

# OSS監視ツールの導入失敗事例から学ぶ みんなが知らない正しい導入手順

2017/6/2

ミラクル・リナックス株式会社  
マーケティング本部  
月城史行



- はじめに
- 監視設計の大方針
- よくある落とし穴
- 便利な小技



障害検知

システムのダウンタイム最小化

リソース  
分析

システムのキャパシティプランニング

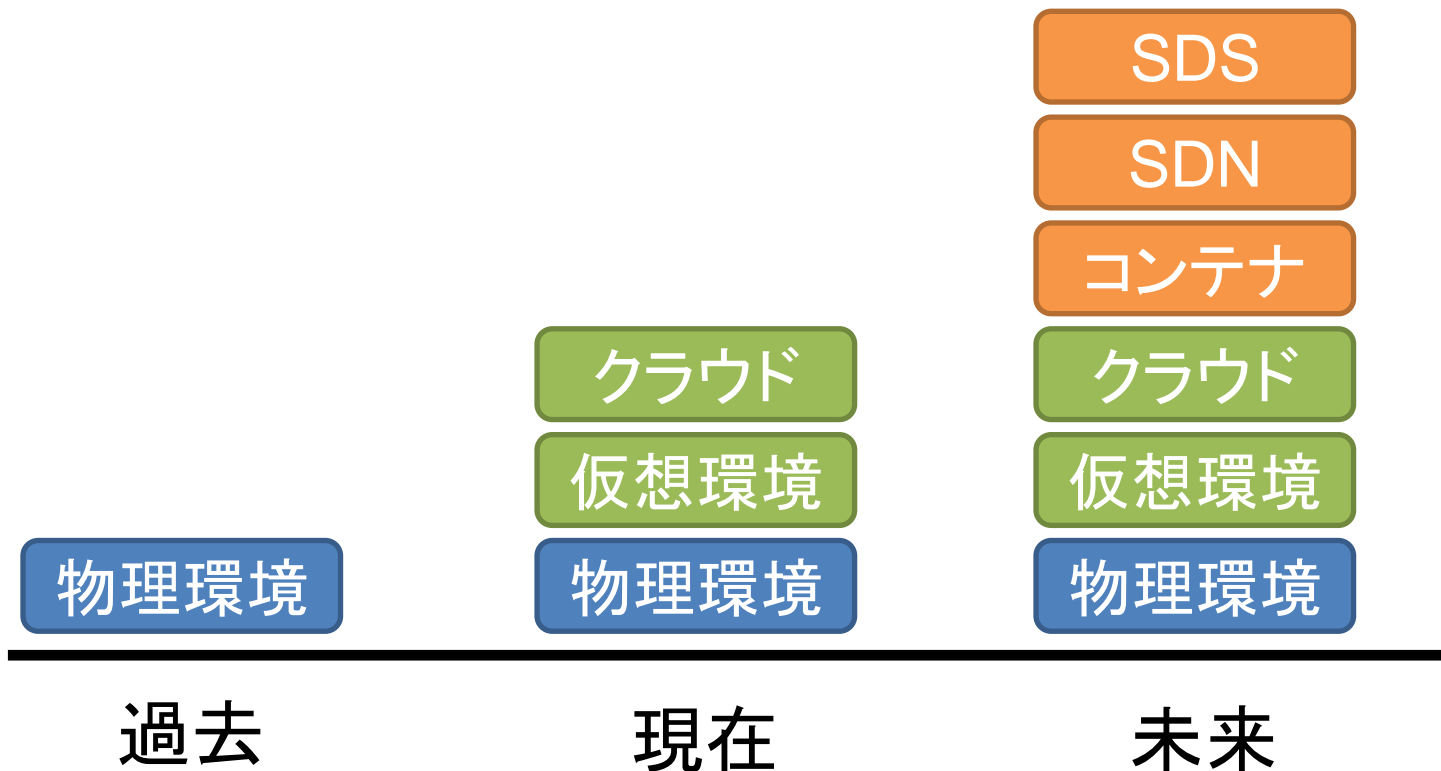
パフォーマンス  
分析

システムのサービスレベル評価

システムを安定運用するには監視が不可欠



# ITインフラの技術変化



増加する監視対象、複雑化するインフラ技術  
