

Zabbix 6.0 新機能解説

SRA OSS, Inc. 日本支社
OSS 事業本部 基盤技術グループ
赤松 俊弘



whoami

- ☑ 赤松 俊弘 (Toshihiro Akamatsu)
- ☑ SRA OSS, Inc. 日本支社
OSS 事業本部 基盤技術グループ
- ☑ Zabbix 認定プロフェッショナル

職務

- ☑ PostgreSQL 以外の OSS 全般の技術サポート、構築
- ☑ 主に Zabbix を担当

アジェンダ

1. Zabbix とは
2. Zabbix 6.0 新機能解説
 1. Zabbix 6.0 について
 2. Zabbix 6.0 の新機能
3. アップグレード時の注意点

Zabbix とは

Zabbix とは



IT インフラやサービス、アプリケーションの可用性や性能を監視するための
エンタープライズ向け統合監視ソフトウェア

OPEN

オープンソースソフトウェアとして開発・公開されており、無償で利用可能



開発元はラトビア共和国の Zabbix LLC

1

Zabbix 1.0 は 2004/3/23 リリース

コンポーネント



Zabbix server

- 全データ集約・保存
- 障害検知・通知



Zabbix Java gateway

- Java アプリケーション監視用デーモン
- JMX を利用



Zabbix agent

- 監視データ収集
- パッシブ・アクティブチェック



Zabbix sender

- コマンドラインユーティリティ
- コマンドラインから監視データ送信



Zabbix proxy

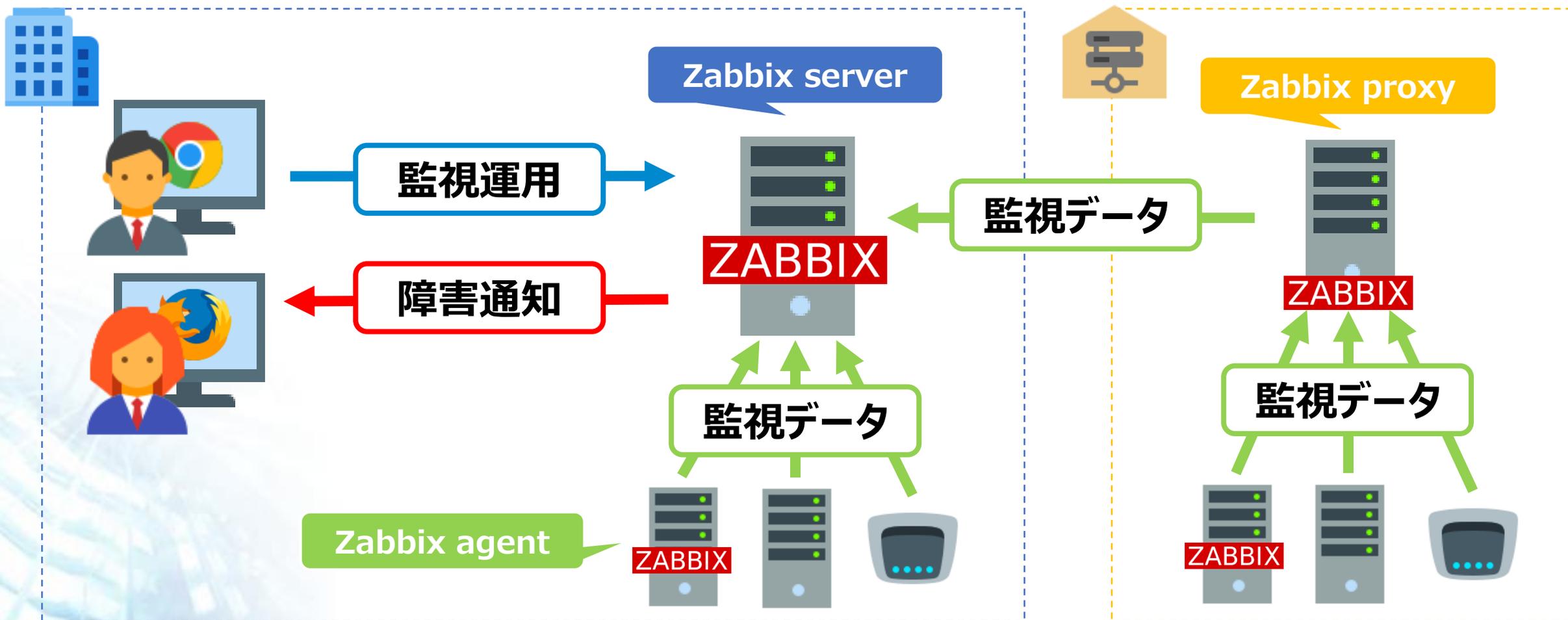
- 監視データ収集の中継
- 負荷分散・集中監視



Zabbix get

- コマンドラインユーティリティ
- エージェントから任意の監視データを取得

アーキテクチャ



対応プラットフォーム

Zabbix server/proxy

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD
- HP-UX
- Mac OS X
- Solaris

Zabbix agent

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD
- HP-UX
- Mac OS X
- Solaris
- Windows

Zabbix agent 2

- Linux
- Windows

監視メトリクス（アイテム）

エージェント監視

リソース監視

- CPU
- メモリ
- ネットワーク
- ファイルシステム

ポート監視

ログ監視

プロセス/サービス 監視

Web 監視

エージェントレス監視

ICMP 監視

DB 監視

SNMP 監視

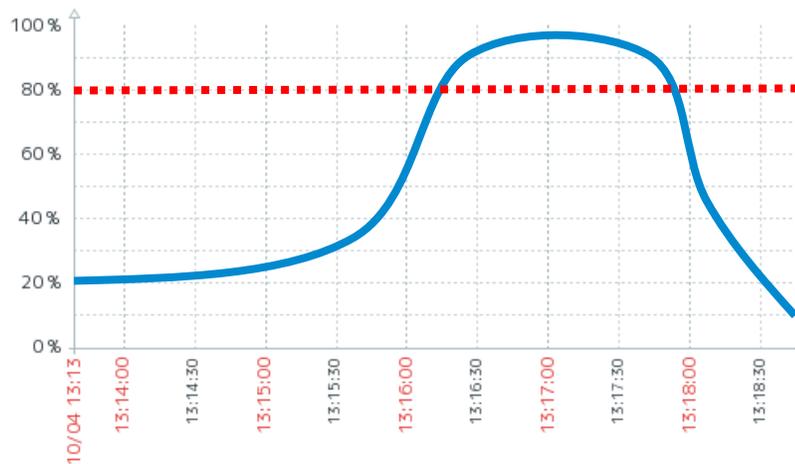
Java 監視

SSH/Telnet 監視

VMware 監視

障害検知 (トリガー)

閾値

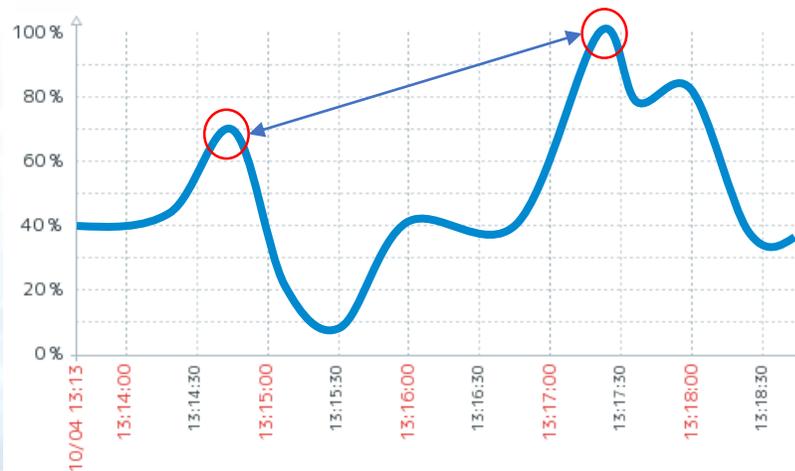


文字列

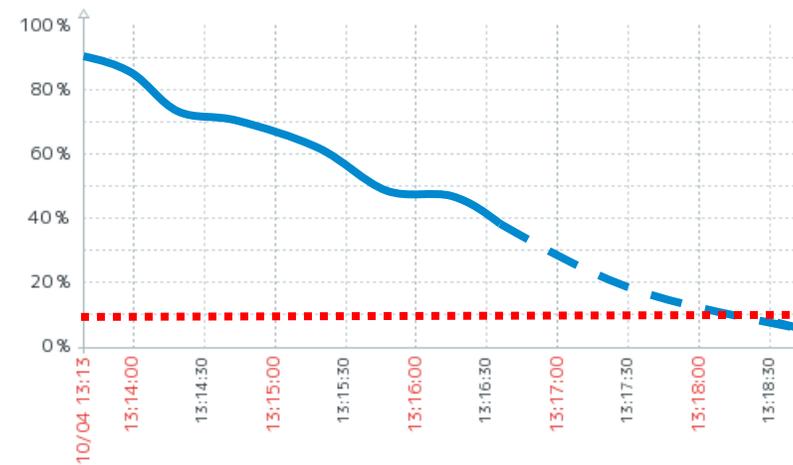
2022 02/14 09:13:25 [INFO] ...
 2022 02/14 09:15:10 [ERR] ...
 2022 02/14 09:21:47 [INFO] ...



