

[ポスター発表] 研究報告

# Situation Awarenessと認知心理学にもとづいた マイクロサービス型システム向け監視ダッシュボードの設計

林 友佳<sup>1,a)</sup> 松原 克弥<sup>1,b)</sup> 鷲北 賢<sup>2</sup> 坪内 佑樹<sup>2</sup>

## 1. はじめに

マイクロサービス型システムは、機能要素毎に分割したサービスをそれぞれ独立したコンポーネントとして実装し、ネットワーク等を介してそれらを疎結合することで構築したシステムである。負荷に応じてコンポーネントの数を増減させたり、不具合が発生したコンポーネントを交換したりすることが容易になり、システムの可用性や拡張性を向上させることができる。一方で、モノリシック型のシステムと比較して、システムが大規模化・複雑化することで、運用や監視に対する負荷や難易度が増す課題が存在する。

マイクロサービス型システムの監視における最大の課題のひとつは、コンポーネントの数や稼働状況が頻繁に変化するために、システム内の状態変化を把握することが難しいことである。システムがアクセス負荷上昇やコンポーネント異常終了などに対処することで、コンポーネントの数、コンポーネントが稼働するマシンノード、コンポーネント間の連携状況が変化する。したがって、個々のコンポーネントのメトリックを監視しているだけでは、これらの状況変化の発生やその理由を特定することができず、結果として、システム全体の状態を把握できないことにつながる。

また、マイクロサービス型システムでは、監視すべきコンポーネントの数が多いことから、障害原因の特定が難しくなる課題も存在する。マイクロサービス型システムのような分散システムでは、複数の要因の組み合わせによって障害が発生する傾向がある [1]。多数のコンポーネントが分散して動作し、それらコンポーネント間の連携にともなう暗黙的な依存関係の発生により、障害がコンポーネント間を伝播する [2]。結果として、障害が伝播した数多くのコンポーネントに対する大量のアラートのなかから、原因となったコンポーネントを特定しなければならない。前述

の状態変化にともなってコンポーネント間の依存関係も変化することから、障害原因の特定のために分析すべきコンポーネントの範囲を特定することも困難となる。

著者らは、マイクロサービス型システムの状態監視や障害対応において、コンポーネント間の依存関係を把握することが最も重要であると考えている。そこで、本研究では、コンポーネントの依存関係を軸としたシステム監視ダッシュボードを提案する。コンポーネントとその依存関係の変化を可視化することで、システム全体の状態把握と変化の認知を支援する。さらに、可視化したコンポーネント間の依存関係に合わせて、各コンポーネントのメトリックを表示したり比較する機能を実現することで、障害の伝播をトレースすることを支援できる。

依存関係を軸とした監視ダッシュボード User Interface (以降, UI) では、コンポーネントの数に応じて表示する情報量や操作の対象が指数関数的に増加する。そのため、監視エンジニアが処理すべき情報量の増加が、システム状態に関する認知へ影響を与える可能性がある。本稿では、Situation Awareness (以降, SA) と認知心理学の知見から、監視ダッシュボードの UI 設計における要件を導出し、それら要件を満たすダッシュボード UI 設計を示す。

## 2. SA と認知的観点による UI 設計要件の導出

SA は、一定の時間と空間の中で現状の構成要素を知覚し、その意味を理解することで現状を把握したのち、近い将来の状態を予測するプロセスである [3]。Endsley ら [4] は、SA のそれぞれの段階を支援し強化するために重要となる、50 のシステム設計法則を提案している。本研究では、50 の設計法則のうち、チームで共有するシステムに関する法則、自動化に関する法則、アラームなどの視覚的な UI 以外に関する法則、不確実な情報が収集された際に関する法則を除いた 16 の法則から、監視における SA のプロセスを促す 4 つの UI 設計要件を定めた；C1. 情報変化の表示, C2. 情報のグループ化, C3. 概要の提供, C4. 操作の複雑性軽減。

また、Few [5] が提案した、認知心理学的の観点による

<sup>1</sup> 公立はこだて未来大学  
Future University Hakodate, Hakodate, Hokkaido, Japan

<sup>2</sup> さくらインターネット株式会社  
SAKURA internet Inc.

