

**沖縄電力株式会社**  
**性能ダッシュボード・性能カスタマイズオプション事例**  
**- 性能カスタマイズオプション導入でシステム負荷が1/3に減少 -**

# はじめに

---

kintoneのワイドコースは、大規模利用向けにアプリ数・スペース数・APIリクエスト数を拡張し、専用機能で円滑な運用を支援します。

本書は、kintoneのエンタープライズ企業への導入支援で得られた性能カスタマイズオプションの事例資料です。

## 本書の目的

---

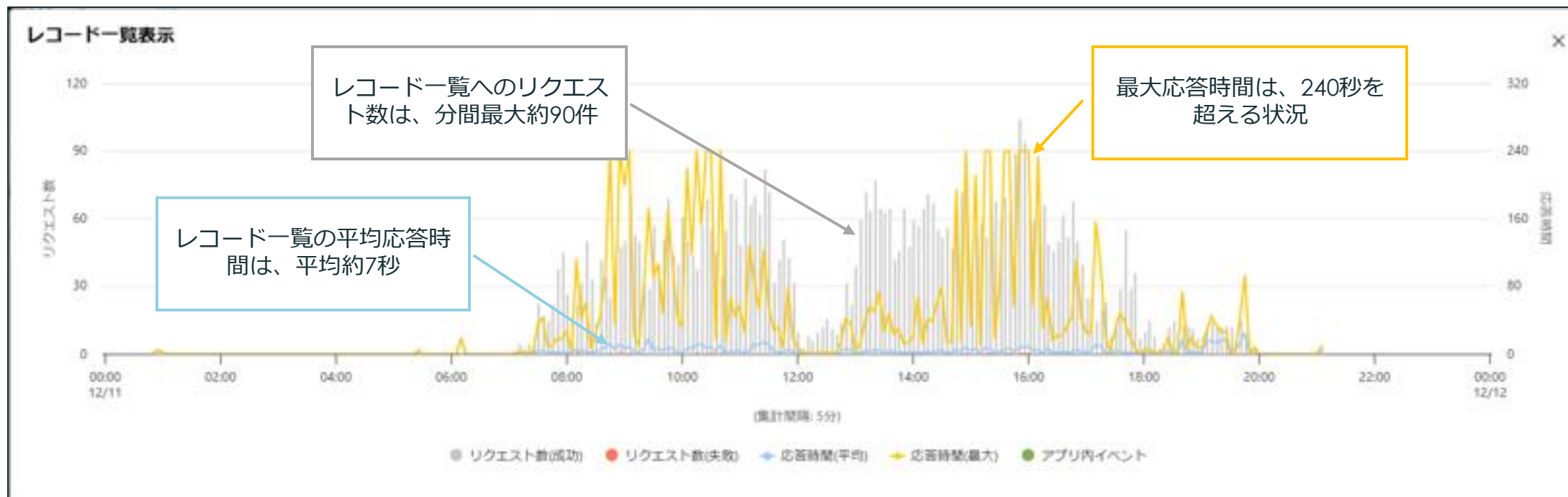
kintoneは部門単位や業務単位での迅速なアプリ構築に強みを持つプラットフォームとして評価されてきました。一方で、全社規模での利用を想定する場合には、レコード件数の増加、同時アクセス数の拡大、複雑な業務要件への対応など、設計および運用面で考慮すべき技術的なポイントが増えてきます。

これからkintoneを全社基盤として活用することを検討している方、および既に導入済みで利用範囲の拡大を計画している方にとって、本書が技術的な判断材料の一助となることを目的としています。