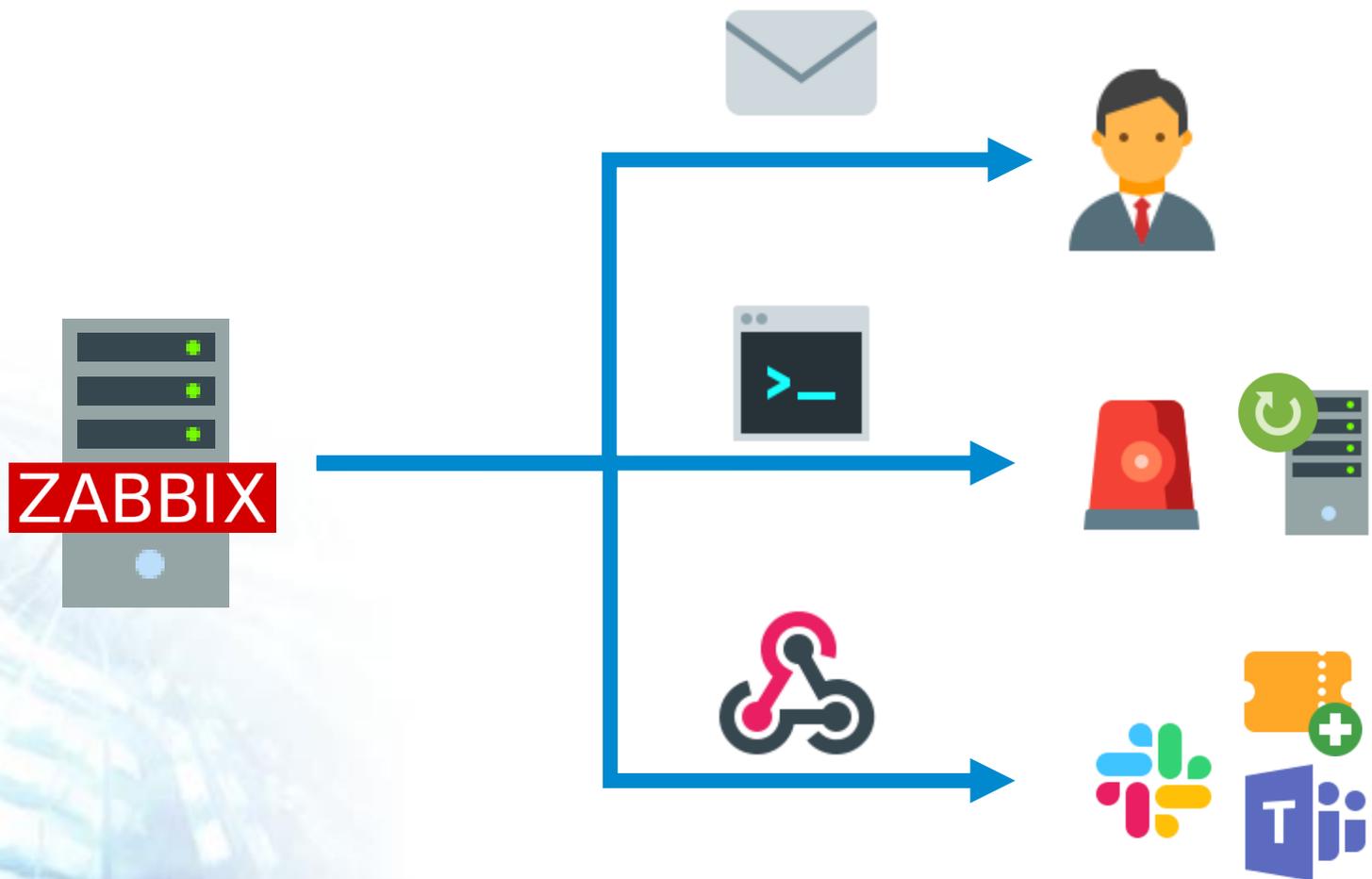
過去との比較



予測値



障害通知 (アクション)



担当者にメール

HTML メールにも対応

コマンド実行

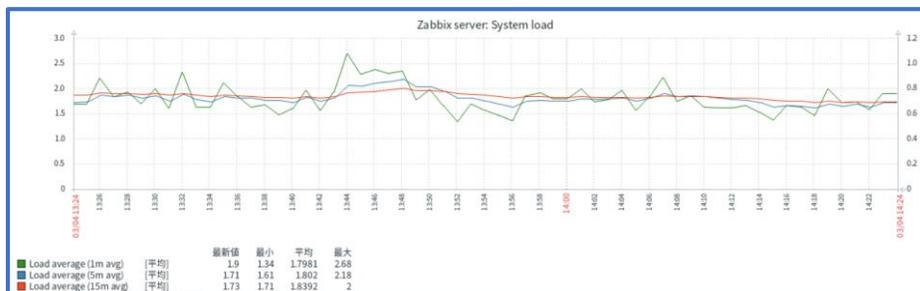
サーバやプロセスの再起動
パトランプの点灯

Webhook

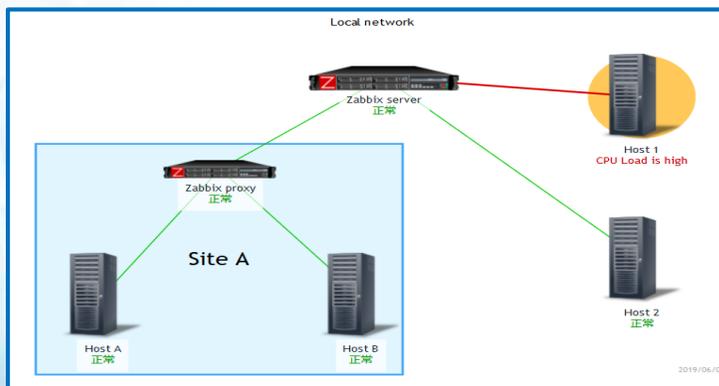
外部アプリとの連携
Slack, Teams, Redmine etc.

データの可視化

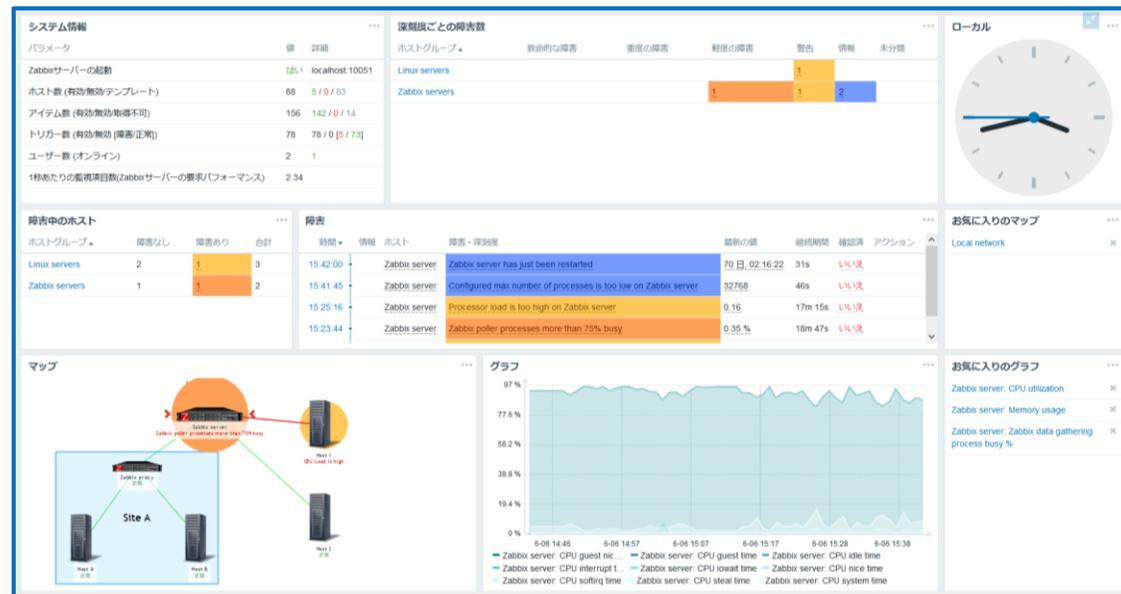
グラフ



ネットワークマップ



ダッシュボード



Zabbix 6.0 新機能解説

Zabbix 6.0

- 2022/02/15 リリース
- 長期サポート（LTS）バージョン
 - フルサポート 3 年、リミテッドサポート 2 年



Zabbix 6.0 対応データベース

DBMS	バージョン
MySQL/Percona	8.0.X
MariaDB	10.5.00 ~ 10.6.X
PostgreSQL	13.0 ~ 14.X
TimescaleDB	2.0.1 ~ 2.3
Oracle	19.c ~ 21.c
SQLite	3.3.5 ~ 3.34.X

