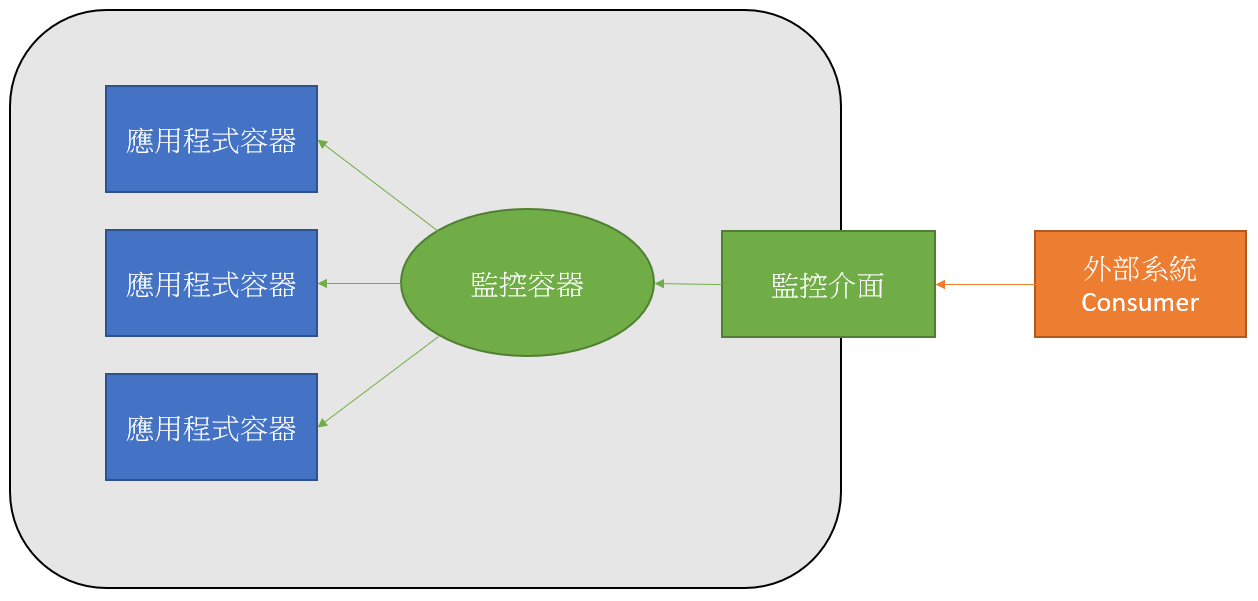
分散式系統中可能由多種異質性應用系統所組成, 某些子系統由供應商提供, 某些程式是開發團隊從無到有自行開發, 某些部分則是由現成開放原始碼軟體, 加上開發團隊客製化調整後編譯而成, 這種異質化的特性, 讓系統使用多種程式語言, 具有格式相異的日誌及監控紀錄以及不同的通用服務函式庫。這樣的特性讓系統管理人員難以從單一的位置收集到所有子系統運作的指標資訊(Metrics), 且個別子系統使用不同介面, 產生的指標資訊(Metrics)可能格式相異, 要進行視覺化分析來監控系統運作狀態, 以及系統出現異常時適時產生警報(Alerts)變得十分困難。

適配器(Adaptor)設計模式提供了解藕系統的解決方案, 使用一個模組化的適配器容器(Adaptor Comtainer) 來適應這種異質性的結構, 為不同的子系統提供的異質監控介面, 提供一個一致的使用介面。



適配器(Adaptor)設計模式, 可以讓系統更易於理解及維護, 新加入一個實作適配器的應用程式時, 可使用相同的介面來監控新系統, 且適配器容器為獨立的容器, 擁有專用的CPU及記憶體資源, 確保監控行為不會導致客戶端服務出現問題。

Prometheus 是一個監控聚合器(Monitoring Aggregator), 負責收集指標, 並將他們集中到一個時間序列(Time-Series)資料庫中, 在資料庫中, Prometheus 提供視覺化工具和查詢語言, 用於內部查詢所收集的指標, 並提供公開的metrics API, 讓Prometheus可以透過單一介面監控不同的程序。

對於無法輸出與Prometheus相容的應用程式容器, 只要增加一個導出器(Prometheus Exporter), 即可轉換導出與Prometheus相容的指標, 符合Prometheus預期的格式, 從而被Prometheus容器所監控。

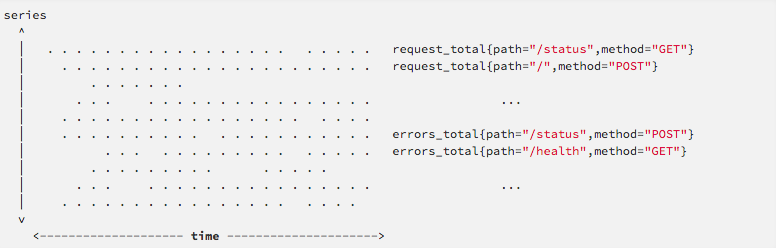
Prometheus 特性:

Time Series Database時間序列資料庫

監控Metrics : 可加上標籤, 監控多種屬性

Pull 監控模式(scraping)

