

データセンター 情報収集システムのポイント

DataCenter とソフトウェア開発ワークショップ

2014/11/21 石川県ハイテク交流センター

株式会社クルウィット 清原智和

データセンターの情報収集における特徴 (1/3)

- 対象数からみる特徴 -

- ・ 大量の機器から様々な情報を収集
 - ・ IT サーバ × 電源状態, FANの回転数, チップの温度 etc..
 - ・ ネットワークスイッチ × パケット転送量 etc..
 - ・ PDU, UPS × 消費電力量 etc..
 - ・ 空調 × 設定温度, 実温度, 湿度 etc..
 - ・ コントロールサービス (OpenStack etc..) × APIコール数 etc..
 - ・ ..

データセンターの情報収集における特徴 (2/3)

- 時間軸からみる特徴 -

- ・ 細かい粒度で収集
 - ・ 秒単位のインターバルで収集
 - ・ 障害の検知・分析
- ・ 長期にわたって収集
 - ・ 年単位の継続性で収集
 - ・ トレンド分析

データセンターの情報収集における特徴 (3/3)

- 変動性からみる特徴 -

- ・ 変化する対象から収集
 - ・ 機材の増減・コントロールサービスの入れ替え
 - ・ 収集対象数の変化
 - ・ 情報収集方法（プロトコル）の変化

データセンター 情報収集システムの要件

- ・ スケーラビリティ
 - ・ 収集する「情報量」を容易に拡張できること
 - ・ 規模透過性
 - ・ 収集する「対象」を容易に拡張できること
 - ・ 異種透過性

実装例

ポイントとなる システム構成要素

- ・ スケールさせやすい
- ・ 情報転送プロトコル
- ・ モジュール構成
- ・ インフラ基盤

ポイントとなる システム構成要素

- ・ スケールさせやすい

今回採用したものの

- ・ 情報転送プロトコル → MQTT

- ・ モジュール構成 → Docker

- ・ インフラ基盤 → Google Cloud Platform

MQTT

- ・ スケールさせやすい情報転送プロトコル
- ・ 多数の主体の間で短いメッセージを頻繁に送受信する用途に向けた軽量なプロトコル
(<http://e-words.jp/w/MQTT.html>)
- ・ Broker を介した pub/sub モデル
- ・ パフォーマンスに優れている