Zabbix 6.0 の新機能



ビジネスサービス監視



Zabbix server の
HA クラスタ



アノマリー検知と
ベースライン監視



Kubernetes 監視



新規ウィジェット

ETC その他の新機能

ビジネスサービス監視

Zabbix のサービス監視とは

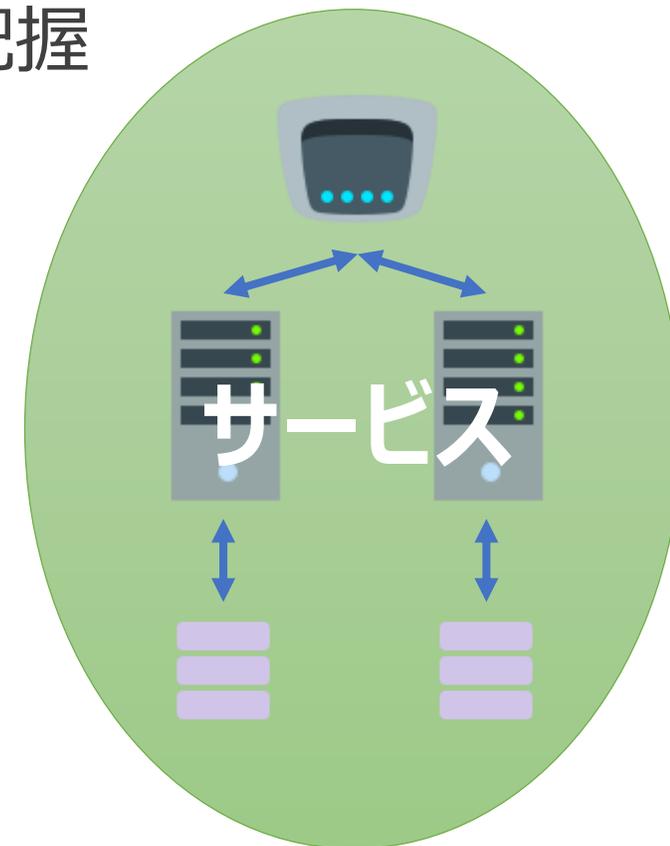
- 監視対象のインフラをハイレベル（ビジネスレベル）で把握

IT サービスの可用性

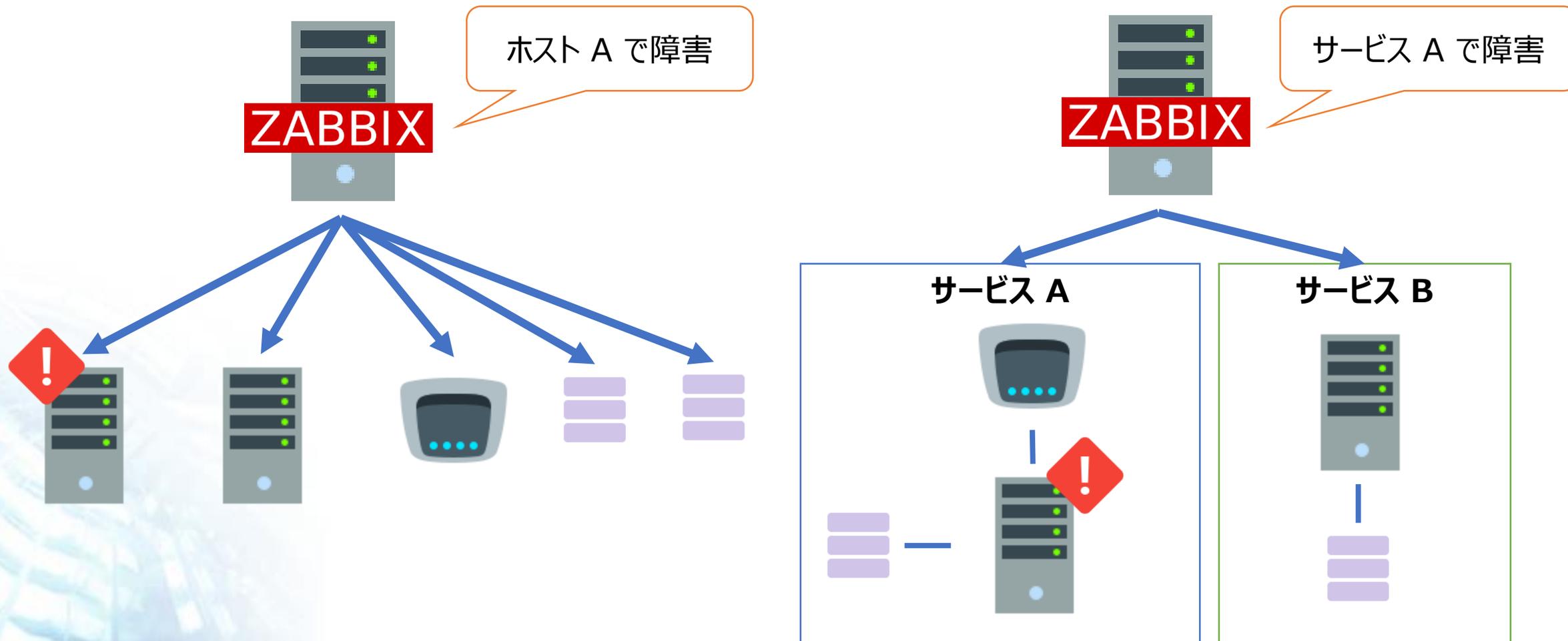
IT サービスの SLA

IT インフラの構造

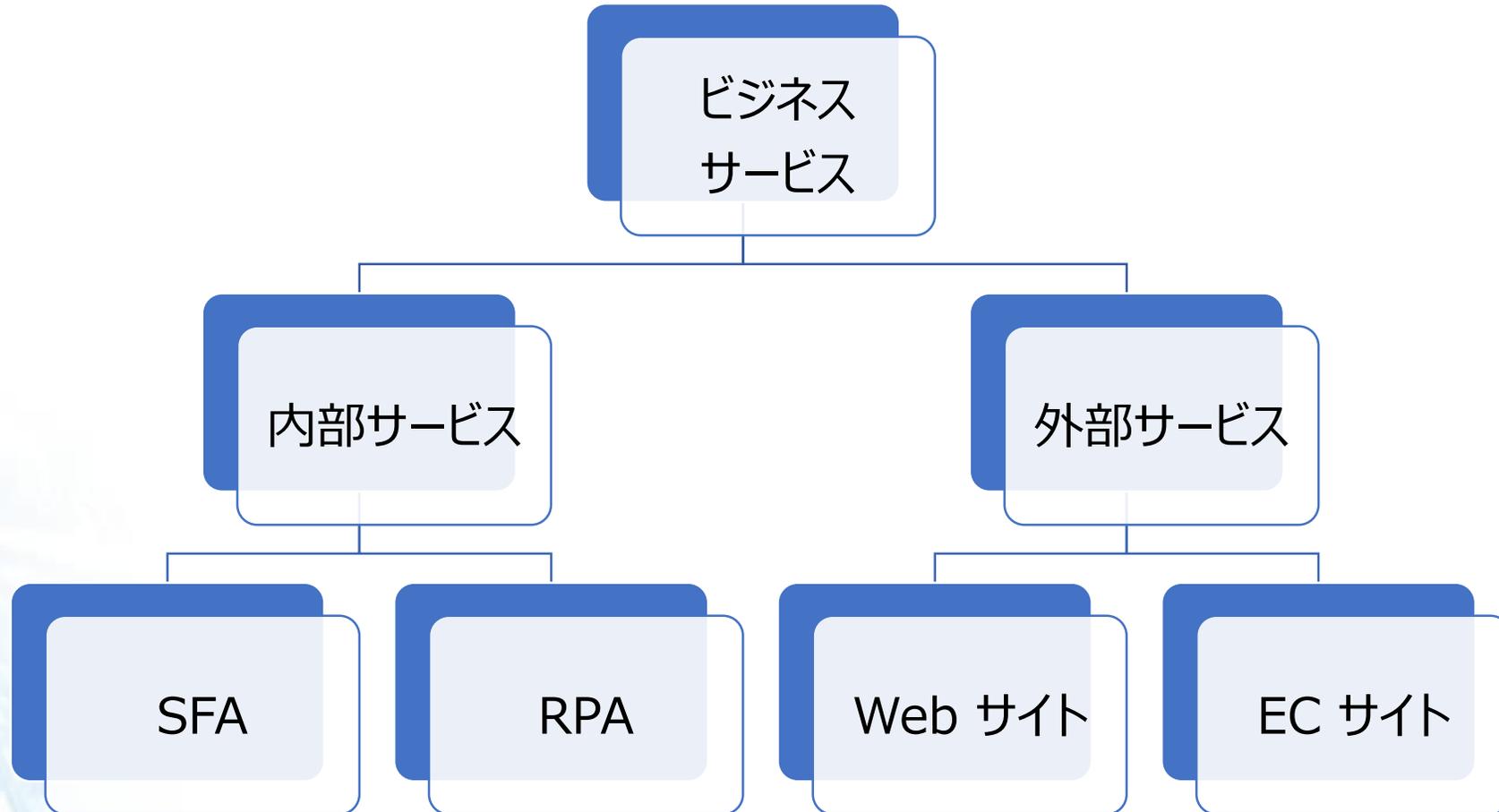
IT インフラの弱点特定



サービス監視のアーキテクチャ



サービスの階層化



サービス監視の機能更新

- ビジネスレベルでのサービス監視を実現
 - タグによる障害の紐づけ
 - 柔軟なステータス計算および伝播
 - サービスのステータス変化によるアラート通知
 - 根本原因の表示
 - サービスへの権限付与

