



2022/06/15

New Relic User Group Vol.3

触ってすぐに感じた

New Relicの”Codeな”管理の体験が素晴らしい話

平山 慎 (Hirayama Makoto)



R&Dイノベーション本部
サービスイノベーション部

趣味

ゲーム (Apex Legends Mobile)
読書 (小説 or 技術書)

好きな技術

コンテナ関連

IaC (Infrastructure as Code)



業務紹介：リアルタイム データ前処理（ETL）基盤



システムとチームについて

- ✓ 約500台の仮想サーバー＋マネージドサービス
- ✓ 約30TB/day(圧縮後)の出力
- ✓ メインの開発メンバーは13人

社内でよりコアなシステムへ 高まる安定性の需要



“NTTドコモ IDAP” で検索すると
様々な記事が出てきます



new relic[®] のここが凄い

1 価格体系

- ✓ New Relicの**特異性**の一つ
- ✓ 弊システムへの導入の大きな理由の一つ

2 User eXperience の良さ

- ✓ 検索不要レベルで使いやすい User Interface
- ✓ **NRQL**を中心とした”Codeな”管理



new relic[®] のここが凄い

1 価格体系

- ✓ New Relicの**特異性**の一つ
- ✓ 弊システムへの導入の大きな理由の一つ

2 User eXperience の良さ

- ✓ 検索不要レベルで使いやすい User Interface
- ✓ NRQLを中心とした”Codeな”管理

new relic の凄さ：価格体系

- 有料プランを適用する**ユーザー数**
- 100GB/月を超えてNew Relic Observability Platformへ送信したデータ量 \$0.30/1GB
Data Plus \$0.50/1GB

➔ **サーバー台数やエージェントの数では課金されない**

少人数でスケールの大きいシステムを取り扱う体制に最適

- ✓ 弊チームと弊システムの状況がマッチ

監視対象の**追加**が容易

- ✓ **機能追加**や**スケールアウト**による監視対象の増大
- ✓ **一時的な**開発用のサーバーや検証用サーバーへの利用



new relic[®] のここが凄い

1 価格体系

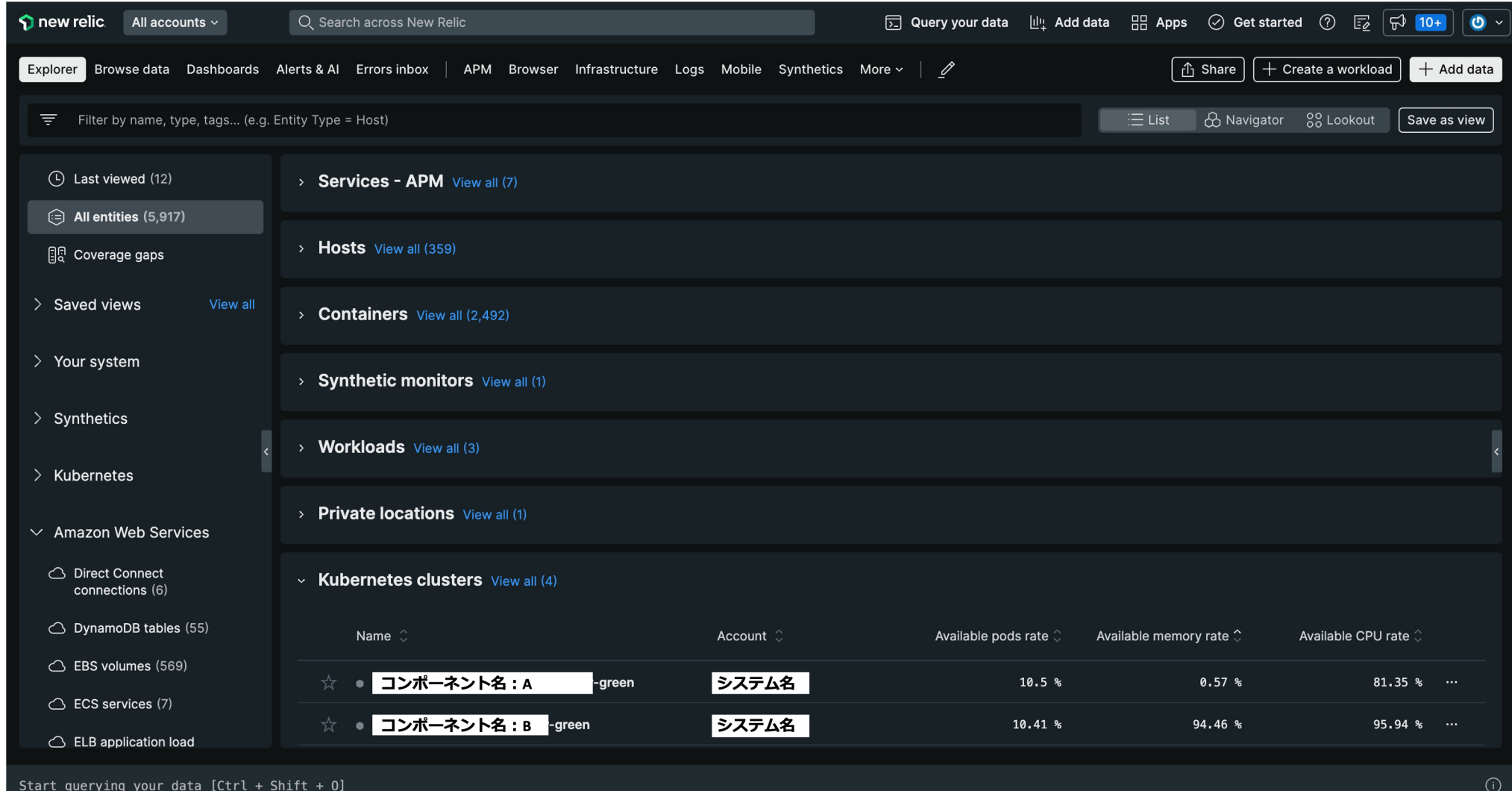
- ✓ New Relicの**特異性**の一つ
- ✓ 弊システムへの導入の大きな理由の一つ