ハイパーコンバージド、IoTといった技術要素も登場

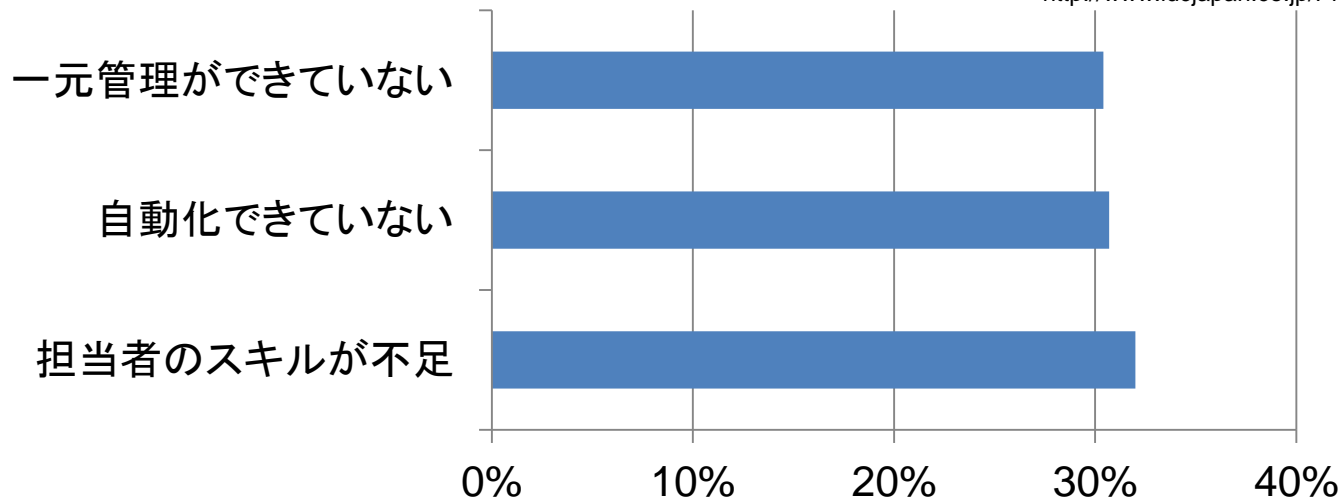


# システム運用管理における課題

\* Source: IDCジャパン

国内企業におけるシステム運用管理実態に関するユーザー調査結果

<http://www.idcjapan.co.jp/Press/Current/20161020Apr.html>



2016年8月、IDCジャパン社が国内企業309社にアンケート。  
「担当者のスキルが不足」が最多回答となった。過去4回の調査においても、スキル不足が最多回答となっており、改善がすすんでいない。



