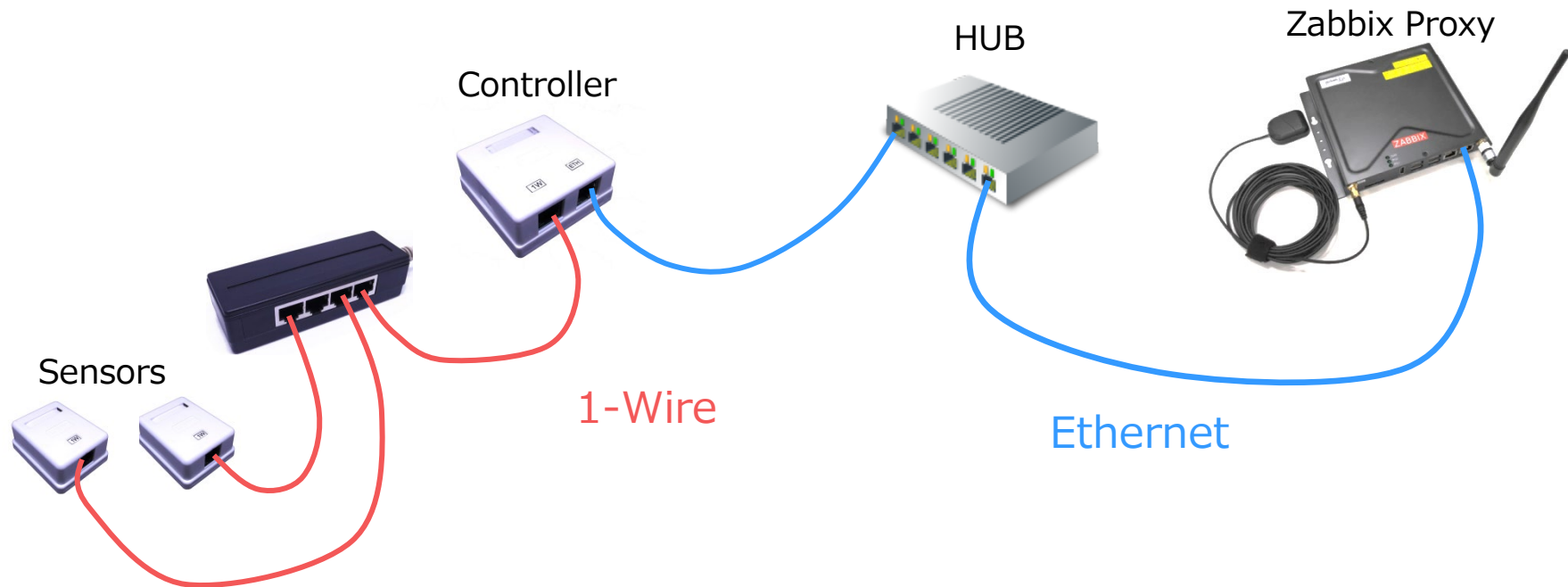


センサー (温湿度気圧)



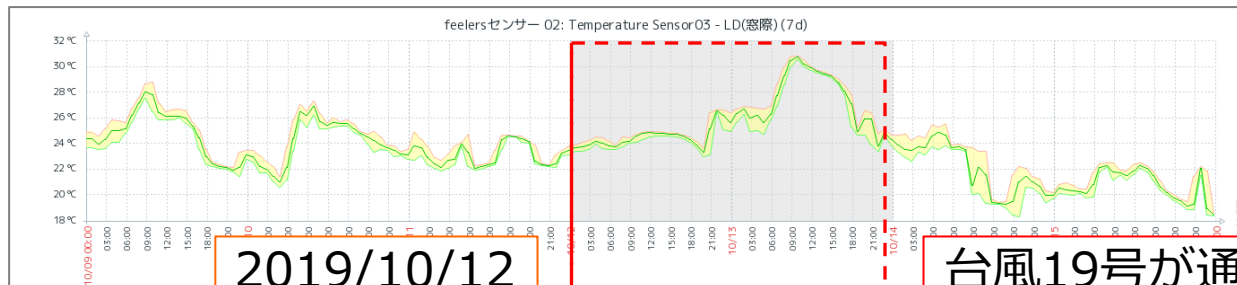
センサー (温度)