2 User eXperience の良さ

- ✓ 検索不要レベルで使いやすい **User Interface**
- ✓ **NRQL**を中心とした”Codeな”管理

new relic の凄さ：UXの良さ（検索不要なUI）

やりたいことを実現する上で**キーワードで検索することはほぼ必要なく**
全てがNew Relic Observability Platform 上で解決



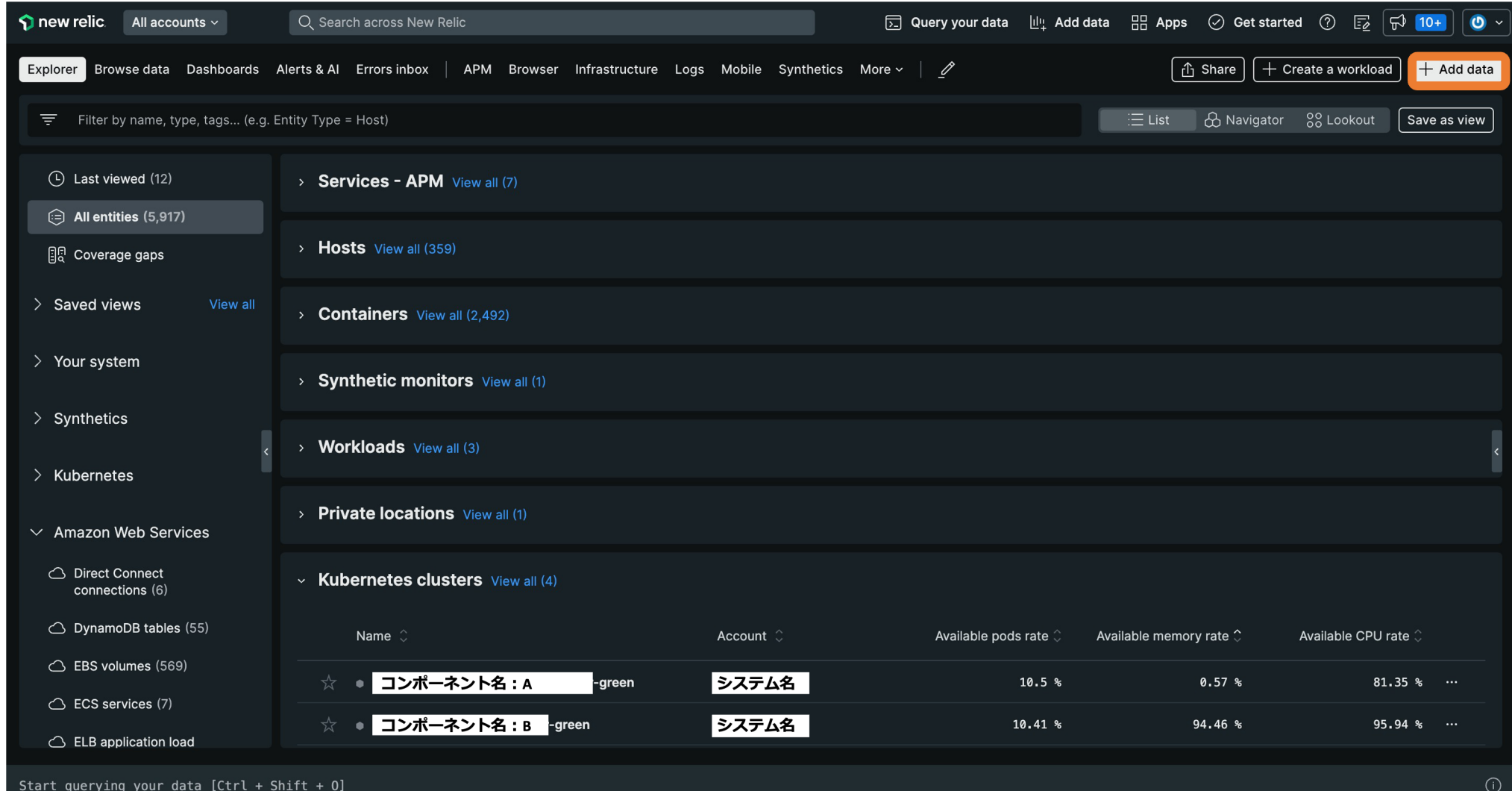
The screenshot displays the New Relic Observability Platform interface. The top navigation bar includes the New Relic logo, account selection, a search bar, and utility buttons like 'Query your data', 'Add data', 'Apps', 'Get started', and user profile. The main content area is divided into a left sidebar and a main panel. The sidebar contains navigation options such as 'Explorer', 'Browse data', 'Dashboards', 'Alerts & AI', 'Errors inbox', 'APM', 'Browser', 'Infrastructure', 'Logs', 'Mobile', 'Synthetics', and 'More'. The main panel shows a list of entities under the 'Kubernetes clusters' section, with columns for Name, Account, Available pods rate, Available memory rate, and Available CPU rate.