支援黑箱(BlackBox)及白箱(WhiteBox)監控

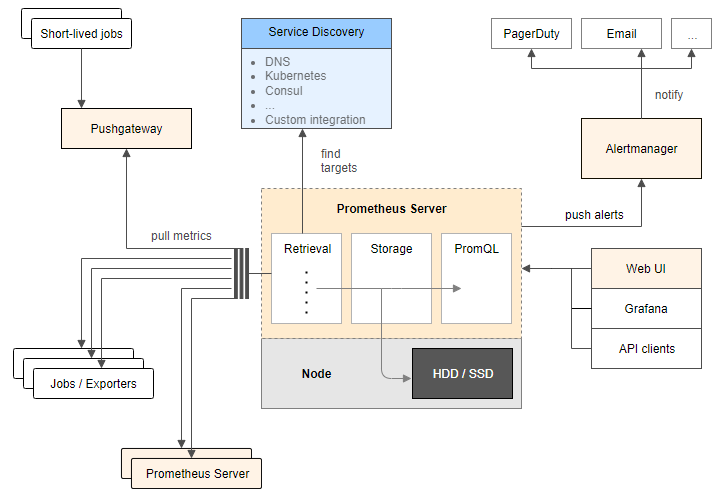


DevOps friendly, Cloud Native Computing Fundation成員專案

警示Alert 機制

龐大社群, 支援多種程式語言及各種導出器Exporters

效能好, 每秒可消費百萬級時間序列資訊, 可監控上千個目標Targets



**Prometheus Server**

Prometheus Server是Prometheus組件中的核心部分，負責實現對監控數據的獲取，儲存以及查詢

Prometheus Server可以通過靜態配置管理監控目標，也可以配合使用Service Discovery的方式動態管理監控目標，從這些監控目標中獲取數據

Prometheus Server本身就是一個時序數據庫，將采集到的監控數據按照時間序列的方式存儲在本地磁盤當中。最後Prometheus Server對外提供了自定義的PromQL語言，實現對數據的查詢以及分析

Prometheus Server內置的Express Browser UI，通過這個UI可以直接通過PromQL實現數據的查詢以及可視化。

**Exporters**

Exporter將監控數據采集的端點通過HTTP服務的形式暴露給Prometheus Server，Prometheus Server通過訪問該Exporter提供的Endpoint端點，即可獲取到需要采集的監控數據。

一般來說可以將Exporter分爲2類：

直接采集：這一類Exporter直接內置了對Prometheus監控的支持，比如cAdvisor，Kubernetes，Etcd，Gokit等，都直接內置了用于向Prometheus暴露監控數據的端點。

間接采集：間接采集，原有監控目標幷不直接支持Prometheus，因此我們需要通過Prometheus提供的Client Library編寫該監控目標的監控采集程序。例如： Mysql Exporter，JMX Exporter，Consul Exporter等。

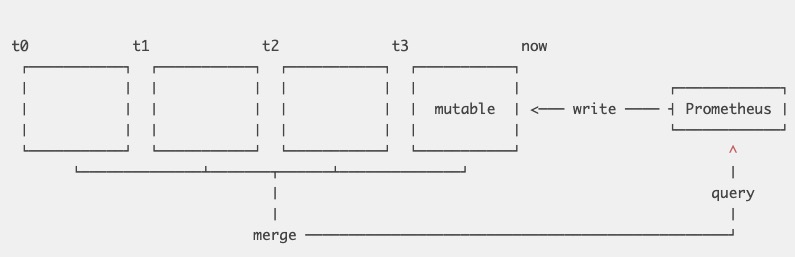
**AlertManager**

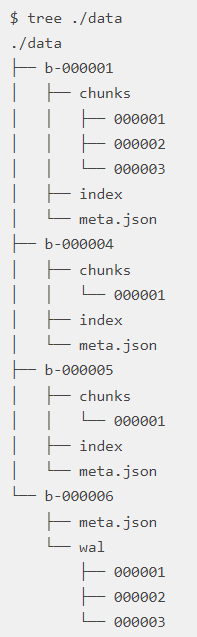
在Prometheus Server中支持基于PromQL創建告警規則，如果滿足PromQL定義的規則，則會産生一條告警，而告警的後續處理流程則由AlertManager進行管理。在AlertManager中我們可以與郵件，Slack等等內置的通知方式進行集成，也可以通過Webhook自定義告警處理方式。AlertManager即Prometheus體系中的告警處理中心。

**PushGateway**

由于Prometheus數據采集基于Pull模型進行設計，因此在網絡環境的配置上必須要讓Prometheus Server能够直接與Exporter進行通信。 當這種網絡需求無法直接滿足時，就可以利用PushGateway來進行中轉。可以通過PushGateway將內部網絡的監控數據主動Push到Gateway當中。而Prometheus Server則可以采用同樣Pull的方式從PushGateway中獲取到監控數據

Prometheus 儲存





Metrics 種類

Counter記數: 只增不減, Http請求數, 下單數

Gauge 測量: 量測數值

Histroggram 直方圖: 統計樣本分佈

Summnary 統計百分位區間

https://www.youtube.com/watch?v=28gisOsxb0s&list=PLoZQ0sz6CBHFm5i5fh9d1wowFZU8\_5XjH&index=119