# OSS監視ツールの評価

基本情報

3.0

2.0

1.0

機能

サポート

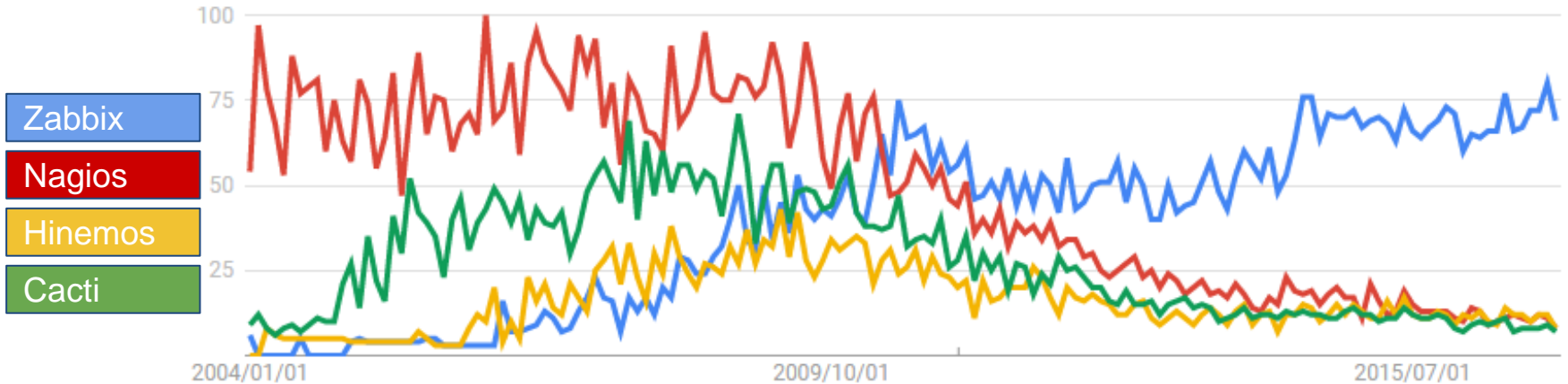
成熟度

	基本情報	サポート	開発の安定性	成熟度	機能
ZABBIX	★★★★★	★★★★★	★★★★☆	★★★★★	★★★★★
Nagios	★★★★★	★★★★★	★★★★☆	★★★★★	★★★★☆
GroundWork Monitor	★★★★★	★★★★★	★★☆☆☆	★★★★☆	★★★★★
Hinemos	★★★★★	★★★☆☆	★★☆☆☆	★★★★☆	★★★★☆
Xymon	★★★★☆	★★★☆☆	★★☆☆☆	★★★★☆	★★★★☆



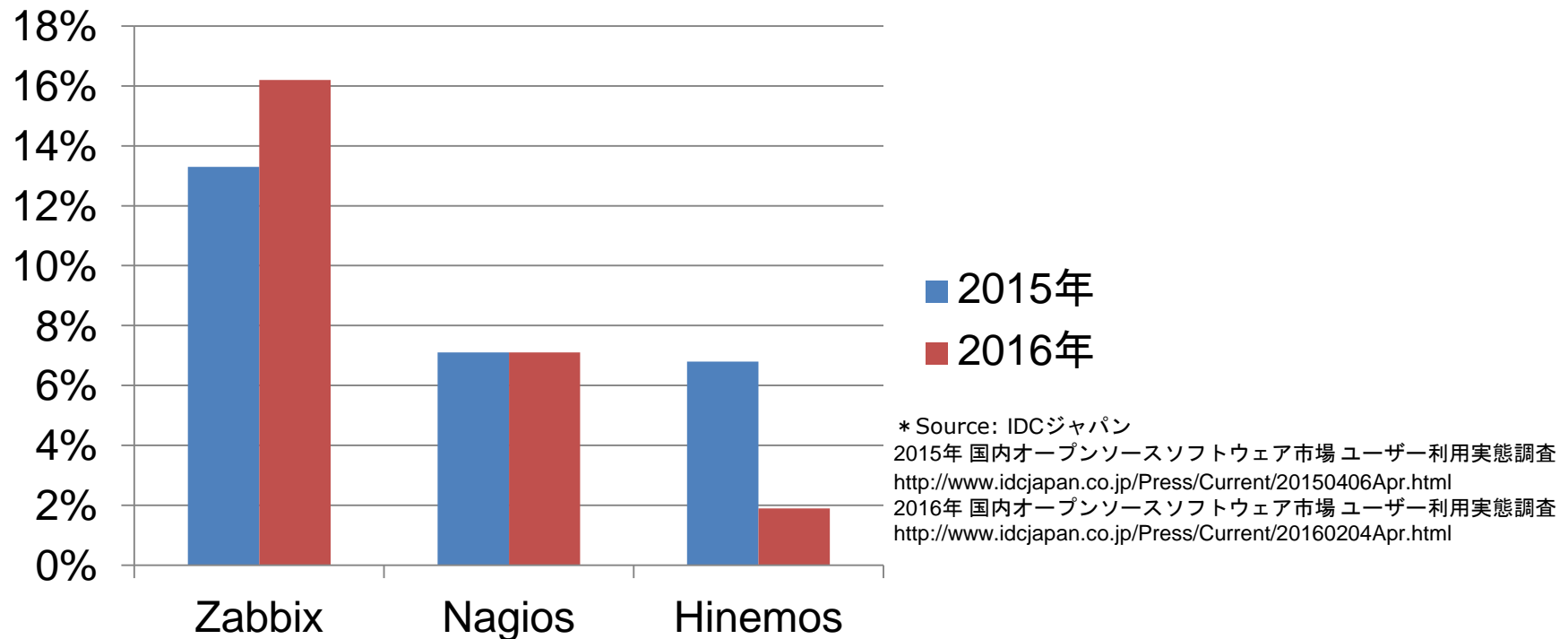
Source: 平成23年6月 独立行政法人 情報処理推進機構  
 社内向けクラウド構築のために活用できる  
 ソフトウェアカタログの作成 調査報告書  
<http://ossipedia.ipa.go.jp/nfs/pdf/pub/1010/209/287/287.pdf>

