評論(no need to justify)：

Identify a context where this technology is appropriate ~~and justify your answer~~

Pros

* base-PULL，無須安裝代理。
* Prometheus只要添加label就能輕易抓取K8S cluster(其可用於部署和管理多機器內的多個container)的指標。
* 支援許多客戶端library和第三方Exporter。
* 針對單節點儲存使其操作簡單。
* 可視化和監控警報對使用者很友善。

Identify a context where this technology is not appropriate ~~and justify your answer~~

Cons

* 內建的query採用PromQL語言，其需綁定Grafana，需要額外進行Grafana [[[1]](#footnote-1)]的設置。
* 不支援clustered storage，對於監控指標的數量有先天限制。
* 監控系統的通病，CAP中的可用性(availability)優先於一致性(consistency)，可能導致部分副本數據丟失。
* 可擴展性(scalability)差，不適合儲存龐大或長期資料 [[[2]](#footnote-2)]，大規模使用需要第三方component(e.g. Thanos, Cortex, etc.)才能實現 [[[3]](#footnote-3)]，這些第三方component也都有各自的問題(不展開討論)。
* 儲存指標而非儲存log，需要第三方的log傳送和分析工具才能保存log(e.g. Loki, ELK stack, etc.)。

1. <https://grafana.com/> [↑](#footnote-ref-1)
2. <https://prometheus.io/docs/prometheus/latest/storage/> [↑](#footnote-ref-2)
3. <https://www.youtube.com/watch?v=3pTG_N8yGSU> [↑](#footnote-ref-3)