

Zabbix Conference Japan 2017

クラウド/DC事業者の中での 普通のZabbixの使い方

株式会社IDCフロンティア

運用本部 運用部 システムオペレーショングループ

千代田 晴良, 渡邊 一雄

2017年11月17日



ヤフー株式会社 100%資本の子会社

主にデータセンターやクラウドなどの

ITインフラの提供を行っております

【データセンター】

首都圏、関西、東北、
九州のデータセンター群



【クラウド IaaS】

東日本リージョン／西日本リージョン





Monitoring



Mail delivery



Push notice

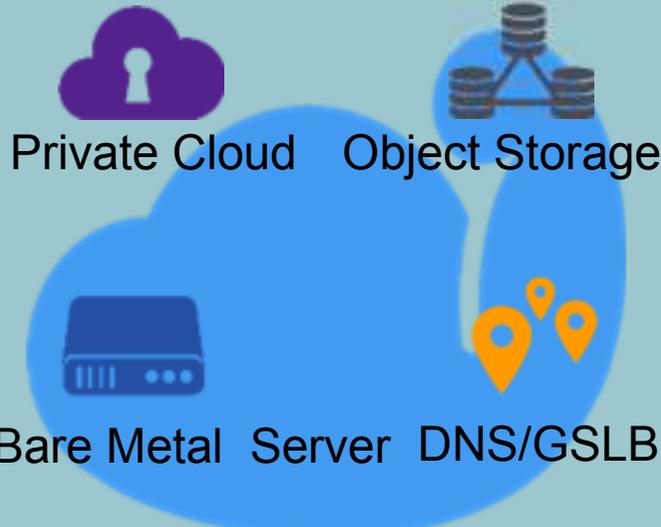


Crash analysis

Application Service



BIG Data



Cloud

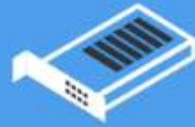


Data Center



Network

IDCFクラウド 3つの"速さ"



Performance

わずか20秒でサーバー作成！オールフラッシュディスクを採用した超高速・パワフルなクラウドです。



Network

IDCFフロンティアのバックボーンネットワークは総容量1,110Gbpsと大容量！急激なアクセス増や大量配信にも最適。



Usability

シンプルなUIとAPIで、素早く簡単にシステム構築。オンラインサポートや請求書発行など、ビジネス利用にも安心。

IDCFクラウド 3つの"安心"



セキュリティ万全

- DDoS対策標準装備しています。
- ファイアウォールやロードバランサーなど無料機能も充実
- IDCFクラウド セキュリティホワイトペーパー



強力なファシリティ

- 世界最新鋭の国内データセンターで運営
- 低災害リスクエリアに立地
- 東西リージョン・ゾーンの利用でBCP対応も万全



充実のサポート

- 24時間365日サポート無料
- 専門スタッフが常駐し、テクニカルなご質問も手厚くサポート
- 大規模システム運用に最適なアドバンスドオペレーションもご用意しております

千代田 晴良



Haruyoshi Chiyoda



法人向けインターネット接続サービスのサポート
チーム立ち上げ、DC運用インフラ整備、営業技
術、システムの運用管理、等を経験。

最近は、CDN、DNSといったコンピューティン
グクラウドサービスの技術チームも担当。

ZABBIX-JP Study#5 Zabbix2.0rc1 SNMP Traps
(May,2012)



渡邊 一雄



Kazuo Watanabe



インフラエンジニア

Cacti : 1年, Zabbix : 5年, Nagios:10年,

HP SiS,OperationsManager : 7年

監視システム構築運用、監視きっかけでの自動化
設計実装などの経験（10年）をコアに、現在も業
務領域を拡大中。現在はDNS系サービスも担当。

株式会社IDCフロンティア（ヤフーグループ）

日本Zabbixユーザ会

開発（インフラ）

サービス企画



CDN, DNS, GSLB, NTP
Monitoring Infra,
Bastion Server, Syslog...