a) g2120042@fun.ac.jp

b) matsu@fun.ac.jp

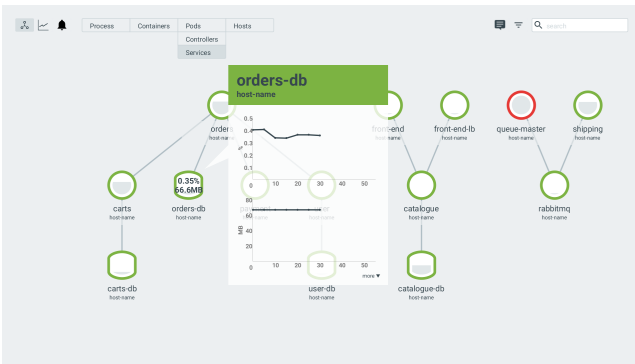


図 1 提案するダッシュボードの UI

ダッシュボード設計法則を本監視ダッシュボードの設計に適用するために、さらに6つのUI設計要件を定めた：C5. 重要情報の一画面集約化, C6. 情報への直接的なインタラクション, C7. 情報の強調手法, C8. 色・形・動きの一貫性, C9. 情報の比較に対する支援, C10. グループ化表示へのゲシュタルト法則の適用。

### 3. 依存関係を軸としたダッシュボード UI

本ダッシュボードは、コンポーネントとそれらの依存関係を可視化したグラフを主画面とし、グラフ要素への直接的なインタラクションによる操作と、グラフへのオーバーレイ表示によるメトリック表示・比較を行うUIにより、システム状態把握と障害発生時の分析作業を支援する(図1)。コンポーネントとそれらの依存関係を可視化したグラフは、C2, C4, C5, C8, C10の要件に沿って設計している。コンポーネントの可視化には、サービスの種類ごとに特定の図形を用いる、依存関係のあるコンポーネント間に線を描画するなどといった、グループ化の手法を取り入れている。このようなグループ化を行うことで、運用エンジニアにかかる負荷を抑えつつ、サービスの種類やコンポーネント間の依存関係を認識できるようにしている。ダッシュボード上の配置は、データの流れて上から下、依存関係の多いコンポーネントから順に左から右となるように配置する。さらに、ユーザである監視エンジニアの要求に合わせて、指定した特定のコンポーネントをダッシュボードの上部に配置したり、フィルタリングを行い一部のコンポーネントのみを表示したりすることも可能である。

さらに、依存関係と同時にシステムの状態を迅速に認識できるように、コンポーネントを表す図形に用いる色を工夫することで、視覚的示唆による強調表現を行う。図形の枠線の色は、CPU使用率やメモリ使用量の値が大きい場合は赤色にし、値が中くらいの場合は黄色、小さい場合は緑色とするといったように区別している。図形の内部は、白とグレーで塗りつぶされており、グレーの面積の広さによってCPU使用率とメモリ使用率の値が示唆される。例えば、図1では、queue-masterというコンポーネントのみ図形

の枠線が赤く、内部のグレーの面積が他のコンポーネントより広いため、現在負荷の高いコンポーネントであることを認知できる。この強調表現は、C3, C4, C7の要件を踏まえたものであり、ダッシュボード上の情報量が増加した際も、過度に情報を絞り込むことなく重要な情報を示唆できる。

また、個々のコンポーネントに関する監視メトリックは、コンポーネントを表す図形をクリックした際に折れ線グラフを描画する。複数のコンポーネントの折れ線グラフを表示した後にドラッグをすることで、1つのグラフ上に複数コンポーネントの監視メトリックの推移を比較表示できる。このメトリックを時系列に可視化する機能は、C1, C4, C6, C9の要件に沿った設計となっている。依存関係と同時に監視メトリックの時間的な推移も可視化することで、現在までにシステムに発生した変化を認識させ、システムの現状の理解や今後の予測を支援できる。加えて、直感的な操作で複数のコンポーネントのグラフを統合できるようにすることで、監視メトリックの比較を容易にし、システムの現状の理解を促進できる。

### 4. 今後の課題

現在、導出したUI要件と設計したダッシュボードを評価するための手法について検討を行っている。一部要件を対象としたアンケートによる予備評価では、本要件の有効性を示唆する結果が得られた。今後、システム運用を担う実エンジニアを被験者とした評価実験によって、マイクロサービス型システムの規模や複雑性に対する本監視ダッシュボードの有効性を検証していきたい。

### 参考文献

- [1] Yuan, D., Luo, Y., Zhuang, X., Rodrigues, G. R., Zhao, X., Zhang, Y., Jain, P. U. and Stumm, M.: Simple testing can prevent most critical failures: An analysis of production failures in distributed data-intensive systems, *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*, pp. 249–265 (2014).
- [2] Tsubouchi, Y., Furukawa, M. and Matsumoto, R.: Transtracer: Socket-Based Tracing of Network Dependencies Among Processes in Distributed Applications, *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1206–1211 (2020).
- [3] Endsley, M.: Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors Journal*, *Human Factors: The Journal of the Human Factors and Ergonomics Society*, Vol. 37, pp. 32–64 (1995).
- [4] Endsley, M., Bolté, B. and G., J. D.: *Designing for Situation Awareness: An Approach to User-Centered Design*, Taylor & Francis (2003).
- [5] Few, S.: *INFORMATION DASHBOARD DESIGN*, O'Reilly Media, Inc. (2006).