タグによる障害の紐づけ

- サービスへの障害の紐づけがトリガーからタグに変更
- サービスと障害の関係が 1:1 から 1:N に
 - 冗長化されているサービスなど複雑な構成に対応可

サービス

サービス SLA ● タグ 子サービス

* 名前 DB server

親サービス Corporate Web site ✕ 選択
検索文字列を入力

障害タグ

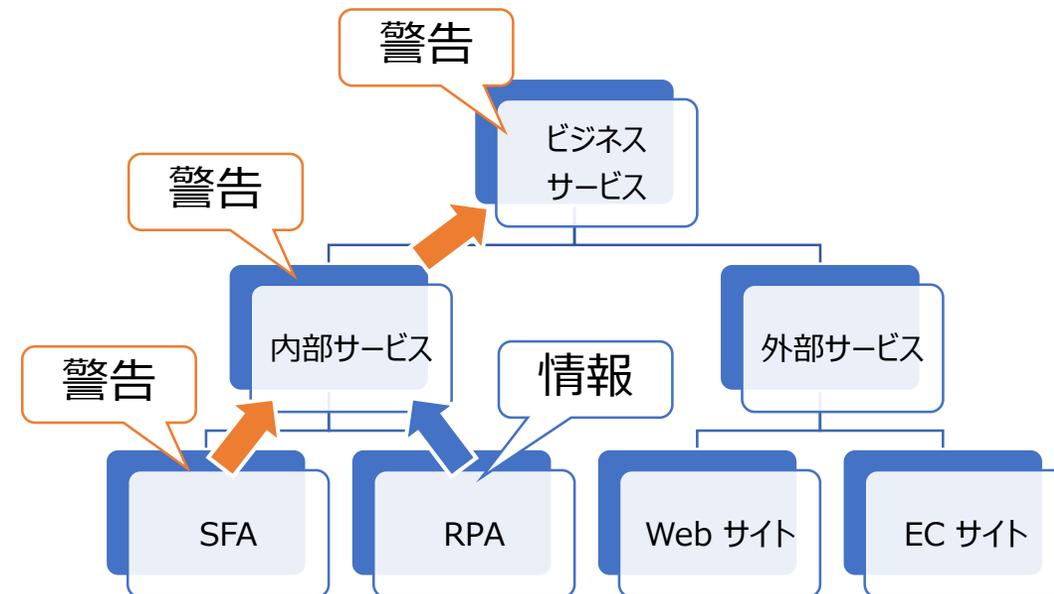
名前	処理内容	値	アクション
Service	等しい ▼	Corp Web site	削除
Server	等しい ▼	DB	削除

[追加](#)

柔軟なステータス計算および伝播 (1)

- 子サービスのステータスから親サービスへのステータスを計算する方法

ステータス計算ルール	説明
子サービスで最も重要	子サービスの障害のうち最も高い深刻度を親サービスのステータスとする
すべての子が障害であったときに最も重要	すべての子サービスが障害であった場合に、その障害のうち最も高い深刻度を親サービスのステータスとする
ステータスを正常に設定	ステータスは計算せず常に正常



柔軟なステータス計算および伝播 (2)

- さらに柔軟なステータス計算のための追加ルール
 - ルールに当てはまる場合に親に伝播する深刻度も合わせて指定

追加のルール

少なくとも **N** 個の子サービスの状態が**ステータス以上**

少なくとも **N%** の子サービスの状態が**ステータス以上**

N 個未満の子サービスのステータスが**ステータス以下**

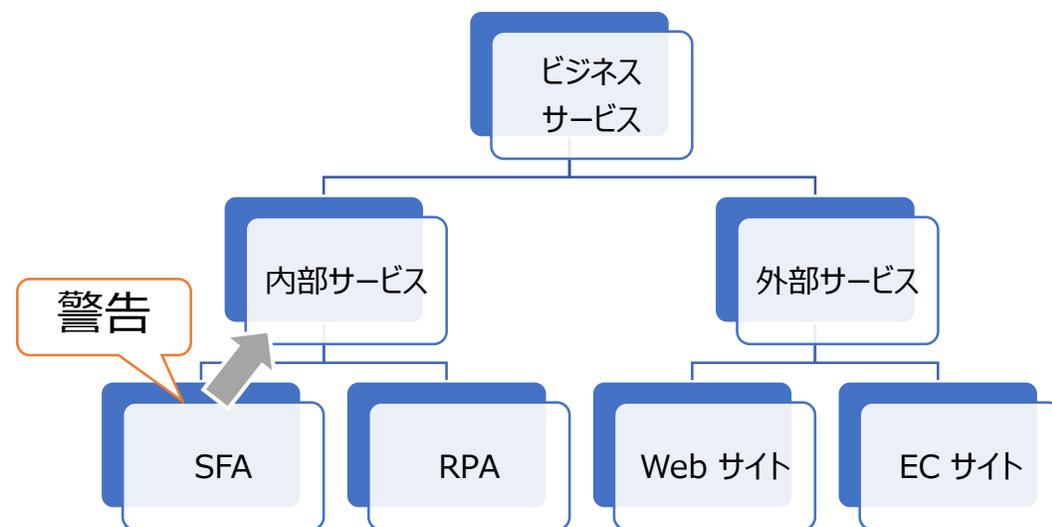
子サービスの **N%** 未満のステータスが**ステータス以下**

ステータスが**ステータス以上**の子サービスの重みが少なくとも **W**

ステータスが**ステータス以上**の子サービスの重みが少なくとも **N%**

ステータスが**ステータス以下**の子サービスの重みが **W** 未満

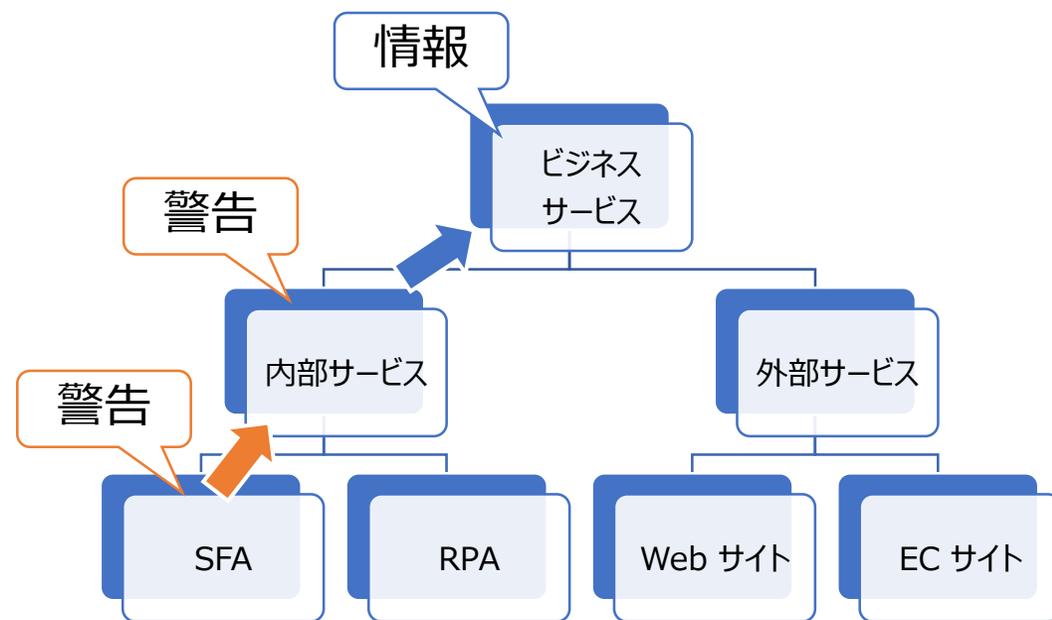
ステータスが**ステータス以下**の子サービスの重みが **N%** 未満



柔軟なステータス計算および伝播 (3)

- ステータス伝播時のルールも詳細に設定可能

ステータス伝播ルール	説明
そのまま	子サービスのステータスをそのまま伝播
深刻度を増やす	深刻度を 1 ~ 5 増やして伝播
深刻度を減らす	深刻度を 1 ~ 5 減らして伝播
このサービスが無視	ステータスを伝播しない
固定ステータス	常に同じステータスを伝播



サービスのステータス変化によるアラート通知

- トリガーによる障害と同様にサービスのステータス変化時にもアクションによる通知が可能

サービスアクション ▾

名前 ステータス

<input type="checkbox"/> 名前 ▲	実行条件	実行内容
<input type="checkbox"/> Corporate Web site action	サービス 等しい Corporate Web site	ユーザーグループにメッセージを送信: Zabbix administrators via Email

根本原因の表示

- サービスのステータスに影響を与えている障害のリストを「根本原因」列に表示

<input type="checkbox"/> 名前	ステータス	根本原因	SLA
<input type="checkbox"/> Corporate Web site 2	重度の障害	PostgreSQL is down	99.9990
<input type="checkbox"/> ERP 3	正常		99.9000
<input type="checkbox"/> SFA 2	正常		99.9000