運用管理基盤整備

ZABBIX
Nagios®

運用実装支援
（監視中心）

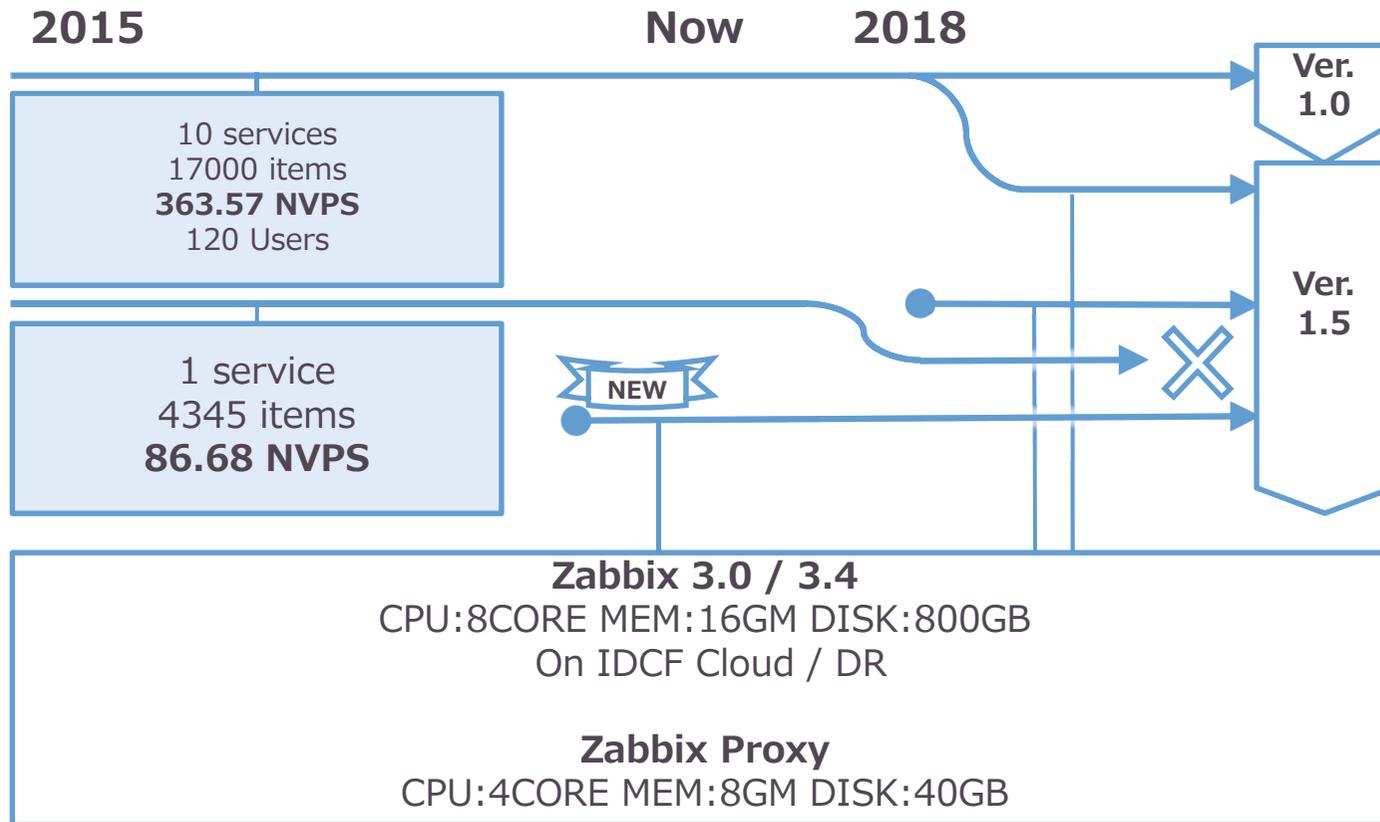
Zabbix管理者、監視実装支援者、システム管理者
3つの立場で本日はお話ししたいと思います