Googleトレンド（国内）の状況でも、2010頃からNagiosを逆転して首位





# Zabbixのシェア拡大



OSSシステム運用管理ソフトのカテゴリでは  
Zabbixがトップシェアに成長

# 監視設計の大方針





## 「監視」自体にふりまわされない

監視はシステムを安定運用するための手段の一つ。  
監視業務によって、インフラ運用業務に悪影響が  
でるのは本末転倒。

⇒監視にふりまわされないためには？



# 監視にふりまわされないために

## シンプル

サーバ構成、監視設定は極力シンプルに

- Zabbixサーバで他のサービスを動かさない
- 複雑な障害判定条件を設定しない
- 障害通知の設定も少なくする

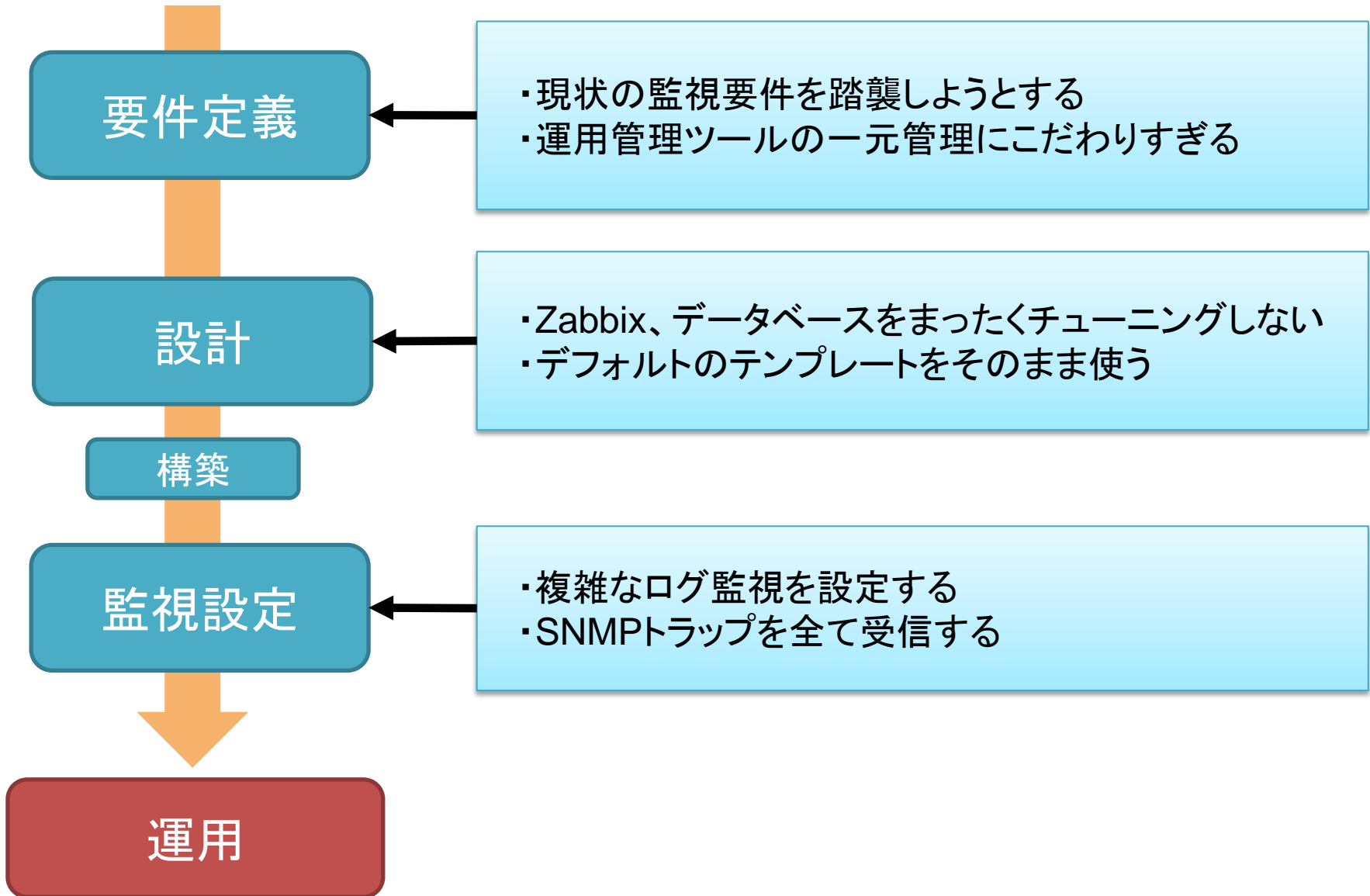
## 適材適所

監視ツールの得意、苦手な点を把握する。  
監視機能だけでなく、非機能要件についても考慮が必要。

- Zabbixはリソース監視が得意だが、ログやSNMPトラップ監視は苦手
- 障害判定式の「トリガー」、障害通知の「アクション」など用語が独特



# よくある落とし穴





# 要件定義の落とし穴1

---

## ■現状の監視要件を踏襲しようとする

「監視要件は現状のまま」は要注意。他の監視ツールと思想、設計、実装が異なるため、同じ監視要件をZabbixで実現できないことも。

### ・監視、障害通知の除外設定

「週末のメンテナンス時間中はサービス監視を停止したい」

「ログに特定のエラーコードが出力されると、対象外としたい」

除外設定は他の監視ツールと差異が大きく、問題になりやすい。

### ・監視異常時の挙動

Zabbixは監視失敗時のリトライ回数の調節はできない。

他の監視ツールでは正常と判断されていた監視対象がZabbixにおきかえると障害と判断されることも。