根本原因に表示される障害は {SERVICE.ROOTCAUSE} マクロで取得可能

サービスへの権限付与

- ユーザロール単位でサービスへの Read-Write 権限が付与可能
 - 個々のサービス、またはタグによる指定が可能
 - サービスのマルチテナント対応

サービスへのアクセス

サービスの表示/設定 なし すべて サービスのリスト

検索文字列を入力

タグとサービスの表示/設定

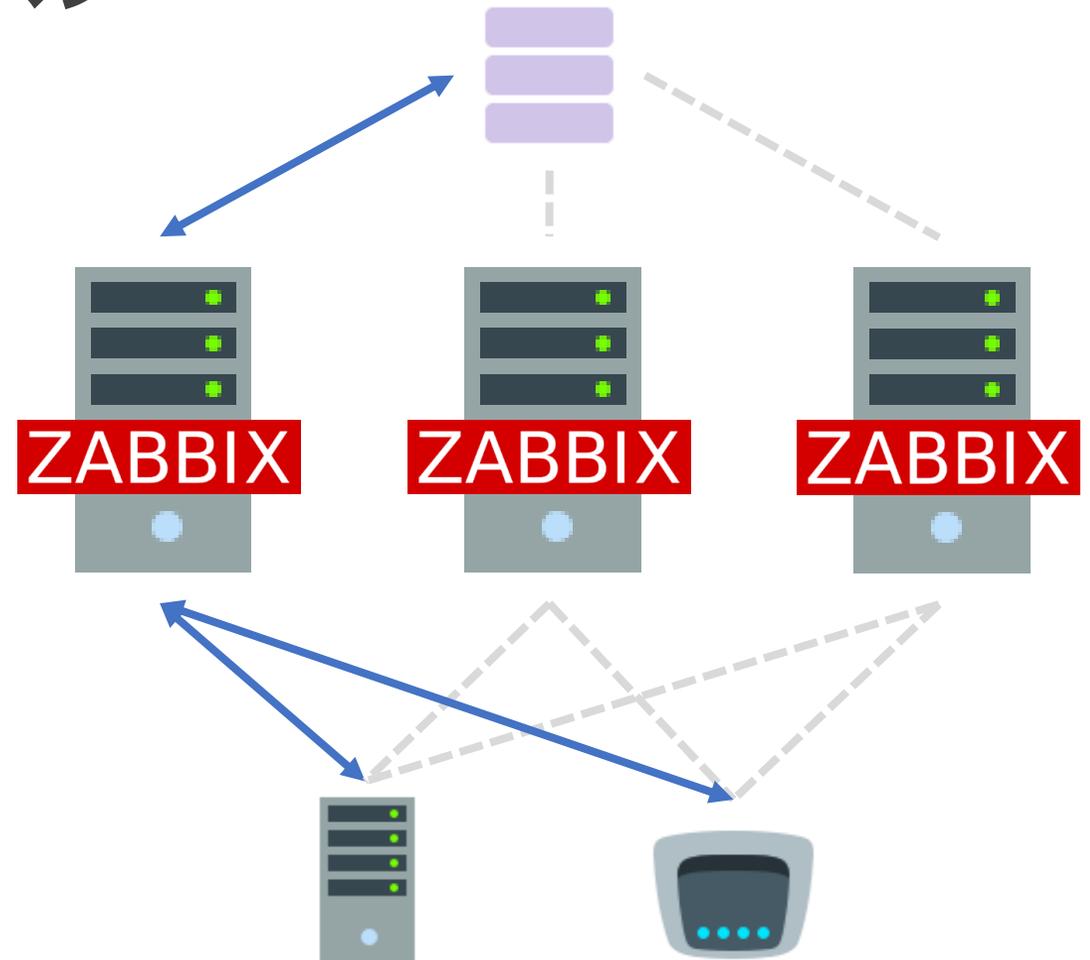
サービスの表示のみ なし すべて サービスのリスト

Zabbix server の HA クラスタ

Zabbix server の HA クラスタ

- Active/Standby
- 1 つの DB を共有
- マイナーバージョン間での互換性あり

詳細は次のセッション
「Zabbix クラスタ構成解説」で

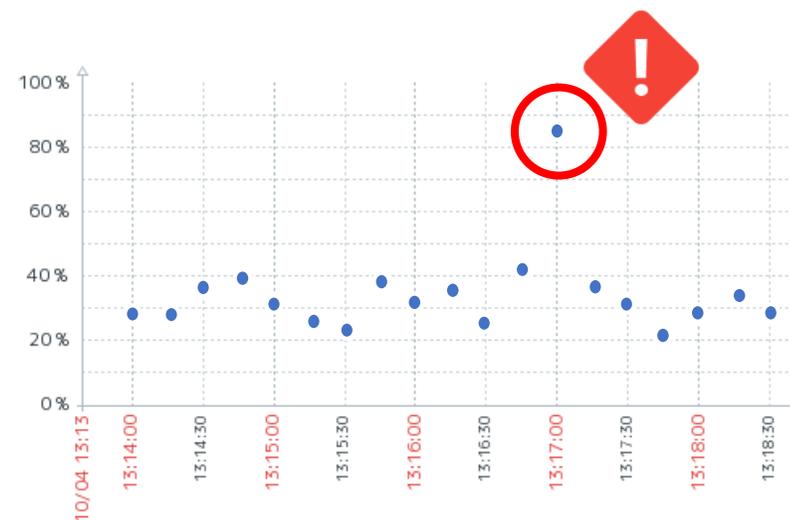


アノマリー検知とベースライン監視

アノマリー検知とベースライン監視

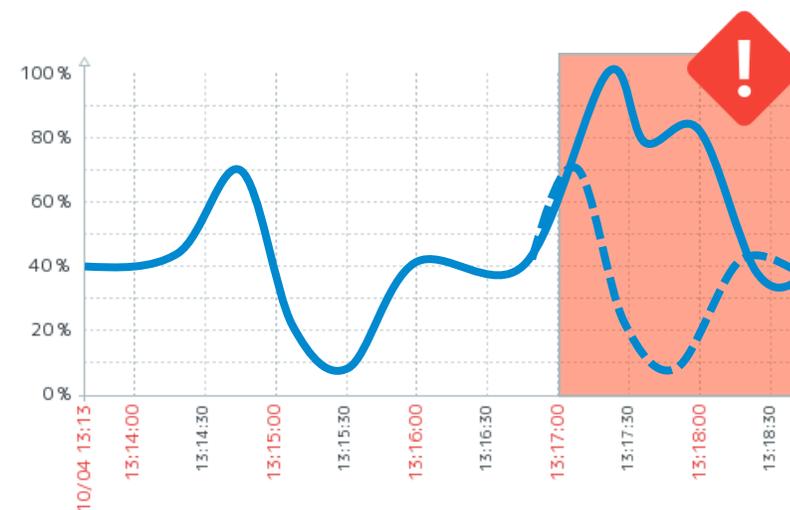
アノマリー検知

過去のデータと比べて大きく異なる値を検知



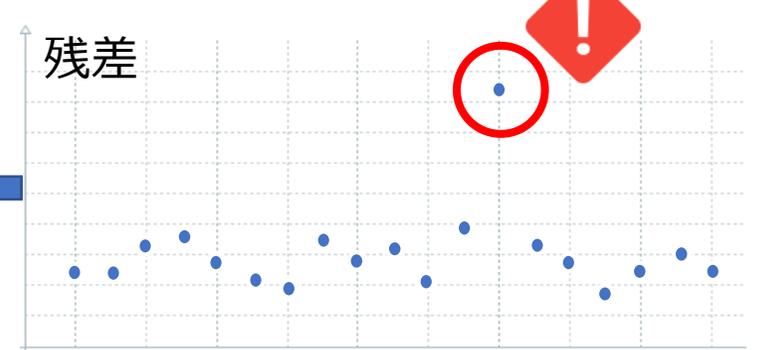
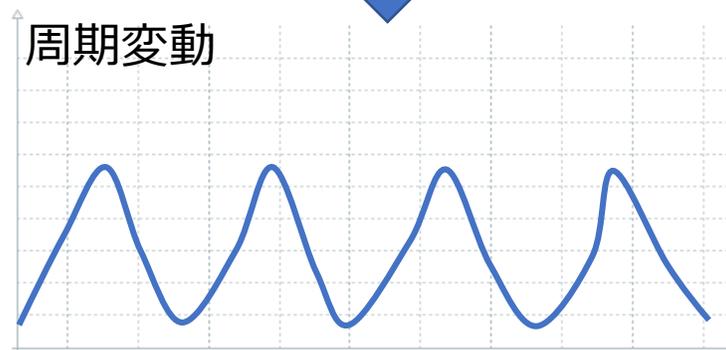
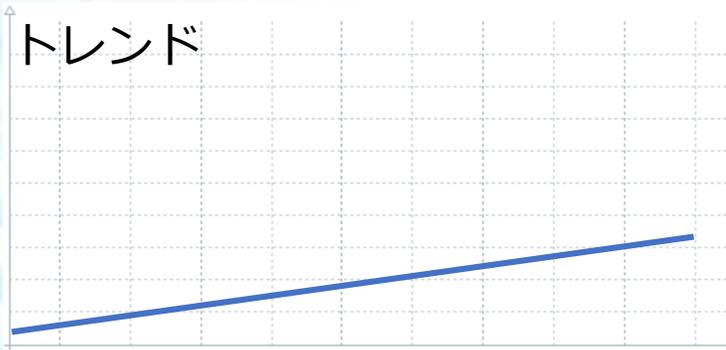
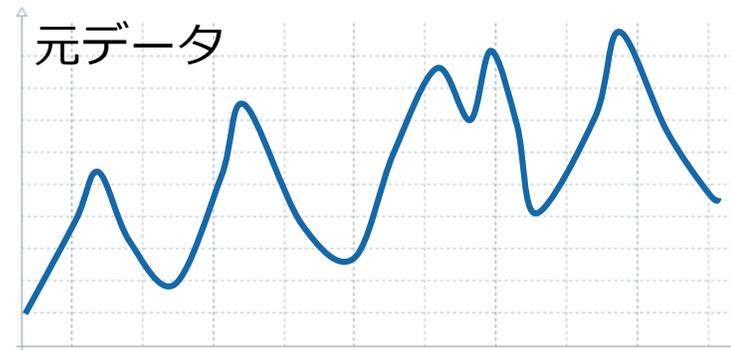
ベースライン監視