- コントローラはIPネットワークに接続
- コントローラとセンサーはLANケーブルで接続（RJ-45コネクタ）
- 複数個のセンサーを接続する場合にはスプリッタで分岐（最大24センサーに対応）

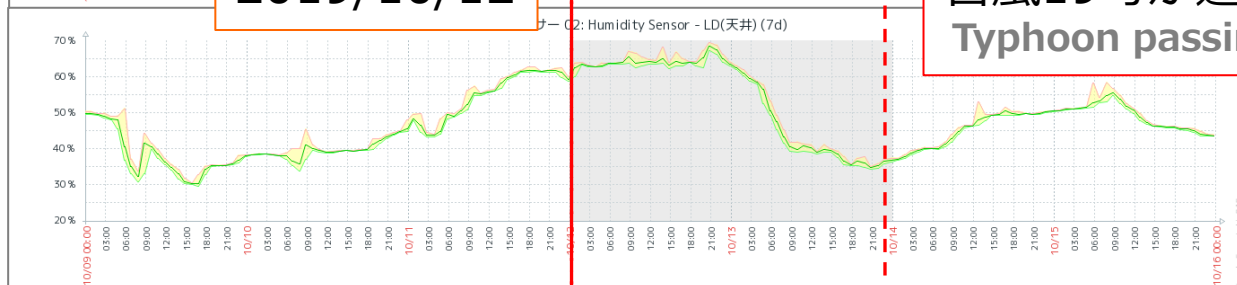




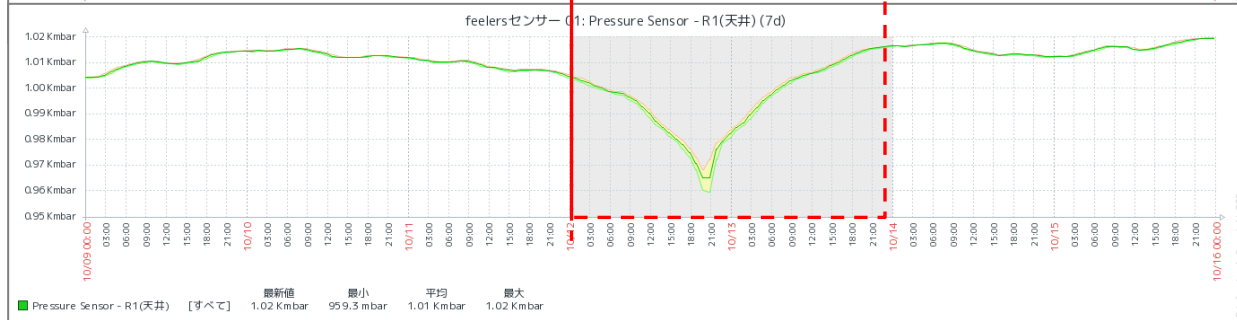
温度



湿度



気圧



データを溜める仕組み

Mechanism for storing data

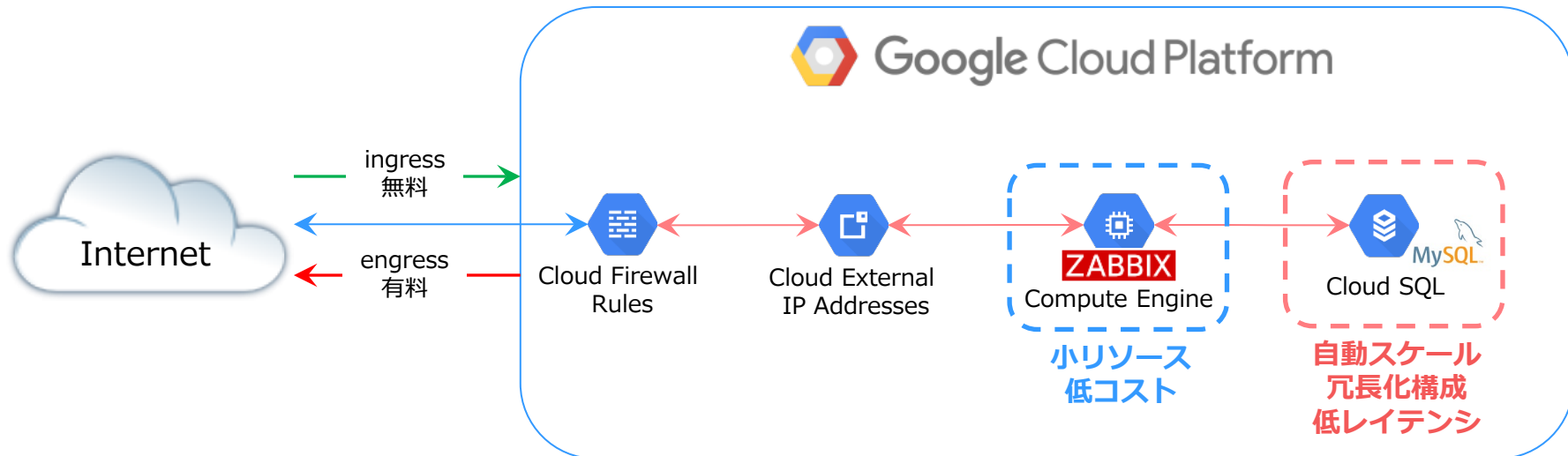
集めた情報保存先には以下の要件が必要となりました。

- ① データ肥大化に対応 ⇒ ストレージサイズの自動スケール
(Response to data enlargement)
- ② データ保持 ⇒ データ保持を優先した冗長化
(Data retention)
- ③ I/O性能の維持 ⇒ データ蓄積時のI/O性能劣化の低減
(I/O performance)

⇒ この要件をオンプレで提供することは厳しいため、
クラウドサービスを使用しました

◆構成のポイント

- ① データベースには**マネージドSQLのサービス**を使用
- ② Zabbix Serverは小リソースのインスタンスを使用して**コスト削減**
- ③ 受信 (ingress) は**通信料金が発生しない**



Zabbix Server用インスタンスは、small (1 共有vCPU、メモリ 1.7GB) を使います。

micro (1 shared vCPU)
0.6 GB memory, f1-micro

small (1 shared vCPU)
1.7 GB memory, g1-small

1 vCPU
3.75 GB memory, n1-standard-1

2 vCPUs
7.5 GB memory, n1-standard-2

4 vCPUs
15 GB memory, n1-standard-4



アイテム	推定費用
共有 vCPU x 1 + メモリ 1.7 GB	\$23.51/月
10 GB の標準の永続ディスク	\$0.52/月