異常時の挙動は要件定義から漏れやすく、運用後に気付く場合が多い。



# 要件定義の落とし穴2

## ■運用管理ツールの一元管理にこだわりすぎる

既存の運用管理ツールをZabbix一本に統合しようとする苦戦。

機能	Zabbixの適性	代替ツール
リソース監視	◎	
ネットワーク、SNMP監視	◎	
ログ、SNMPトラップ監視	△ 大量の監視、複雑な設定は困難	Fluentdなど
ハードウェア監視	△ SNMP、IPMIで代用	HWベンダーの監視ツール
パフォーマンス分析 (Web、データベース)	△ 基本的な監視のみ	対象アプリに特化したツール(APMなど)
インシデント管理	△ イベントでのコメント機能のみ	Redmineなど
ジョブ管理	×	JP1など



# 設計の落とし穴1

- Zabbix、データベースをまったくチューニングしない  
デフォルト設定ではパフォーマンスが出にくい。

- **Mariadb**

```
innodb_buffer_pool_size = 物理メモリの半分  
innodb_log_file_size = 256MB  
innodb_log_files_in_group = 2
```

物理メモリが少ない場合、パラメータが以下の式となるように調整

$$\text{innodb\_log\_file\_size} * \text{innodb\_log\_files\_in\_group} < \text{innodb\_buffer\_pool\_size}$$

- **Zabbix**

```
StartPollers = 10以上  
StartTrappers = 10以上
```

Zabbixインターナルアイテムで負荷状況を見ながらパラメータを修正。  
デフォルトで設定されてある「Template App Zabbix Server」に  
パフォーマンス関連の監視アイテムが定義されてある。