過去のデータの周期性から外れたデータを検知



アノマリー検知

STL 分解で元データをトレンド、周期変動、残差に分解し異常値を検知



アノマリー検知のトリガー関数

- `trendstl(/host/key,eval period:time shift, detection period,season,<deviations>,<devalg>,<s_window>)`
 - 全体のデータのうち異常値の数の割合を0~1の間の値で返す

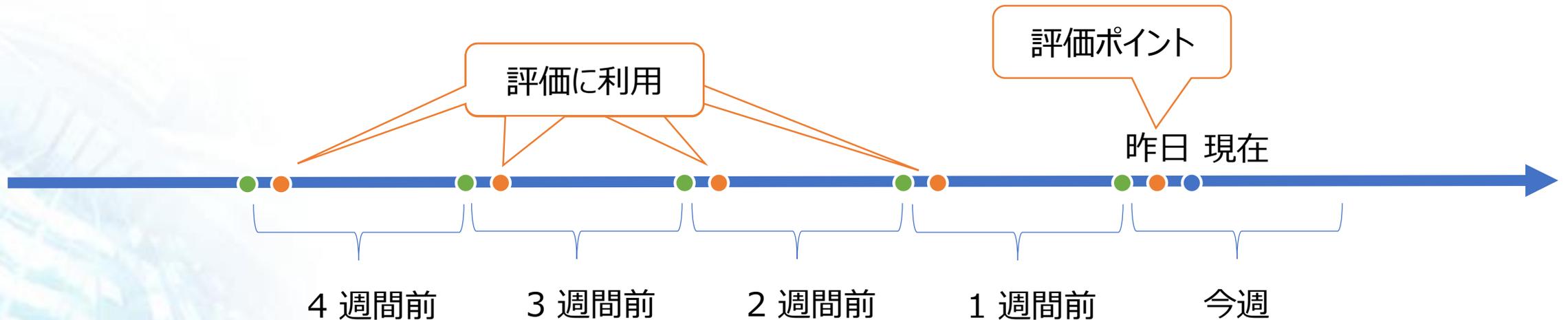
パラメータ	説明
<code>/host/key</code>	ホストおよびアイテム
<code>eval period</code>	STL 分解に利用するデータの期間
<code>detection period</code>	アノマリー（異常値）の検知対象期間
<code>season</code>	周期の期間
<code>deviations</code>	異常値とする偏差値（デフォルト：3）
<code>devalg</code>	偏差アルゴリズム（デフォルト：mad）
<code>s_window</code>	loess ウィンドウのスパン （デフォルト：10 * 評価期間中のエントリ数 + 1）

アノマリー検知の設定例

- `trendstl(/host/key,100h:now/h,10h,2h,2.1,"mad")`
 - 過去 100 時間のトレンドデータを元に 2 時間を周期として直近 10 時間の異常率を返す
ただし評価期間中の各残差の MAD (中央絶対偏差) を計算し偏差値が 2.1 に達した場合、その値を異常とみなす
- `trendstl(/host/key,1M:now/M-1y,1d,2h,, "stddevsamp")`
 - 1 年前を起点に過去 1 ヶ月のトレンドデータを元に 2 時間を周期として直近 1 日の異常率を返す
ただし評価期間中の各残差のサンプル標準偏差を計算し偏差値が 3 に達した場合、異常とみなす

ベースライン監視

- 過去のデータの指定期間の繰り返しから得られる値を監視に利用
 - トレンドデータから計算
- 期間: 1日、周期: 4週間の場合
 - 現在が火曜日の場合、4週間の月曜日のデータを元に前日の月曜日を評価



ベースライン監視のトリガー関数

- `baselinedev(/host/key,data period:time shift,season_unit,num_seasons)`
 - 周期の中の指定した期間の標準偏差を返す
- `baselinewma(/host/key,data period:time shift,season_unit,num_seasons)`
 - 周期の中の指定した期間の平均値を返す

パラメータ	説明
<code>/host/key</code>	ホストおよびアイテム
<code>data period</code>	データ期間
<code>season_unit</code>	周期単位
<code>num_seasons</code>	周期数

ベースライン監視の設定例

- `baselinedev(/host/key,1d:now/d,"M",6)`
 - 前日と過去 6 ヶ月間の同じ日の標準偏差数（母数）を計算
- `baselinewma(/host/key,1h:now/h,"d",3)`
 - 昨日までの3日間のうち、最後の1時間を基準として平均値を計算

アノマリー検知とベースライン監視の評価方法



トリガー関数の計算結果を基準としてどう評価するかは
ユーザ次第

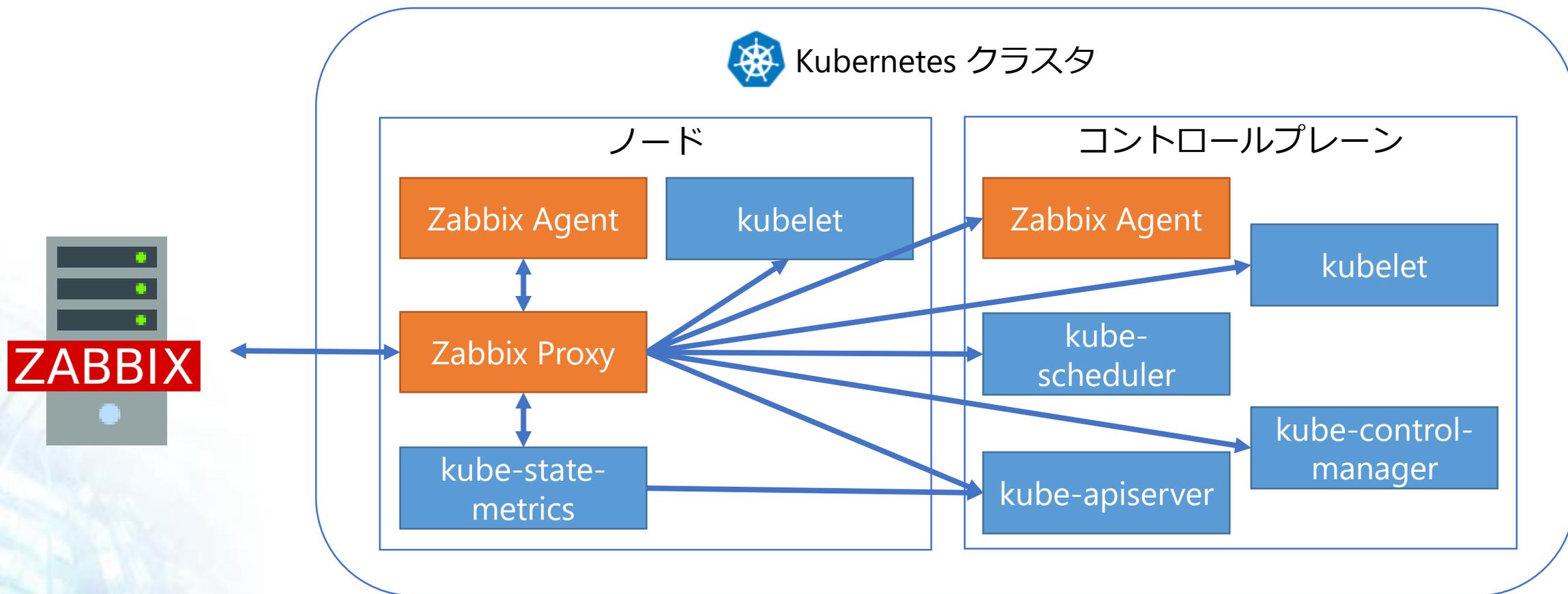
Kubernetes 監視

Kubernetes 監視

- ノードと Pod の自動検出と監視
- ノードや Pod に関する情報をエージェントレスで収集
- ノードホストから詳細な情報を取得可能

- 以下のテンプレートが追加
 - Kubernetes nodes by HTTP
 - Kubernetes cluster state by HTTP
 - Kubernetes API server by HTTP
 - Kubernetes Controller manager by HTTP
 - Kubernetes Scheduler by HTTP
 - Kubernetes kubelet by HTTP