Name	Account	Available pods rate	Available memory rate	Available CPU rate
☆ ● コンポーネント名 : A -green	システム名	10.5 %	0.57 %	81.35 %
☆ ● コンポーネント名 : B -green	システム名	10.41 %	94.46 %	95.94 %

new relic の凄さ : UXの良さ (検索不要なUI)

AWSのデータをNew Relicで監視したい

→ + Add data

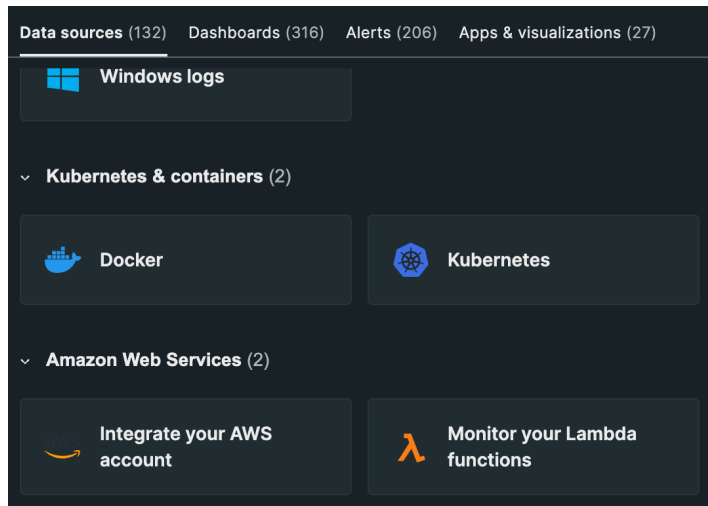


The screenshot shows the New Relic dashboard interface. The top navigation bar includes the New Relic logo, account selection, a search bar, and various utility buttons like 'Query your data', 'Add data', 'Apps', 'Get started', and a notification bell. Below the navigation bar, there are tabs for 'Explorer', 'Browse data', 'Dashboards', 'Alerts & AI', 'Errors inbox', 'APM', 'Browser', 'Infrastructure', 'Logs', 'Mobile', 'Synthetics', and 'More'. A prominent orange button labeled '+ Add data' is visible in the top right corner of the main content area. The main content area is divided into a left sidebar and a main panel. The sidebar contains sections for 'Last viewed (12)', 'All entities (5,917)', 'Coverage gaps', 'Saved views', 'Your system', 'Synthetics', 'Kubernetes', and 'Amazon Web Services'. The 'Amazon Web Services' section is expanded, showing 'Direct Connect connections (6)', 'DynamoDB tables (55)', 'EBS volumes (569)', 'ECS services (7)', and 'ELB application load'. The main panel displays a list of entity categories: 'Services - APM (7)', 'Hosts (359)', 'Containers (2,492)', 'Synthetic monitors (1)', 'Workloads (3)', 'Private locations (1)', and 'Kubernetes clusters (4)'. The 'Kubernetes clusters' section is expanded, showing a table with columns for 'Name', 'Account', 'Available pods rate', 'Available memory rate', and 'Available CPU rate'. Two clusters are listed: 'コンポーネント名 : A' and 'コンポーネント名 : B', both with a status of '-green'.