当社Zabbixの構成のご紹介

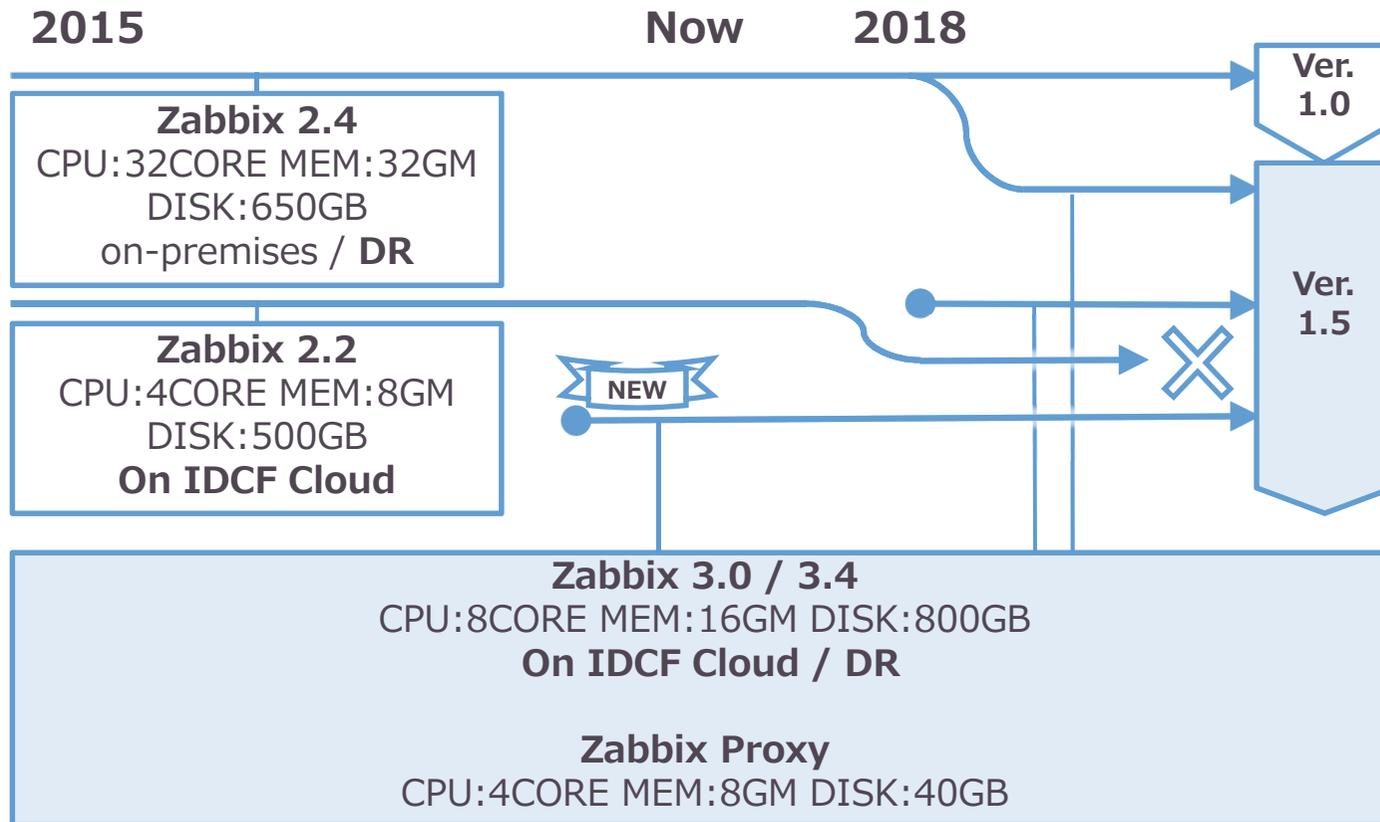
IDCFクラウドでZabbix運用してみませんか？



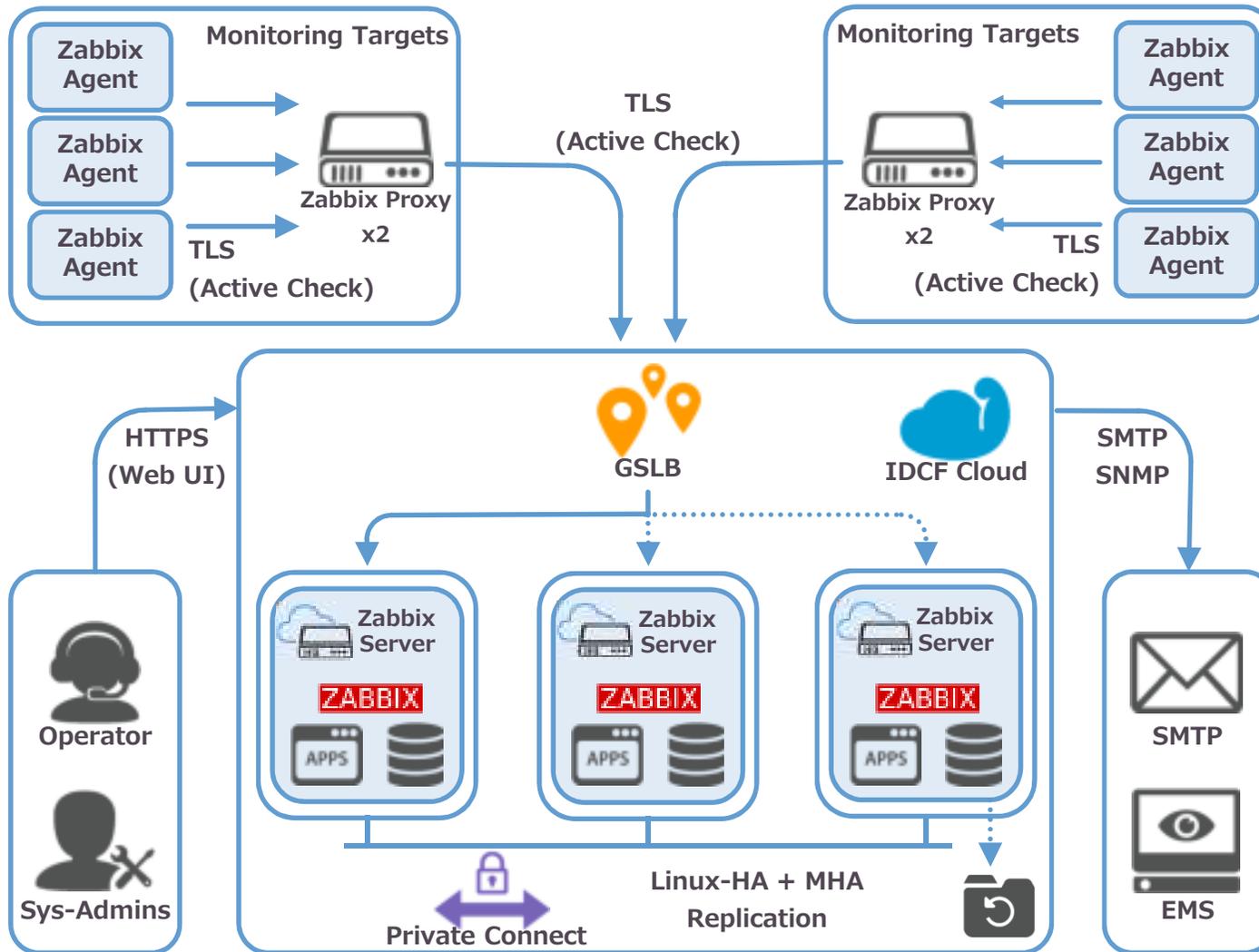
最近の Zabbix 運用状況



最近の Zabbix 運用状況



IDCFクラウド上に構築するZabbix DR構成



ダウンタイム 最大1分

通知遅延のみ影響 (履歴欠損なし)

詰め込みすぎない / 複数メジャーバージョン存在

- 監視対象を集約しすぎることによる障害影響規模の分散
- Zabbix Ver. UP > サービスライフサイクル
- LTS配慮 < より新しいバージョンでの恩恵享受

インフラ構成 (リージョン冗長、ディスク、メモリについて)