継続利用割引	-\$7.05/月
合計	\$16.97/月

¥1,830/月

Boot disk

New 10 GB standard persistent disk
Image
CentOS 7

※ 2019年10月現在のデータです

micro (1 共有vCPU、メモリ 0.6GB) で構築することもできます。


micro (1 shared vCPU) 0.6 GB memory, f1-micro
small (1 shared vCPU) 1.7 GB memory, g1-small
1 vCPU 3.75 GB memory, n1-standard-1
2 vCPUs 7.5 GB memory, n1-standard-2
4 vCPUs 15 GB memory, n1-standard-4



アイテム	推定費用
共有 vCPU x 1 + メモリ 0.6 GB	\$6.72/月
10 GB の標準の永続ディスク	\$0.52/月
継続利用割引 ⓘ	-\$2.01/月
合計	\$5.22/月

¥560/月

Boot disk ⓘ



New 10 GB standard persistent disk
Image
CentOS 7

※ 2019年10月現在のデータです

最小構成でZabbix Serverを構築すると運用上の問題が・・・

micro (1 shared vCPU)
0.6 GB memory, f1-micro

small (1 shared vCPU)
1.7 GB memory, g1-small

1 vCPU
3.75 GB memory, n1-standard-1

2 vCPUs
7.5 GB memory, n1-standard-2

4 vCPUs
15 GB memory, n1-standard-4

```
# yum install nmap
```

～略～

```
Upgrade 1 Package (+1 Dependent package)
```

```
Total download size: 4.2 M
```

```
Is this ok [y/d/N]: y
```