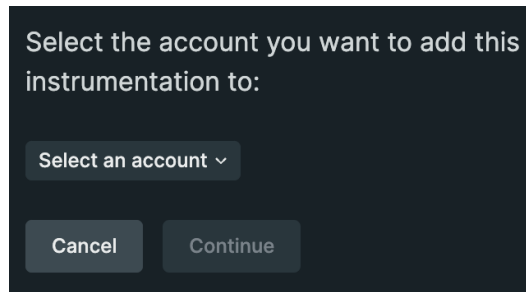
Name	Account	Available pods rate	Available memory rate	Available CPU rate
☆ ● コンポーネント名 : A -green	システム名	10.5 %	0.57 %	81.35 %
☆ ● コンポーネント名 : B -green	システム名	10.41 %	94.46 %	95.94 %

AWSのデータをNew Relicで監視したい

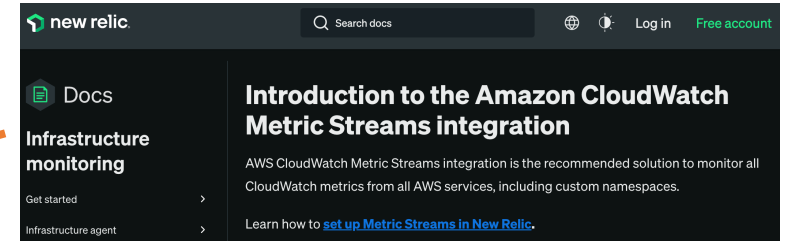
1