- 障害への配慮 / インフラライフサイクルへの配慮
- DBストレージ部分だけは拡張ディスクでいざとなれば拡張
- メモリは多めに確保

プライベートコネクでデータ共有、冗長コントロール

- MHA for MySQL : DB
- corosync + Pacemaker : HTTPD + Zabbix + grafana (IPは対象外)

GSLBでZabbixが活着ているリージョンへ接続

プロキシを利用して集約・ダウン時のデータバッファとして確保

- 管理経路の単純化 > Remote Command 機能

暗号化必須

- 社外クラウド、レンサバが監視対象でもセキュリティ担保
- Active Check のみのため TLS / PSK のみ利用

Zabbixを システム運用管理の 自社フレームワークにしている話



Zabbix Severity	サービス状態 (目安)	管理者通知 (SMTP)	組織共有	運用連携
Disaster	大規模障害	(x)	(x)	(x)
High	障害	(x)	(x)	(x)
Average	ワークアラウンド対応	x	-	(x)
Warning	不具合兆候	x	-	-
Information	調査時参考閾値	-	-	-
Recovery	復旧	x	(x)	(x)

Severity と 通知範囲 を 事前定義

参考/出典 [Qiita] Zabbix Advent Calendar 2015
Zabbix trigger severities の運用設計事例 (個人的ベストプラクティス)
<https://qiita.com/halchiyo/items/165c2aa585a865cb584e>

分類	種類	テンプレート
Network機器	6種	15件
		Cisco, A10, Alaxala, Juniper など (検証中)
OS	2種	11件
	RHEL系:	CentOS 6 / 7
	Windows系:	Windows Server 2008 / 2012
Middleware	19種	32件
		Docker, MySQL, ExtraDB, MongoDB, PostgreSQL, Redis, RabbitMQ, ATS, Sensu, Nagios, Unbound, PowerDNS, nginx, Apache Traffic Server ...

社内最適化テンプレート 約60件
(アイテム : 800)
サービス毎に管理者に配布

system.cpu.util[,user]	60
vm.memory.size[available]	60
vm.memory.size[pavailable]	60
proc.num[,,,,usr/libexec/postfix/master]	60
system.users.num	60
system.swap.size[,pfree]	60
system.localtime	60
system.swap.size[,free]	60
system.cpu.load[percpu,avg1]	60
system.cpu.util[,interrupt]	60
system.cpu.load[percpu,avg15]	60
custom.tcp.conn.stat[LAST_ACK]	60
custom.tcp.conn.stat[SYN_RCVD]	60
custom.tcp.conn.stat[FIN_WAIT1]	60

- 複数サービスの最大公約数
- 事後追跡を想定
- 原則1分間隔
- 日々改善

密度の濃い監視
メトリックは細かくマメに収集

TCP connection num LAST_ACK	
TCP connection num SYN_RCVD	
TCP connection num SYN_SENT	
TCP connection num TIME_WAIT	
Kernel max open files	トリガー (1)
Kernel max process	トリガー (1)
Service SMTP	
Service SSH	
Proc postfix	
OS number of running processes	
Proc crond	
Proc rsyslogd	
OS number of processes	
OS host boot time	
CPU Interrupts per second	
CPU LOAD1	
CPU LOAD5	
CPU LOAD15	

深刻度	名前
警告	chronyc offset threshold over
警告	CPU idle(%) is too low
警告	CPU load is too high
情報	File check sum of /etc/passwd was changed
情報	Kernel maxfiles is too low
情報	Kernel maxproc is too low
警告	Memory available(%) is too low
警告	OS rpm list changed
警告	Proc rsyslog is down

- 出しすぎない
 - アラートに麻痺しない
- 初期Severity設定は、管理者限定通知だけ
 - カジュアルに使い始めたい
- アラートの障害影響をふまえ、管理者がSeverityを変更 (通知範囲調整)

運用展開無しで使い始められる
必要に応じすぐ運用連携できる

```
#  
D 1 ntpd ntpd -u ntp:ntp -p /var/run/ntpd.pid  
H 1 snmpd /usr/sbin/snmpd -LS0-6d -c /etc/snmp/snmpd.conf  
A 1 postfix master /usr/libexec/postfix/master
```