～略～

```
Error downloading packages:
```

```
2:nmap: [Errno 5] [Errno 12] Cannot allocate memory
```

```
2:nmap-ncat: [Errno 5] [Errno 12] Cannot allocate memory
```

```
# free
```

	total	used	free
Mem:	602,512	345,928	105,968 ← 単位はKB

マネージドSQL用インスタンスはdb-n1-standard1（1vCPU / 3.75GBメモリ）を使います
ディスク容量は初期10GB、自動拡張を有効にします。

Shared-core machines

db-f1-micro
1vCPU, 614.4 MB

db-g1-small
1vCPU, 1.7 GB

Standard machines

db-n1-standard-1
1vCPU, 3.75 GB

db-n1-standard-2
2vCPU, 7.5 GB

Storage capacity ?
10–10230 GB. Higher capacity improves performance, up to the limits set by the machine type. Capacity cannot be decreased later.

10 GB

Enable automatic storage increases
Whenever you're near capacity, space will be incrementally increased. All increases are permanent. [Learn more](#)

hosts	1,150
items	43,000
trigger	500
nvps	650



3ヶ月後

Configuration

vCPUs	Memory	SSD storage
1	3.75 GB	287 GB

Database version is MySQL 5.7

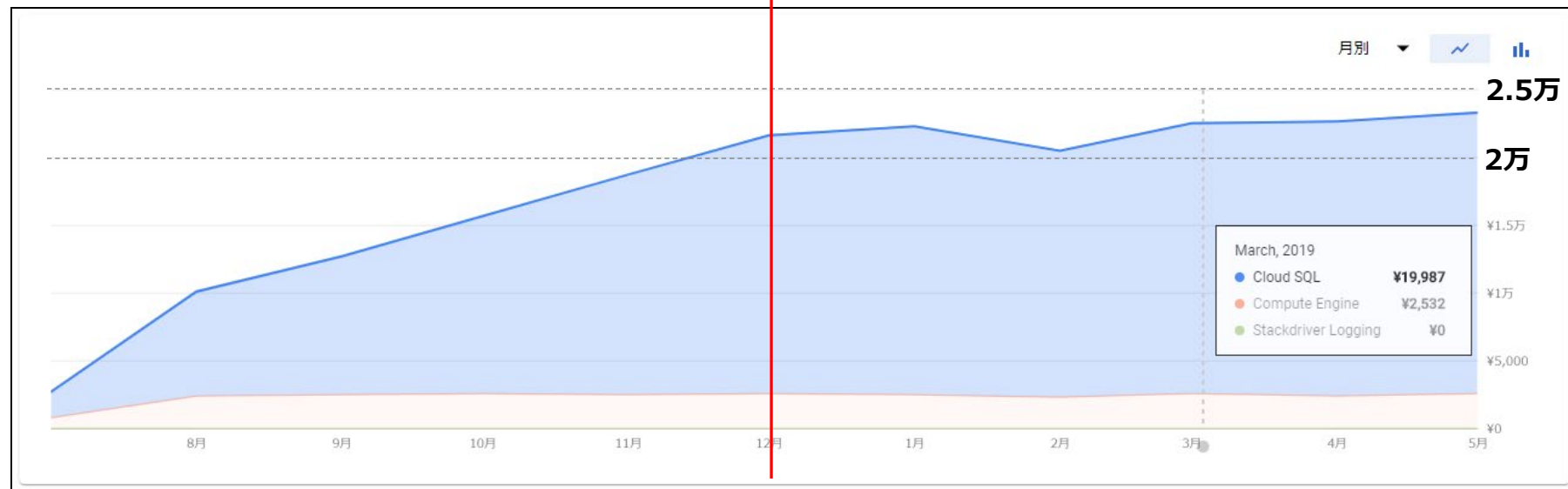
データ保存の費用

Data storage costs

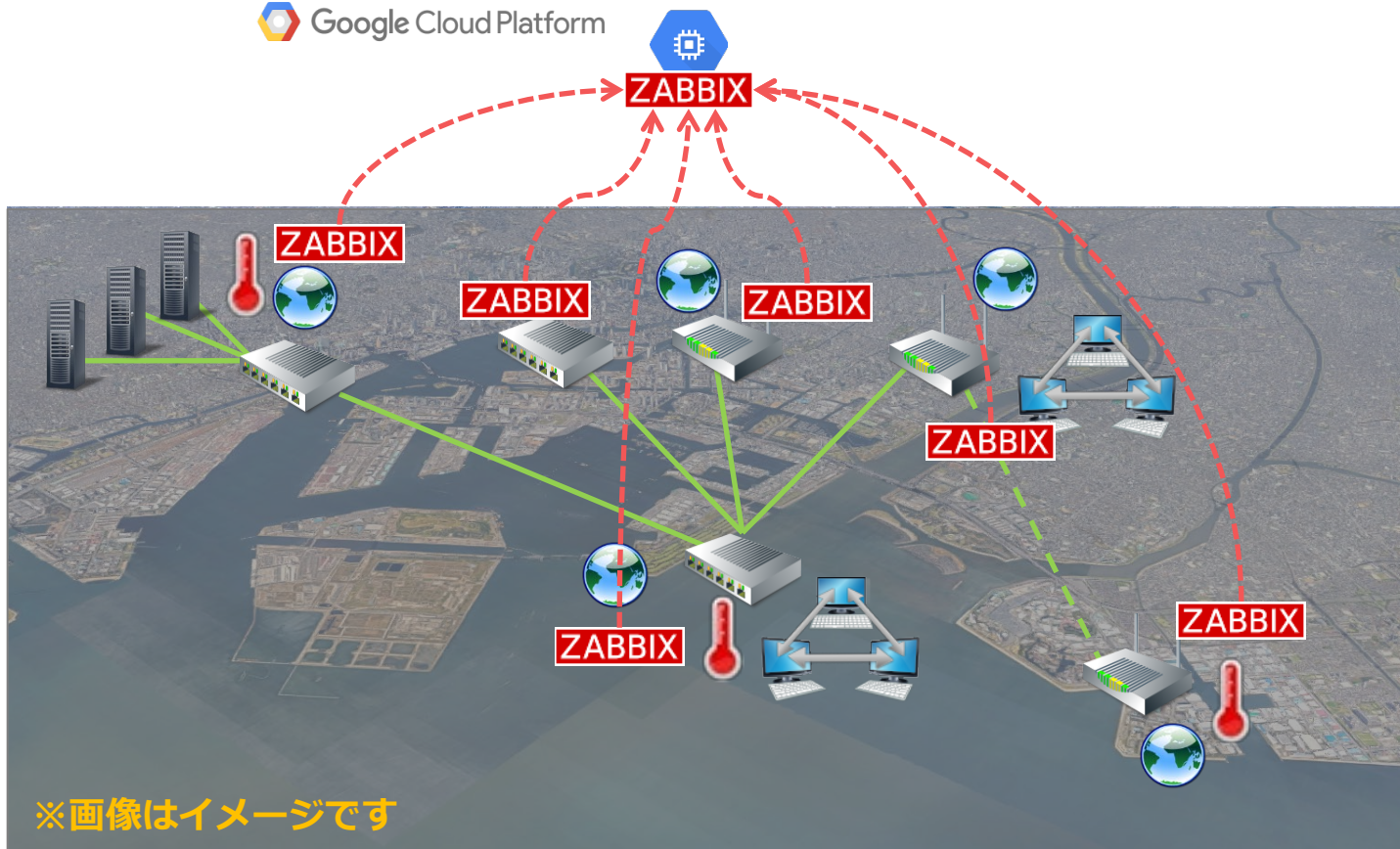
- **650 nvps、ヒストリ 90日、トレンド 365日**でデータを保管
- 最初の3か月はデータ容量と費用は右肩上がり
- 4か月目からは、データ容量はトレンドの増加分のみとなる
- Zabbix Serverのインスタンスと合わせると**2.3万/月**

March, 2019	
● Cloud SQL	¥19,987
● Compute Engine	¥2,532
● Stackdriver Logging	¥0