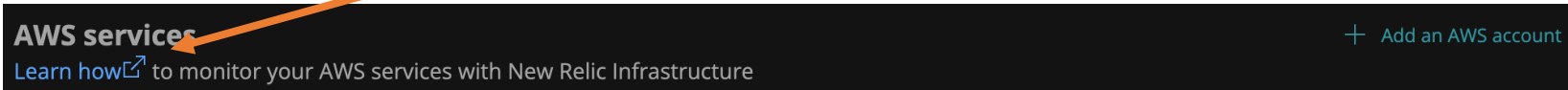
2



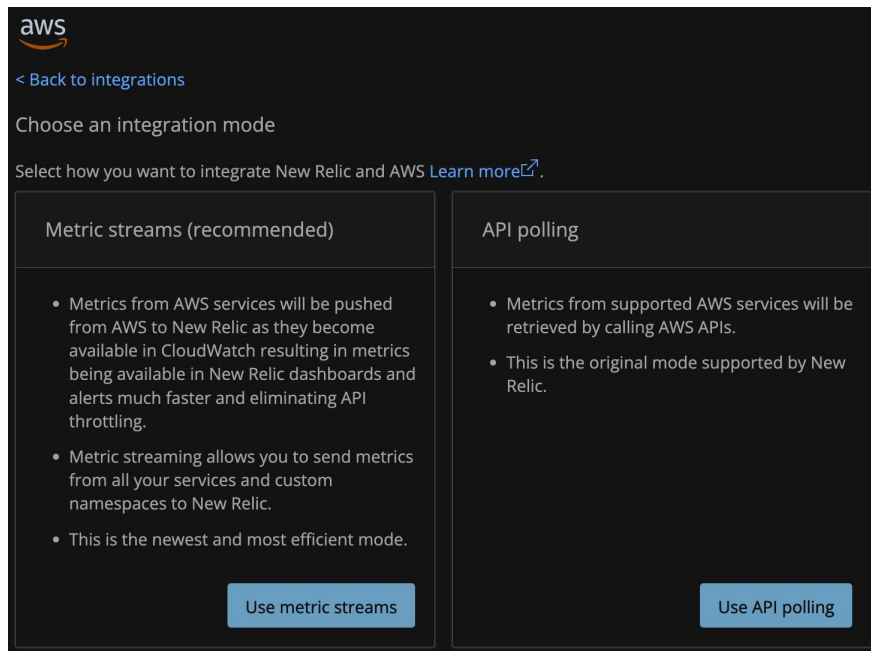
詳細ページへのリンク



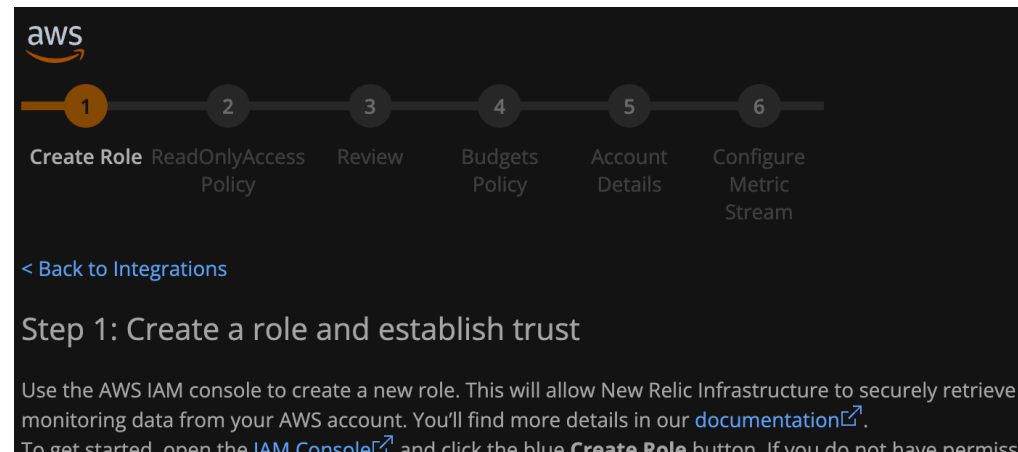
3



4



5



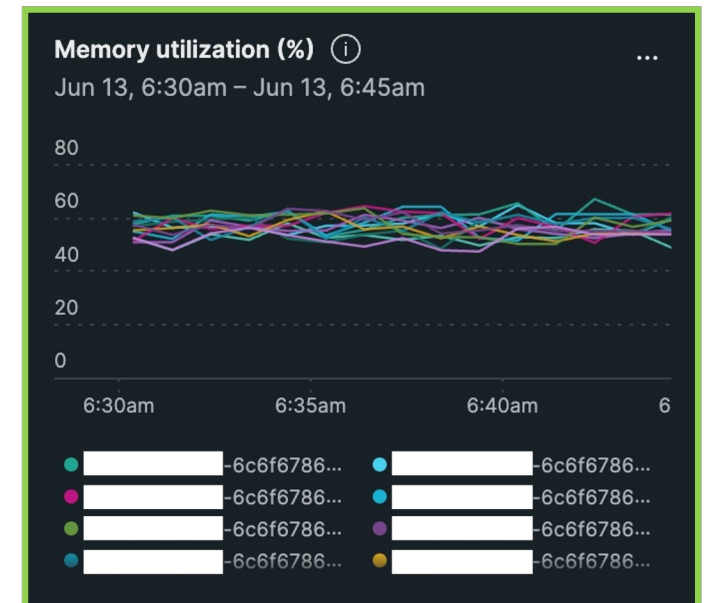
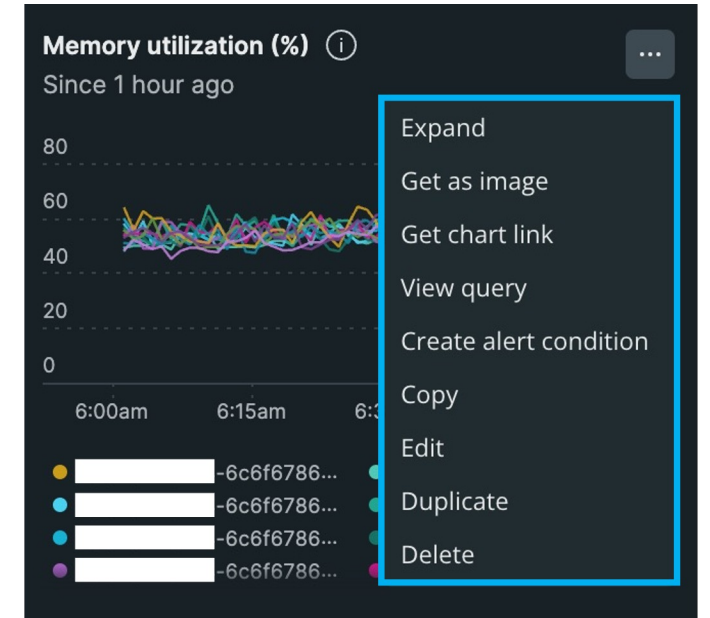
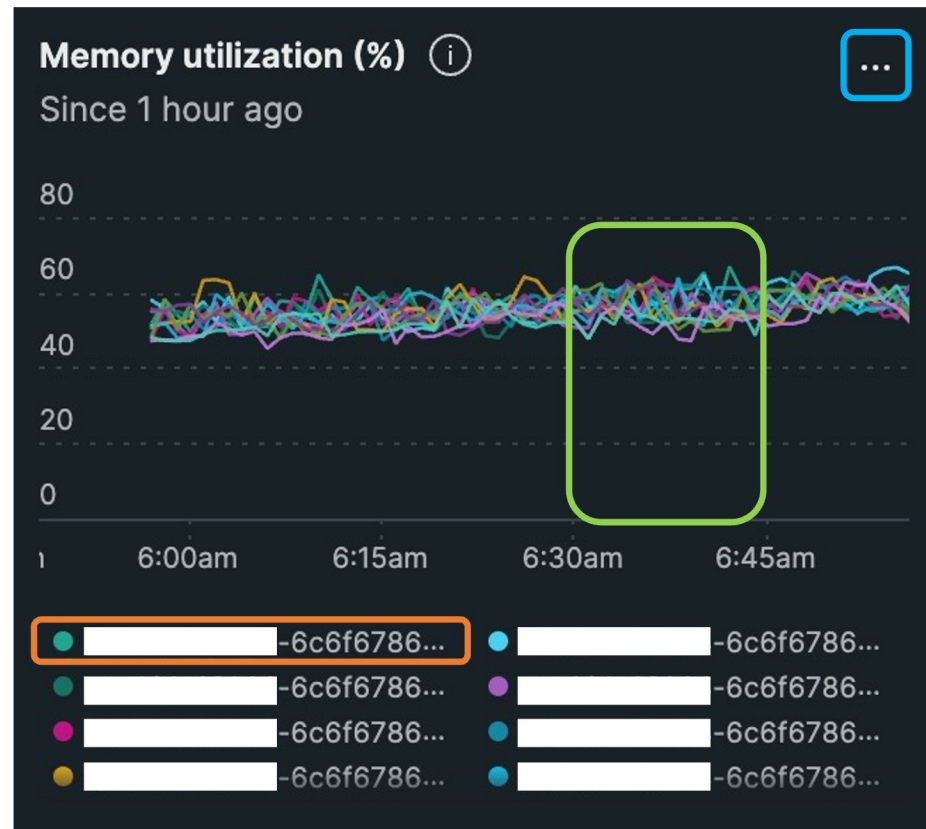
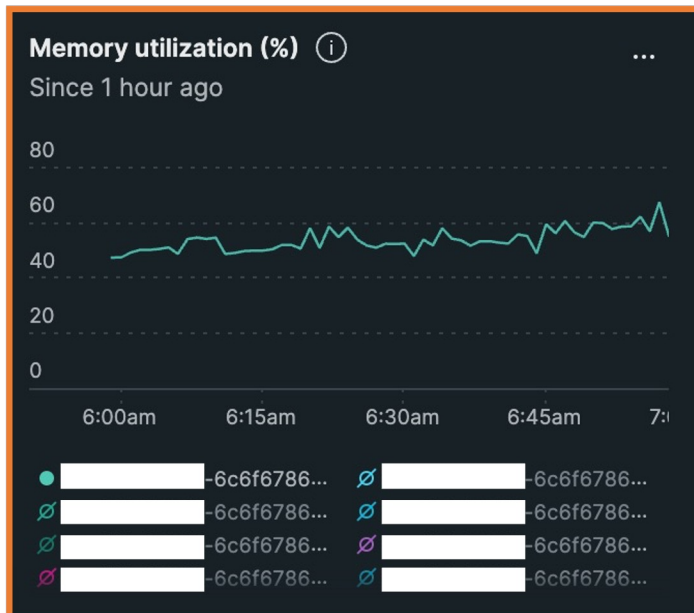
6