Docker

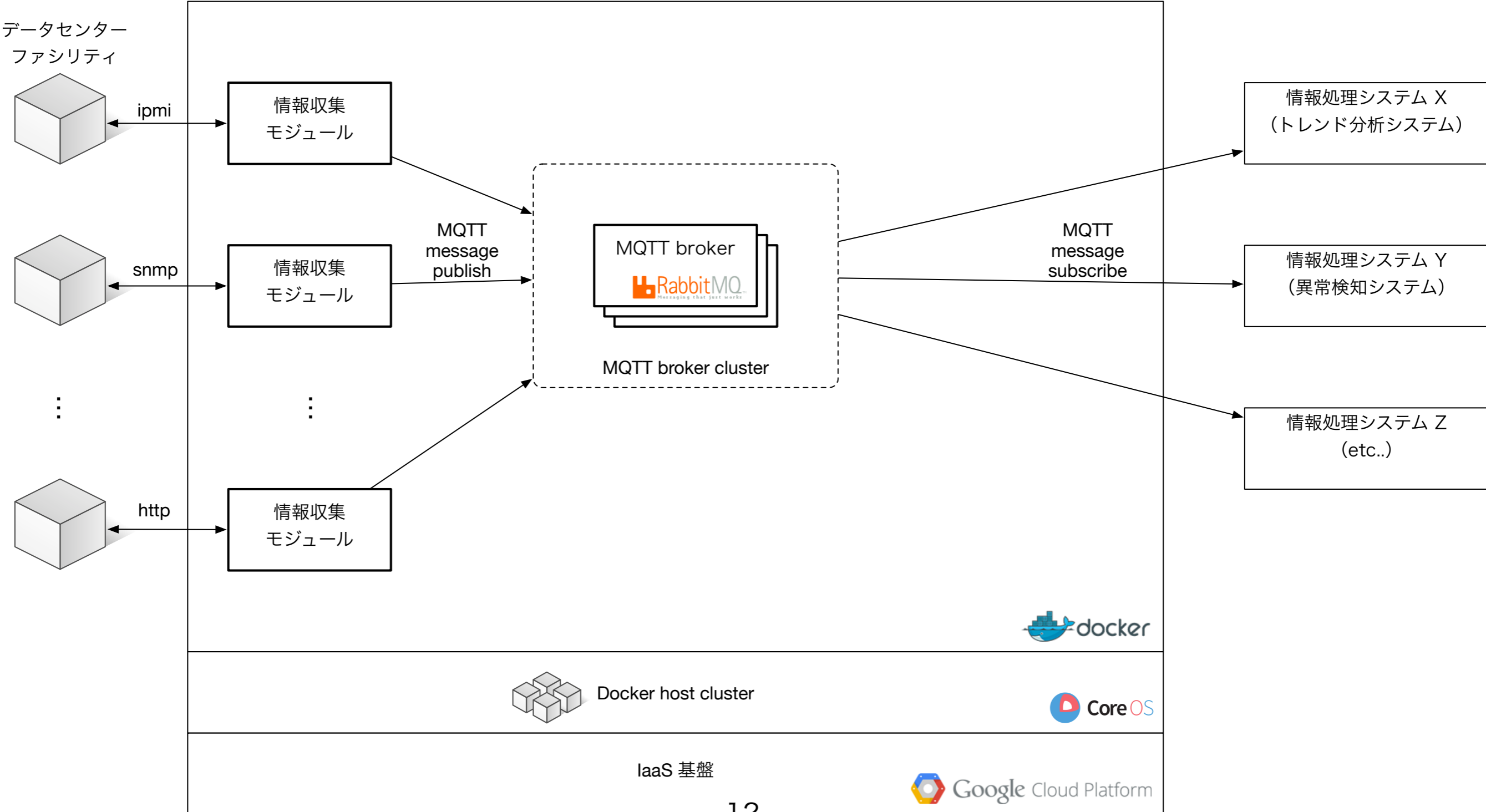
- ・ スケールさせやすいモジュール構成
- ・ アプリケーション実行環境をイメージとして作成・
配付・実行するためのフレームワーク
- ・ Linux コンテナ等の技術を組み合わせて実現
- ・ ポータビリティに優れている

Google Cloud Platform

- ・ スケールさせやすいインフラ基盤
 - ・ Google が提供する基盤プラットフォーム
 - ・ Google Compute Engine (IaaS)
- ・ スケーラビリティ（規模透過性）に優れている