Kubernetes 監視の概要図



Kubernetes 監視導入ツール

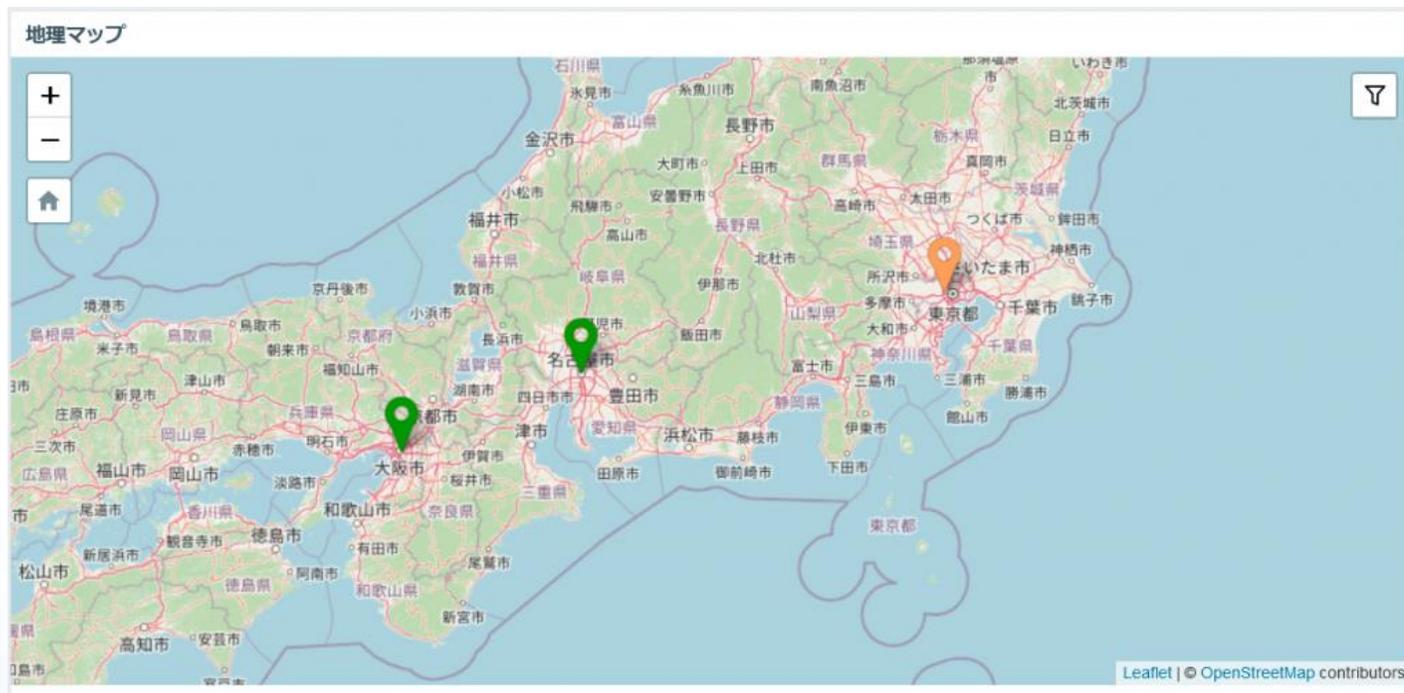
- Kubernetes 監視をフルに活用するためには Zabbix proxy と Zabbix agent を Kubernetes クラスタに導入する必要あり
- そのためのツールである Zabbix Helm Chart を提供
 - <https://git.zabbix.com/projects/ZT/repos/kubernetes-helm/browse?at=refs%2Fheads%2Frelease%2F6.0>
 - 要件
 - Kubernetes cluster 1.18 以上
 - Helm 3.0 以上
 - Zabbix server 6.0 以上
 - kube-state-metrics 2.13.2 以上



新規ウィジェット

地理マップウィジェット

- 地図に監視対象の場所を表示
- 発生している障害によって色分け
- インベントリで緯度、経度の設定が必要



アイテムの値ウィジェット

- 単一のアイテムの現在値を表示
- 表示スタイル、アイコンも変更可能

2022-02-09 10:38:36

Up (1) ↑↓

Server4: Zabbix agent ping

Zabbix server: Available memory

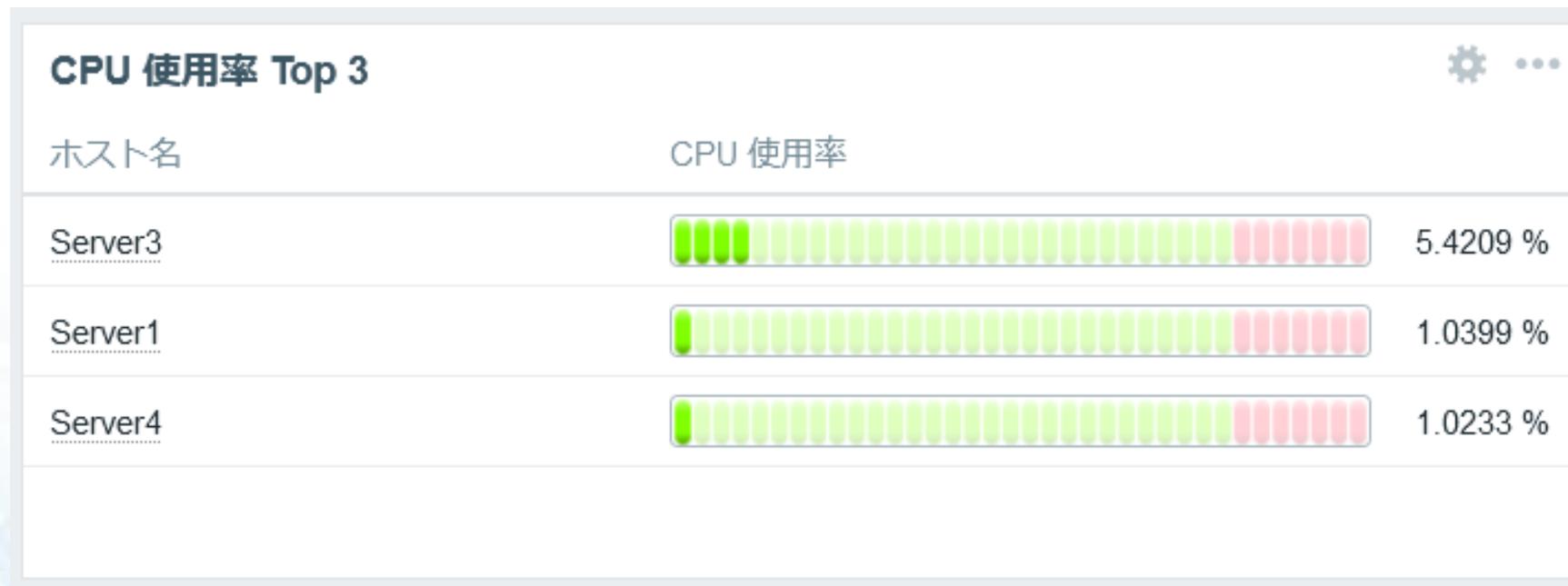
2022-03-08 11:19:19

789.16 ↓ MB

Available memory

上位ホストウィジェット

- ユーザがカスタマイズ可能なデータ概要テーブルを表示
- 特定アイテムの TOP N やバーゲージ表示が可能



その他の新機能

データベースのバージョンチェック機能

- Zabbix server/proxy 起動時 DB のバージョンをチェック
- 対応バージョン外なら起動しない

zabbix_server.log

Unable to start Zabbix server due to unsupported PostgreSQL database server version (10.17)

Must be at least (13.0)

Use of supported database version is highly recommended.

Override by setting AllowUnsupportedDBVersions=1 in Zabbix server configuration file at your own risk.

新規追加されたアイテムキー

- agent.hostmetadata : エージェントのホストメタデータ
- kernel.openfiles : 現在開いているファイルディスクリプタ数
- net.tcp.socket.count[] : パラメータに一致する TCP ソケット数
- net.udp.socket.count[] : パラメータに一致する UDP ソケット数
- vfs.dir.get[] : JSON 形式のディレクトリ情報
- vfs.file.get[] : JSON 形式のファイル情報
- vfs.file.owner[] : ファイルの所有者情報
- vfs.file.permissions[] : ファイルのパーミッション

 詳細は [Appendix II](#) へ

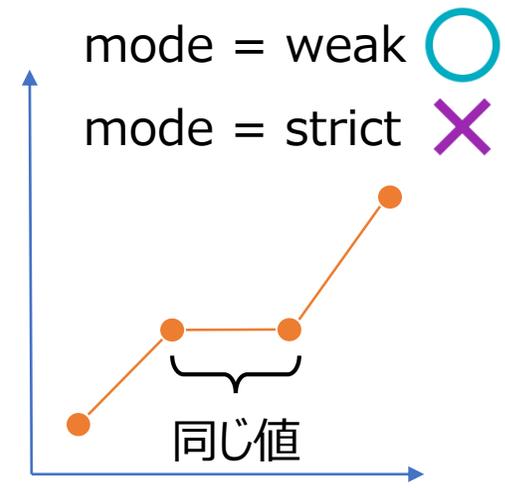
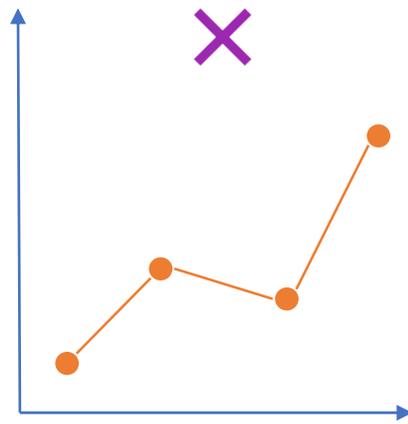
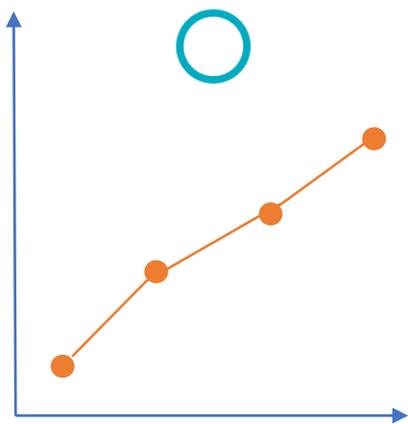
更新されたアイテムキー