new relic の凄さ : UXの良さ (直感的なUI)

欲しいUIが
欲しい粒度で
欲しい場所に存在

直感的で楽しい
優秀なUIの勉強にも？





new relic® のここが凄い

1 価格体系

- ✓ New Relicの**特異性**の一つ
- ✓ 弊システムへの導入の大きな理由の一つ

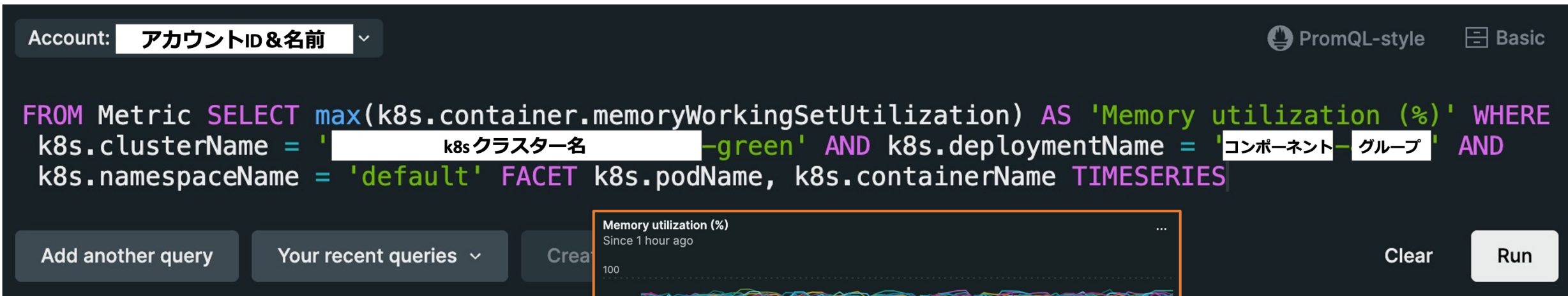
2 User eXperience の良さ

- ✓ 検索不要レベルで使いやすい User Interface
- ✓ **NRQL**を中心とした”Codeな”管理

new relic の凄さ : UXの良さ (“Codeな”管理)

New Relic Query Language

Metrics Event Log Trace へ適用するクエリ言語



The screenshot shows the New Relic query interface. At the top, there's an account dropdown and a 'PromQL-style' button. The main area contains a query: `FROM Metric SELECT max(k8s.container.memoryWorkingSetUtilization) AS 'Memory utilization (%)' WHERE k8s.clusterName = 'k8s クラスター名 -green' AND k8s.deploymentName = 'コンポーネント -グループ' AND k8s.namespaceName = 'default' FACET k8s.podName, k8s.containerName TIMESERIES`. Below the query are buttons for 'Add another query', 'Your recent queries', and 'Run'. A line chart titled 'Memory utilization (%)' shows data from 2:00pm to 2:50pm. The chart has a legend with entries like 'C_ID - G_ID - 6c6f6786b7-45qrb, Pod名'.