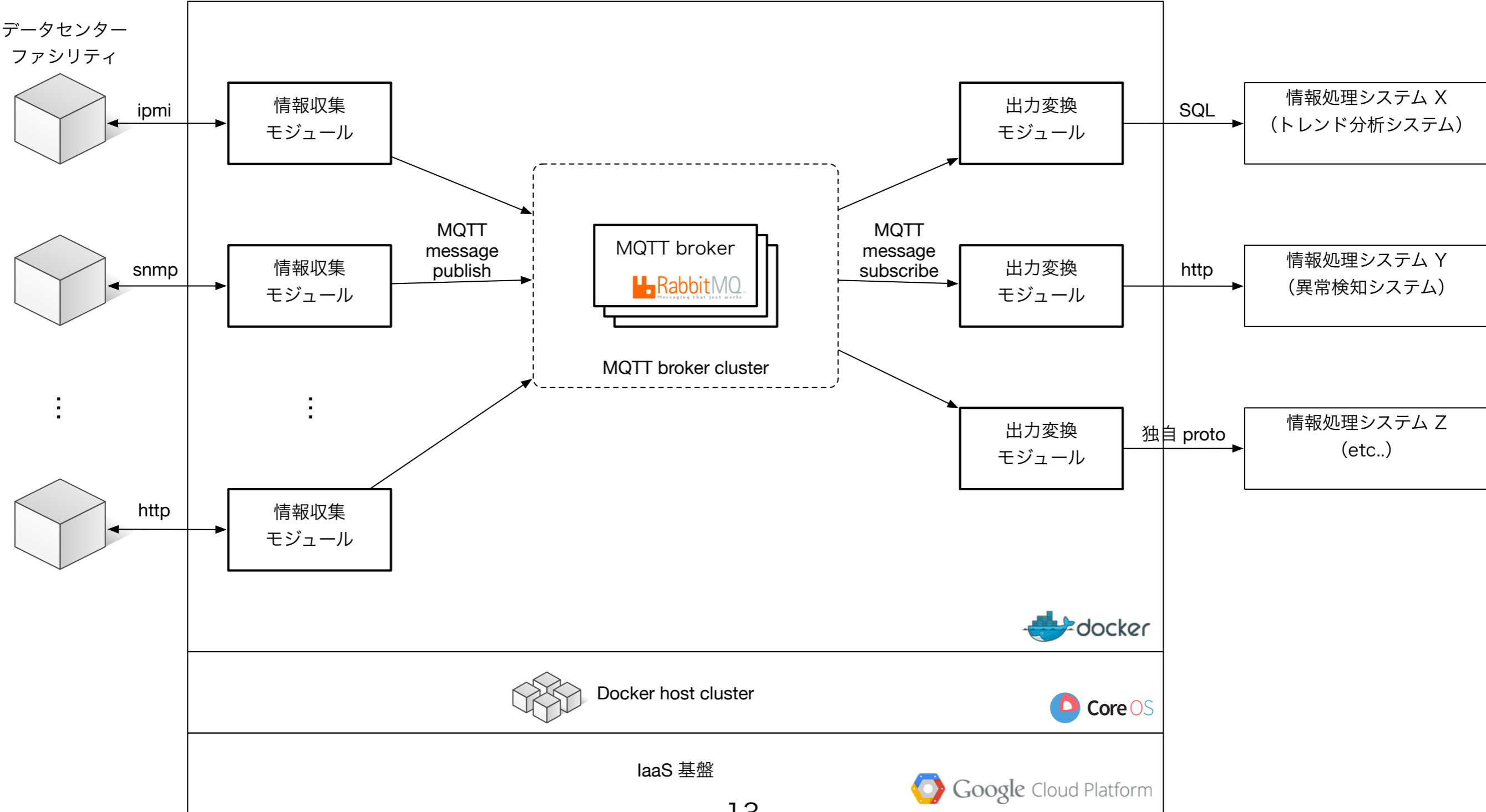
構成図 (1/2)

データセンター情報収集システム



構成図 (2/2)

データセンター情報収集システム



ファシリティ コントロールモジュール

ファシリティ

コントロールモジュール

- ・ ファシリティの状態変化を実現するモジュール
 - ・ 例: ipmi で電源を落とす
- ・ 情報収集モジュールと同居させることが望ましい
 - ・ クエリ量をコントロールしやすい
 - ・ 例: 多量の ipmi クエリは BMC を容易に落とす
- ・ 必要な機能の多くを流用できる

ファシリティコントロールモジュール 設計のポイント

- ・ 情報収集モジュールにファシリティコントロールの機能を追加しようとした場合、必要となるメモリリソースが増大する。
- ・ 大量のファシリティ向けにモジュールを多数動作させることを考えると、効率化の取り組みが大きく「効く」部分