← **ヒストリ削除なし** → | ← **ヒストリ削除あり** →



Google Cloud Platform



まとめ

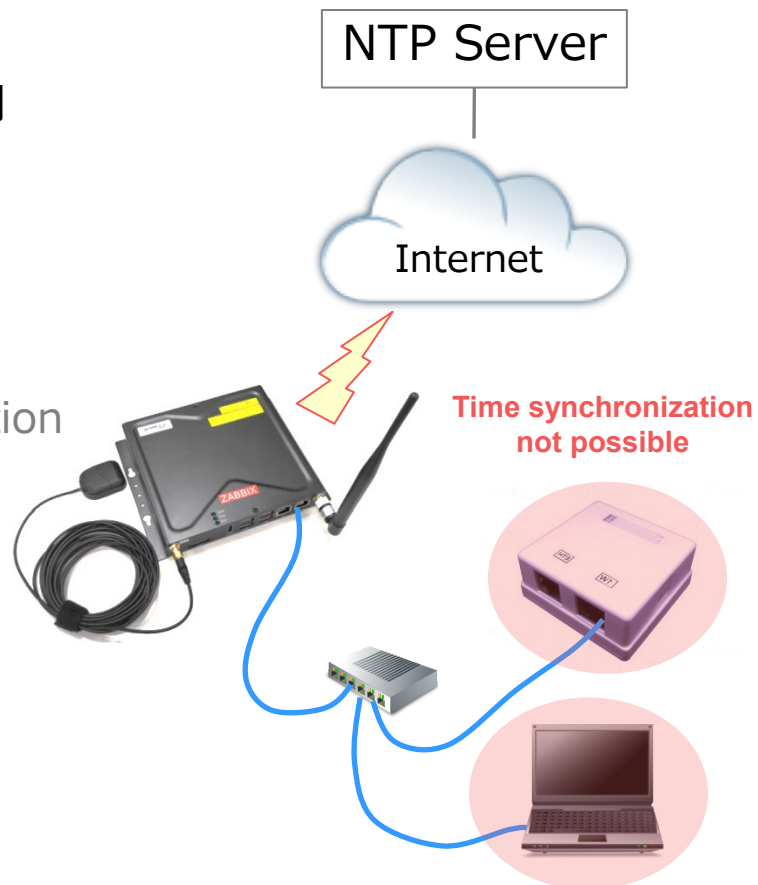
Summary of results

◆状況

- ・インターネットに接続できない監視対象機器の時刻同期
Problems with time synchronization of monitored devices that cannot connect to the Internet.

◆問題点

- ・打刻されていないデータは、Zabbixが受け取らない
Zabbix does not accept data without time information
- ・誤った時刻が打刻されても、Zabbixでは補正できない
Zabbix cannot correct incorrect time information



◆状況

- 簡単な構造の機器では、時刻情報を**保持していない**場合がある
A device with a simple structure may not hold time information
- 時刻情報を保持するには**電池が必要**となる（電池交換の運用が必要となる）
Battery is required to hold time information (battery replacement is required)

◆改善要望（Request for improvement）

- Zabbixが受信時刻をヒストリに打刻する機能が必要
Zabbix needs a function to set the reception time in the history
- アイテム毎に打刻処理を制御できると良い
It is good to be able to control time writing for each item

◆状況

- ホストインベントリに値を格納すると、直近値を参照 / 取得する場合に便利
Storing values in the host inventory is useful for referencing / getting the latest values
- ホストインベントリに格納された値の収集時間が分からない
I do not know the time to collect the values stored in the host inventory

◆改善要望 (Request for improvement)

- ホストインベントリに収集時間を表示してほしい
I want to display the collection time in the host inventory

まとめ

- ・ データ収集ゲートウェイの作成にあたり、Zabbix Japan社より多大なご支援を頂きました。
- ・ 特別仕様のアプライアンスなどに興味がある方は、Zabbix Japan社へご相談下さい。

Zabbixは
任せとけ！



弊社も
よろしく

NTT Com Solutions
an NTT Communications Company

ご清聴ありがとうございました

Thank you!