- ✓ SQLライクな構文
- ✓ 優秀な補完機能

構文はもちろんField値だけでなくValue値も補完対象

- ✓ 切り口が自由自在なFACET

k8s cluster > k8s.deployment の中のものを k8s.pod > k8s.container 毎に

- ✓ 過去のクエリの保存機能

k8s cluster

deployment

pod



NRQLのGUIによる副次的な生成

The screenshot shows the 'Data explorer' interface with the 'Metrics' tab selected. A search bar contains 'k8s.'. A list of metrics is displayed, including 'k8s.container.fsUsedBytes', 'k8s.container.fsUsedPercent', 'k8s.container.isReady', 'k8s.container.memoryLimitBytes', 'k8s.container.memoryMappedFileBytes', 'k8s.container.memoryRequestedBytes', 'k8s.container.memoryUsedBytes', and 'k8s.container.memoryUtilization'. The 'k8s.container.memoryUtilization' metric is selected, and a dropdown menu shows aggregation functions: Latest, Average, Count, Sum, Max, and Min.

The screenshot shows the 'Dimensions' section of the interface. A search bar contains 'k8s.'. A list of dimensions is displayed, including 'k8s.clusterName', 'k8s.containerId', 'k8s.containerImage', 'k8s.containerImageId', 'k8s.containerName', 'k8s.deploymentName', 'k8s.namespaceName', 'k8s.nodeIp', 'k8s.nodeName', and 'k8s.podName'. The 'k8s.containerName' dimension is selected.



The screenshot shows the NRQL query editor with the following query: `NRQL SELECT max(`k8s.container.memoryUtilization`) FROM Metric FACET `k8s.containerName` WHERE `k8s.clusterName` = 'k8s クラスター名' AND `k8s.deploymentName` = 'deploy名' AND `k8s.podName` = 'pod名' SINCE 30 MINUTES AGO TIMESERIES`. Below the query editor, a context menu is open with options: 'Get as image', 'Get chart link', 'Create alert condition', and 'Add to dashboard'. To the right, a line chart titled 'k8s.container.memoryUtilization' is displayed, showing memory utilization over time for various containers.