ファシリティコントロールモジュール 設計のポイント

- ・ ひとつのモジュールに機能を同居させたい
- ・ 必要となるメモリリソースは少なくしたい

ファシリティコントロールモジュール 効率化に向けたアイデア

- ・ 前提

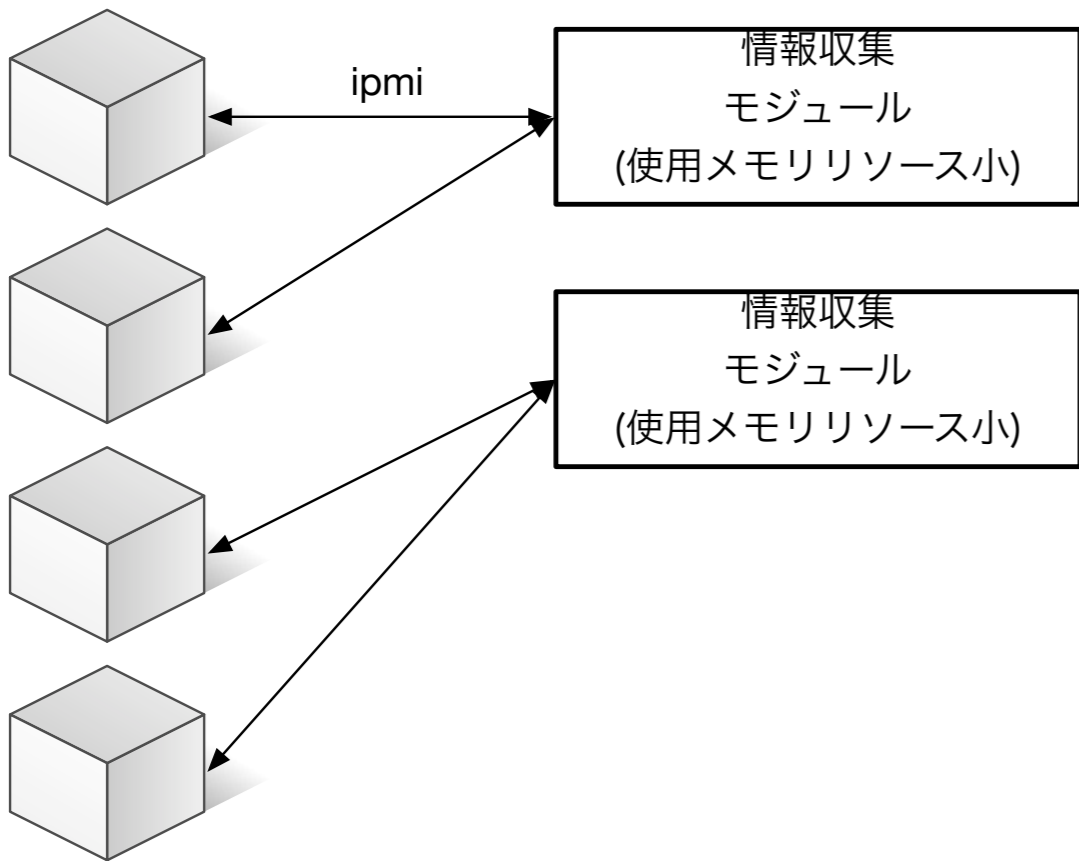
- ・ ファシリティを「操作」というオペレーションは情報を収集するオペレーションに比べ
- ・ 出現頻度が低いと予想される
- ・ 対象となるファシリティが限定的と予想される