# 設計の落とし穴2

---

## ■ デフォルトのテンプレートをそのまま使う

Zabbixではデフォルトの監視テンプレートがOS別に定義されている。リソース監視以外の監視アイテムも含まれているので、再設定を検討。

### ・ パフォーマンス以外の監視アイテム

/etc/passwdのチェックサム、ホスト名、システムのuptimeなど  
不要なら削除

### ・ agent.ping

Zabbixエージェントの死活判定にはシンプルチェックを推奨。  
詳細は弊社のブログにて公開中。

<https://www.miraclelinux.com/tech-blog/835fku>

### ・ 監視間隔

デフォルトではCPU、メモリなどは1分間隔で監視。  
重要度が低ければ、5分や10分間隔に変更。



# 監視設計の落とし穴

---

## ■ 複雑なログ監視を設定する

Zabbixの仕様上、検知したくない文字列を設定するのは困難。  
正規表現を駆使して実現できなくもないが、可読性も保守性も低い。

### ⇒他の監視アイテムを検討する

- ・ ログ監視の代わりにプロセス監視、サービス監視を使う
- ・ Zabbixエージェントでコマンド(スクリプト)を実行するユーザパラメータを使う。

## ■ SNMPトラップを全て受信する

SNMPトラップ監視はパフォーマンスが悪く、柔軟な設定もできない。

### ⇒SNMPトラップの数を減らす

- ・ 必要最低限のSNMPトラップを送出する設定に変更。
- ・ SNMPポーリングで監視できる機器は、SNMPポーリング監視に統一。





# 落とし穴を避けるために

---

## ■ 移行時には監視要件を見直す

「現状踏襲」は複雑になりやすいのでさける。Zabbixの得意、苦手な点を把握して要件を整理。除外設定や監視異常時の挙動にも隠れた要件があるので要注意。

## ■ シンプル、スモールスタート

初めから複雑な監視設定で運用しない。最低限の監視設定からスタートして必要に応じて監視設定を増やしていく。  
ただし、最低限の設定であってもチューニングを推奨。

## ■ Zabbixと他のツールとの連携

Zabbixが苦手とする分野は他のツールとの連携で解決できることも。  
当然、運用管理ツールが増えることによる管理コスト増大とのトレードオフとなる。

# 落とし穴を避ける便利な小技



# 他のツールとの連携用のコマンド

## ■ zabbix\_get

Zabbixエージェントと通信して、情報を取得するツール。  
動作確認、監視障害時の問題切り分けに利用できる。

```
$ zabbix_get -s [IPアドレス] -p [ポート番号] -k [アイテムキー]
```

```
$ zabbix_get -s 127.0.0.1 -k system.cpu.num  
2
```

動作確認、監視障害時の問題切り分けに利用できる。

Zabbix 3.0以降、ZabbixサーバとZabbixエージェント間の通信暗号化機能が実装されたが、zabbix\_getもオプションで暗号通信可能。

## ■ zabbix\_sender

Zabbixサーバと通信して、監視情報を送信するツール。  
事前にZabbixサーバで監視アイテムの設定が必要。

```
$ zabbix_get -s [IPアドレス] -s [ホスト名] -k [アイテムキー] -o 値
```



# チューニング時に役立つ監視アイテム

- 「Template App Zabbix Server」で定義されている監視アイテム

監視アイテム名	Zabbix_server.conf
Zabbix busy discoverer processes, in %	StartDiscoverers
Zabbix busy escalator processes, in %	StartEscalators
Zabbix busy history syncer processes, in %	StartDBSyncers
Zabbix busy http poller processes, in %	StartHTTPPollers
Zabbix busy icmp pinger processes, in %	StartPingers
Zabbix busy ipmi poller processes, in %	StartIPMIPollers
Zabbix busy java poller processes, in %	StartJavaPollers
Zabbix busy poller processes, in %	StartPollers
Zabbix busy snmp trapper processes, in %	StartSNMPTrapper
Zabbix busy trapper processes, in %	StartTrappers

基本的には、監視アイテムと設定値は同じ。  
例外はhistory syncerとDBsyncers。



オープンソースで人々を幸せ  
に

～ イノベーションで未来をつくる ～

ソフトバンク・テクノロジー グループ



ソフトバンク・テクノロジー



エムソリューションズ



フォントワークス



環



サイバートラスト



ミラクル・リナックス



モードツー



アソラテック



リデン