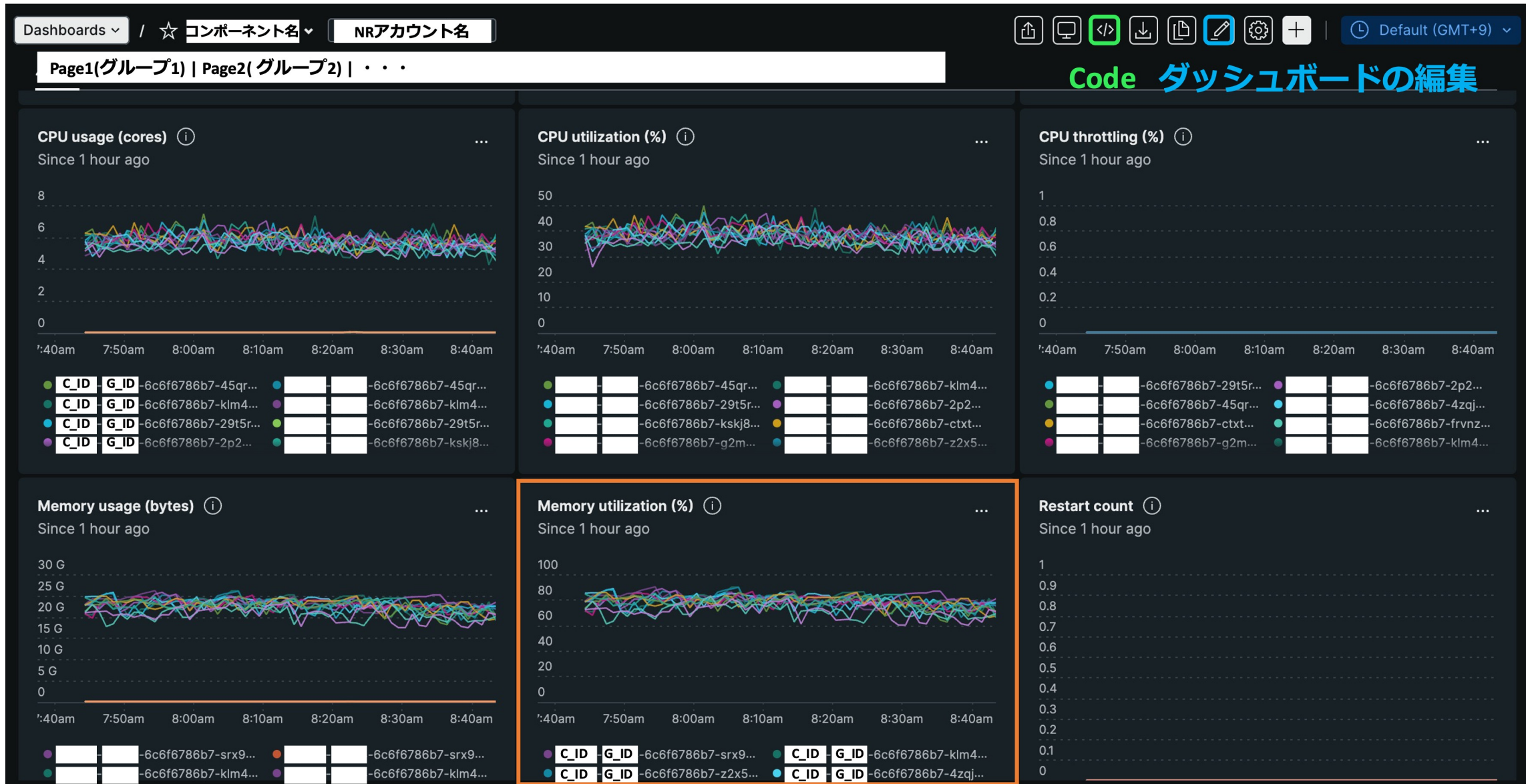


The screenshot shows the final NRQL query output in the GUI. The query is: `SELECT max(`k8s.container.memoryUtilization`) FROM Metric FACET `k8s.containerName` WHERE `k8s.clusterName` = 'k8s クラスター名' AND `k8s.deploymentName` = 'deploy名' AND `k8s.podName` = 'pod名' SINCE 30 MINUTES AGO TIMESERIES`. The interface includes buttons for 'Add another query', 'Your recent queries', 'Create alert', 'Clear', and 'Run'.

NRQLはほぼGUIでも作成可能

ダッシュボードの作成時の"Codeな"管理

GUIやNRQLを用いて要素を作成しまとめたダッシュボード



ダッシュボード
GUI

管理
ダッシュボード

```
{  
  "title": "Memory utilization (%)",  
  "layout": {  
    "column": 5,  
    "row": 7,  
    "width": 4,  
    "height": 3  
  },  
  "linkedEntityGuids": null,  
  "visualization": {  
    "id": "viz.line"  
  },  
  "rawConfiguration": {  
    "dataFormatters": [],  
    "nrqlQueries": [  
      {  
        "accountId": "アカ名",  
        "query": "FROM Metric SELECT max(k8s.container.  
memoryWorkingSetUtilization) AS 'Memory utilization (%)' WHERE k8s.  
clusterName = 'サービス名-C_ID-green' AND k8s.deploymentName  
= 'C_ID-G_ID' AND k8s.namespaceName = 'default' FACET k8s.podName,  
k8s.containerName TIMESERIES" "C_ID": Unknown word.  
      }  
    ],  
    "thresholds": []  
  }  
},  
}
```

NRQL

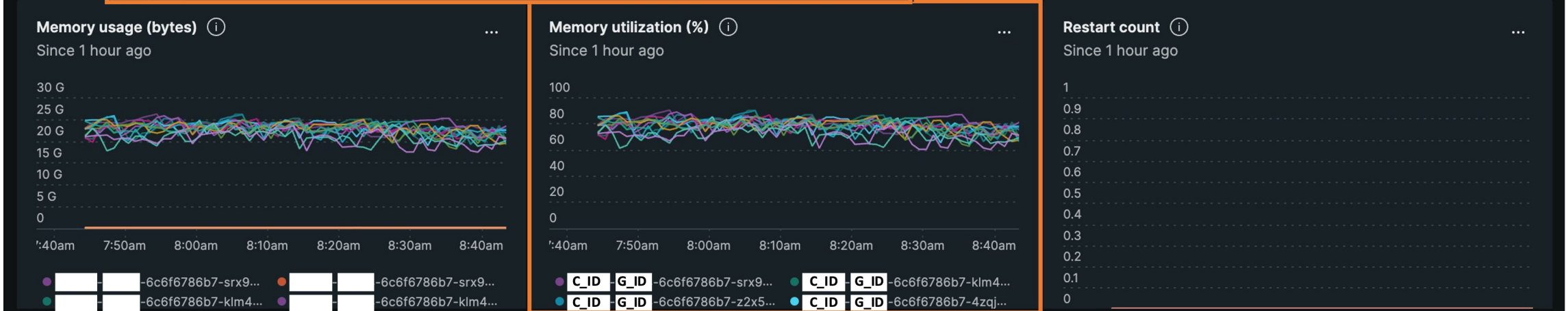
Memory utilization (%) や

サービス名

C_ID(Component_ID)

G_ID(Group_ID) の変更で

容易に複製や追加が可能



 new relic の凄さ：UXの良さ (“Codeな”管理)

“Code” を用いた

“高度”なマネージメントが

簡単に可能





new relic® のここが凄い

- 1 有料ユーザー毎の価格体系
- 2 User eXperience の素晴らしさ

Basicな話のみでしたが

New Relicには様々な高機能(Code Stream等)が存在

必要に応じて導入し快適な**Observability Life**を