ファシリティコントロールモジュール 効率化に向けたアイデア

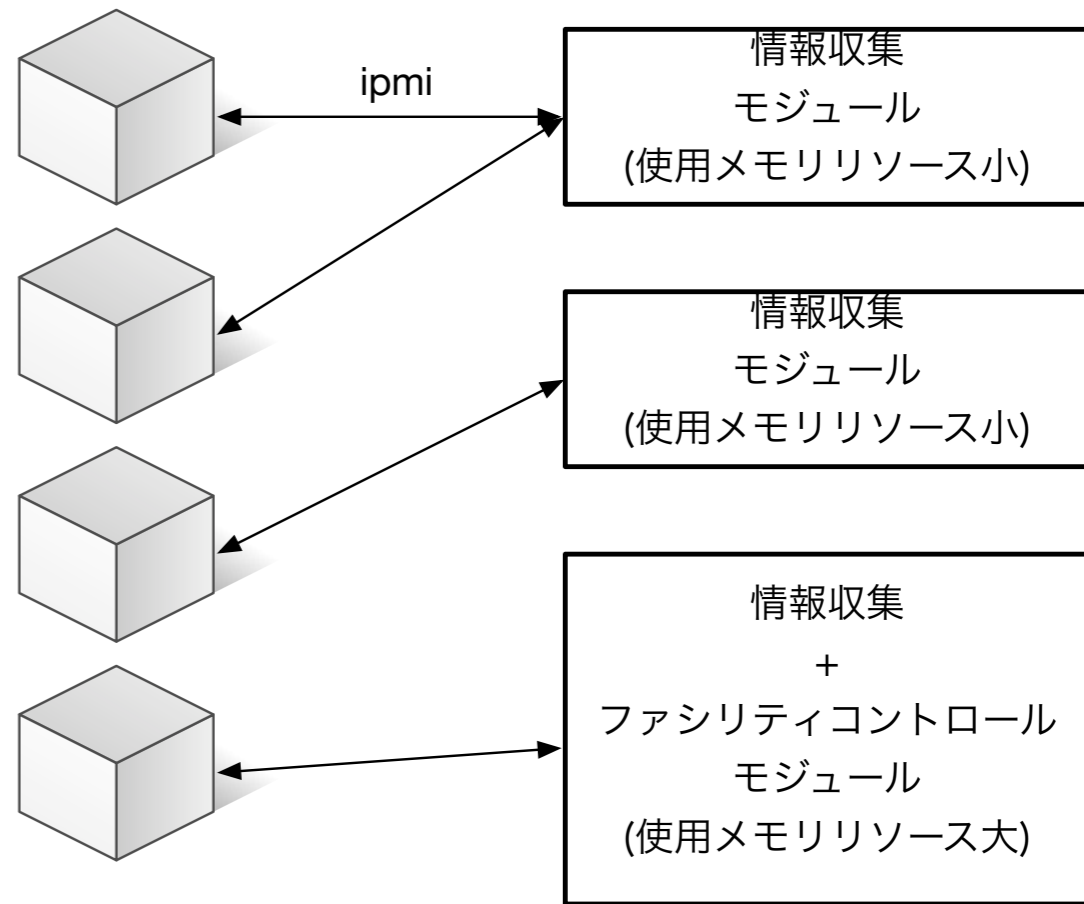
- ・ アプローチ
 - ・ 通常は「情報収集のみ」のモジュールを動作
 - ・ 必要に応じて「情報収集＋コントロール」のモジュールに動的に差し替える

イメージ図

データセンター
ファシリティ



データセンター
ファシリティ



コントロール要求が発生したタイミングで動的に生成し割り当てる

ファシリティコントロールモジュール 効率化に向けたアイデア

- ・ メリット
 - ・ メモリリソースの効率化
- ・ デメリット
 - ・ モジュール差し替えのオーバーヘッド
 - ・ 必要となるリソースの増大
 - ・ 差し替え速度

まとめ

- ・ データセンターの情報収集に関して
 - ・ 特徴・要件をまとめた
 - ・ スケーラビリティが肝要
 - ・ 要件を実現するための実装例を示した
 - ・ MQTT, Docker, Google Cloud Platform
- ・ データセンターファシリティのコントロールを見据えて
 - ・ 効率化につなげる取り組みを紹介した