- `vfs.file.cksum[]` : 第 2 パラメータ `<mode>` が追加
(`crc32` (デフォルト), `md5`, `sha256`)
- `vfs.file.size[]` : 第 2 パラメータ `<mode>` が追加
(`bytes` (デフォルト), `lines`)
- `vfs.fs.discovery` : Windows の場合、ボリューム名を
{#FSLABEL} マクロで取得
- `vfs.fs.get` : Windows の場合、ボリューム名を
{#FSLABEL} マクロで取得

単調変化の検知

- `monoinc()` 関数 : 任意期間の単調増加を検知
- `monodec()` 関数 : 任意期間の単調減少を検知

i `monoinc()` の場合



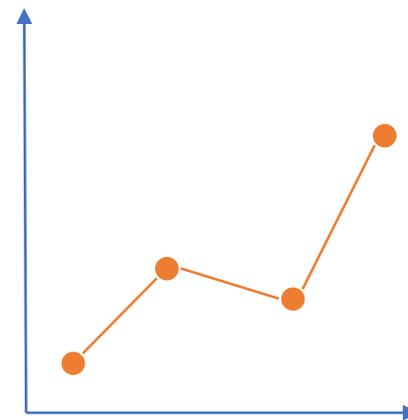
変化回数の検知

- `changecount()` 関数 : 任意期間に隣接する値の間で変化があった回数を返す

i

mode

- `all` : 増減問わずカウント ⇒ 3
- `inc` : 増加のみカウント ⇒ 2
- `dec` : 減少のみカウント ⇒ 1



Github メディアタイプの追加

- Webhook で Github の issue を登録可能



Zabbix ユーザのパスワードポリシー設定

- 最小パスワード長
- 必須項目
 - 大文字小文字
 - 数字
 - 記号
- 推測が容易な文字列の禁止
 - よくあるパスワード
 - ユーザ名を含む

The screenshot shows the Zabbix authentication settings page. The title is "認証" (Authentication). There are four tabs: "認証", "HTTP認証の設定", "LDAP認証の設定", and "SAML認証の設定". The "認証" tab is selected. Under "デフォルトの認証" (Default authentication), there are two buttons: "Zabbixデータベース内のユーザー情報" (selected) and "LDAP". Under "パスワードポリシー" (Password policy), there is a "最小パスワード長" (Minimum password length) field set to "8". Below that, there are three checkboxes for "パスワード必須項目" (Password required items): "英字の大文字と小文字" (English uppercase and lowercase), "数字" (Numbers), and "特殊記号" (Special characters). All three are currently unchecked. At the bottom, there is a checkbox for "推測されやすいパスワードの回避" (Avoidance of easily guessable passwords), which is checked. A blue "更新" (Update) button is at the bottom right.

Zabbix get、Zabbix sender のタイムアウト

zabbix_get、zabbix_sender コマンドで
タイムアウトを指定するパラメータをサポート

```
-t <seconds>  
--timeout <seconds>
```

- Zabbix get : 1 ~ 30 秒 (デフォルト: 30 秒)
- Zabbix sender : 1 ~ 300 秒 (デフォルト: 60 秒)

アップグレード時の注意点

アップグレードの前に

- 公式ドキュメントの新機能およびアップグレードに関するページを確認
 - 1. Introduction – 5 What's new in Zabbix 6.0.0 ~ 6.0.x
 - 4. Installation – 7 Upgrade procedure
 - 4. Installation – 8 Known issues
 - 4. Installation – 10 Upgrade notes for 6.0.0 ~ 6.0.x

Zabbix 6.0 アップグレード時の注意点 1

- RHEL で Zabbix server のパッケージは 8 系のみ提供
 - Zabbix proxy は RHEL 7 のパッケージも提供
 - Zabbix agent は RHEL 5、agent 2 は RHEL 6 以上
- Zabbix server と Zabbix proxy は要同一メジャーバージョン
 - Zabbix agent は後方互換性あり
- DB のサポートバージョンに注意
 - サポート外だとデフォルトでは起動しない
 - RHEL 8 の公式リポジトリの PostgreSQL、MySQL を利用するには RHEL 8.4 以上が必要

Zabbix 6.0 アップグレード時の注意点 2

- history テーブルへの主キー追加は別途スクリプトを実行
 - スクリプトは zabbix-sql-scripts パッケージに同梱
 - テーブルの変換には history テーブル分の空き容量が必要
- PCRE が PCRE2 に変更
 - 正規表現の記述方法に一部違い
- 監査ログの履歴は移行不可
 - 6.0 からの監査ログのスキーマの変更による

Zabbix 6.0 アップグレード時の注意点 3

- アイテム名での-positionalマクロ (\$1, \$2 ...) のサポート削除
 - アップグレードで自動変換はされない
 - Zabbix のサポート契約があるとマクロを変換してくれる
コマンドラインユーティリティが利用可能

参考

- Zabbix 6.0 manual
 - <https://www.zabbix.com/documentation/6.0/en/manual>
- What's new in Zabbix 6.0.0
 - <https://www.zabbix.com/documentation/6.0/manual/introduction/whatsnew600>
- Zabbix roadmap
 - <https://www.zabbix.com/jp/roadmap>

SRA OSS Tech Blog

- 弊社 Tech Blog では Zabbix を始め、様々な OSS の技術情報を掲載
- Zabbix 6.0 に関する技術情報も随時掲載予定
- SRA OSS Tech Blog
 - <https://www.sraoss.co.jp/tech-blog/>



オープンソースとともに



SRA OSS, INC.

Appendix

新規追加されたアイテムキー

agent.hostmetadata

- エージェントの**ホストメタデータ**を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : 文字列
 - パラメータ : なし



ホストメタデータ

1. zabbix_agentd.conf の HostMetadata に設定した文字列
2. HostMetadataItem に設定したアイテムキーの取得結果
3. 両方設定していなければ空文字列

kernel.openfiles

- 現在開いている**ファイルディスクリプタ数**を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : 数値 (整数)
 - パラメータ : なし

net.tcp.socket.count[]

- パラメータに一致する **TCP ソケット数**を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : 数値 (整数)
 - パラメータ : <laddr> ローカルの IPv4/v6 アドレス (CIDR 表記可)
<lport> ローカルのポート番号もしくはサービス名
<raddr> リモートの IPv4/v6 アドレス (CIDR 表記可)
<rport> リモートのポート番号もしくはサービス名
<state> 接続状態 (established, syn_sent, syn_recv, fin_wait1, fin_wait2, time_wait, close, close_wait, last_ack, listen, closing)

net.udp.socket.count[]

- パラメータに一致する **UDP ソケット数**を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : 数値 (整数)
 - パラメータ : <laddr> ローカルの IPv4/v6 アドレス (CIDR 表記可)
<lport> ローカルのポート番号もしくはサービス名
<raddr> リモートの IPv4/v6 アドレス (CIDR 表記可)
<rport> リモートのポート番号もしくはサービス名
<state> 接続状態 (established, unconn)

vfs.dir.get[]

- ディレクトリの情報を JSON 形式で返す

- タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
- データ型 : テキスト
- パラメータ : dir ディレクトリのパス
 - <regex_incl> 含めるエンティティの正規表現
 - <regex_excl> 除外するエンティティの正規表現
 - <types_incl> リストする種類 (file, dir, sym, sock, bdev, cdev, fifo, dev, all)
 - <types_excl> リストしない種類
 - <max_depth> 探索するサブディレクトリの深さ
 - <min_size> リストする最小 byte 数
 - <max_size> リストする最大 byte 数
 - <min_age> リストする最小タイムスタンプ
 - <max_age> リストする最大タイムスタンプ
 - <regex_excl_dir> 除外するディレクトリの正規表現



```
[
  {
    "basename":"zabbix_agentd.d",
    "pathname":"/etc/zabbix/zabbix_agentd.d",
    "dirname":"/etc/zabbix",
    "type":"dir",
    "user":"root",
    "group":"root",
    "permissions":"0755",
    "uid":0,
    "gid":0,
    "size":4096,
    "time": {
      "access":"2021-11-10T04:16:58+0900",
      "modify":"2021-11-10T00:50:35+0900",
      "change":"2021-11-10T04:16:56+0900"
    },
    "timestamp": {
      "access":1636485418,
      "modify":1636473035,
      "change":1636485416
    }
  },
],
```

vfs.file.get[]

- ファイルの情報を JSON 形式で返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : テキスト
 - パラメータ : file ファイルのフルパス



```
{
  "type": "file",
  "user": "root",
  "group": "zabbix",
  "permissions": "0600",
  "uid": 0,
  "gid": 975,
  "size": 25903,
  "time": {
    "access": "2021-11-09T11:22:32+0900",
    "modify": "2021-11-09T11:22:28+0900",
    "change": "2021-11-09T11:22:28+0900"
  },
  "timestamp": {
    "access": 1636424552,
    "modify": 1636424548,
    "change": 1636424548
  }
}
```

vfs.file.owner[]

- ファイルの所有者情報を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント (アクティブ)
 - データ型 : 文字列
 - パラメータ : file ファイルのフルパス
 <ownertype> 所有者の種類 (user, group)
 <resulttype> 所有者の表示形式 (name, id)

vfs.file.permissions[]

- ファイルのパーミッション情報（8進数4桁）を返す
 - タイプ : Zabbix エージェント
Zabbix エージェント（アクティブ）
 - データ型 : 文字列
 - パラメータ : file ファイルのフルパス



Windows の Zabbix agent ではサポートされない