Name ↓↑	Items
Process warning discovery	Item prototypes (1)
Process not classified discovery	Item prototypes (1)
Process information discovery	Item prototypes (1)

Severity	Name ↓↑
Warning	Proc num {#ITEMNAME} less than {#LTHRE}

標準LLD利用

- ディスク
- インタフェース

カスタムLLD利用

- プロセス監視
- エージェント側で定義
 - Severity
 - プロセスと起動数
 - 表示名

典型的な監視なら Web UI 不要

**管理者が使いたくなる
使えば運用の質が揃う**

Severity と 通知範囲 を 事前定義

- 組織連携無しで使い始められる
- 必要に応じてすぐ組織連携できる

管理者目線の気配りと充実した品目

- 密度の濃い監視で事後調査にも有益
- トリガーは出しすぎない

Zabbix知らなくても使い始められる

- 典型的な監視なら Web UI 不要

サーバ内作業

- エージェントインストール、要件SW追加、Proxy IPの指定
- プロセス監視の定義

Zabbix操作

- テンプレートの調整（主にTriggerとSeverityチューニングで完結）
- ホストの登録とテンプレートの割り当て（API自動化事例もあり）

サービスごとに数件程度発生する相談

- スクリプト監視でのスクリプトの書き方とか動かし方
- ログ監視のチューニング & 新規作成
- 存在しない種類の監視対象のテンプレートの作成相談（大体共同作成）

**慣れは必要だが負担は大きくない
（不満が出たことはない）**

Before

- 監視経路を1台ずつ確認
- 監視対象ホストを登録するたびに通知先確認
- 全監視項目の閾値とサーバ構成を確認してテンプレート作成
- 何かあるごとに支援相談
- スピード,実装能力の上限が監視支援者依存

After

- Zabbix使い方簡単に説明
- **Proxyがないセグメントだけ経路調整**
- あとは管理者が調整
 - Severity (通知先)
 - サービス
- 不明点とか個別実装だけ相談対応
- 管理者の知恵と工夫を取り込めるようになった

標準的な実装工数 5日 → 2日

テンプレートと通知条件のパターン化追及

- 監視設定の質と量の底上げ
- 管理者、監視実装支援者双方の負担軽減

管理者への設定権限解放

- ボトルネック緩和
- 視点・発想の多様性
- 管理者の専門性活用

(監視対象のノウハウ・自動化等)

まとめ

IaaS上でのZabbix DR構築事例

弊社サービスを利用し、東西冗長構成で構築した事例を紹介
是非お試しください

『システム運用のフレームワークとして』 Zabbixを利用

テンプレートと通知条件のパターン化を迫及したら好循環した話

- 監視設定の質と量の底上げ
- 管理者、監視実装支援者、双方の負担軽減
- 管理者の専門性活用
- 多様性強化

これから

『サービスレベルの判断ツール』に Zabbixを利用できないか？

これまで
組織とシステムから決める監視

密度の濃い監視
フレームに沿った連携
早期検知
早期着手
原因特定手段拡大
監視実装負担軽減
多様性強化

↓
サービスレベルの維持と改善



さらに
サービスレベルから決める監視

サービスレベル
(サービスの障害規定)



エラーバジェット



エラー率 × 継続時間
(複合判定の監視実装)



サービスとして障害判断迅速化

『アラート数削減』と『アラート発生時の判断とアクションの迅速化』に期待
実証検証の為、設計検討開始

管理者側の対応円滑化

手順書のRPM化、Ansible PlayBook の配布、 etc.

テンプレートの充実

LLDのさらなる活用

現行テンプレートに対する予兆閾値の追加

資産管理業務も楽にしたい…

履歴データ活用もしたい/できるなら管理対象は少ないほうがよい

そもそもZabbixがベストなのかも含め検討していきたい

今後運用に載せていきたい

コンテナ技術の取り込み

自社サービスのさらなる活用

End

…またどこかでお会いしましょう