## 概要

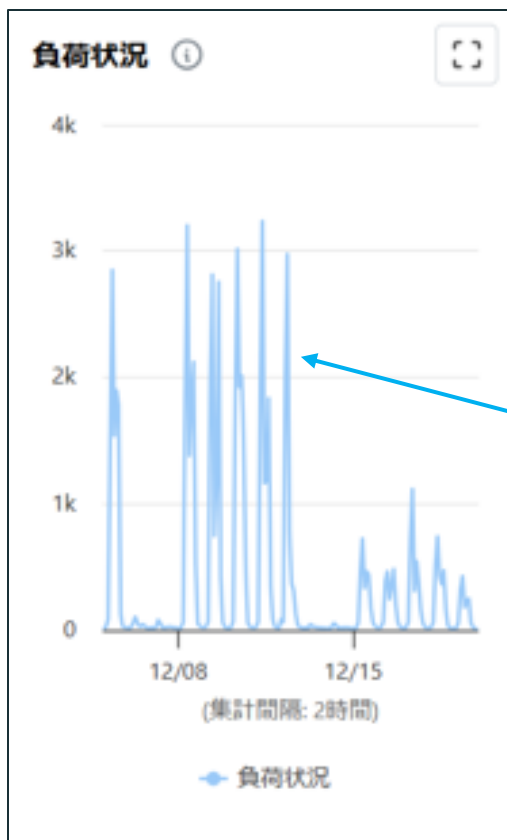
「設備管理」アプリの利用ユーザー数が増加したことにより、データベース処理時間が遅延し、結果としてシステム全体の負荷増大の要因となっていました。

「設備管理」アプリのレコード一覧へのアクセスは、分間最大90アクセスにも関わらず、多くのリクエストが10秒以上かかり、長いリクエストで240秒を超えてしまう場合もありました。



レコード一覧の画面操作の成功/失敗のリクエスト数、平均/最大の応答時間（秒）及びイベント発生状況がグラフとして表示されています。

# システムへの影響



「設備管理」アプリのレコード一覧が表示されるまでに時間がかかるだけでなく、データベース処理遅延の影響で、時間あたりのリクエスト並列処理数が3,000を超えていました。その結果、システム全体の負荷増大の要因となりました。

時間あたりのリクエストの  
並列処理が3,000を超える  
状況

時間あたりのリクエストの並列処理数の平均値（応答時間の平均×リクエスト数/集計時間）がグラフとして表示されます。

負荷状況の値が高いと並列処理数が多く、システムの負荷が高くなっている可能性があります。

# レコード一覧設定内容

「設備管理」アプリ起動時に表示されるデフォルトのレコード一覧において、設定されている絞り込み条件が複雑な内容となっていました。

「設備管理」アプリのデフォルトのレコード一覧に、「複数選択系フィールド」に分類される「組織」「ユーザー」を指定する絞り込みが設定されている状況でした。

絞り込み

条件

プロジェクト担当者 ▼ 次のいずれかを含む ▼ ユーザーを追加 🔍 👤 + -  
● 優先する組織 ×

作業担当者 ▼ 次のいずれかを含む ▼ ユーザーを追加 🔍 👤 + -  
● ログインユーザー ×

管轄・グループ ▼ 次のいずれかを含む ▼ ユーザーを追加 🔍 👤 + -  
● ログインユーザー ×

すべての条件を満たす  いずれかの条件を満たす

すべてクリア

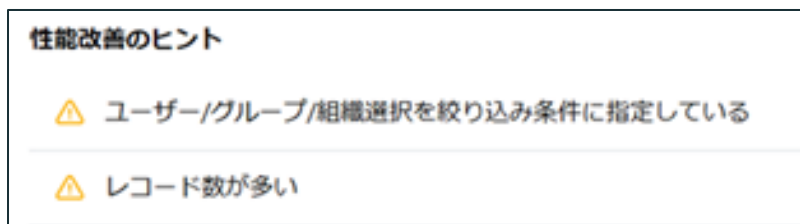
## 補足：複数選択系フィールドについて

- 「ユーザー」「組織」「グループ」選択フィールドを使った絞り込みは「複数選択系フィールド」に分類されます。「複数選択系フィールド」を使った絞り込みでは、データベースで実行されるクエリが複雑になります。
- 「複数選択系フィールド」に該当するフィールドには、「ユーザー」「組織」「グループ」以外にも、「ラジオボタン」「ドロップダウン」「チェックボックス」「複数選択」があります。

# 状況と対応内容

## ■ 状況

性能ダッシュボードに表示される「性能改善のヒント」において、「ユーザー/グループ/組織選択を絞り込み条件に指定している」「レコード数が多い」の警告が表示されている状況でした。



## ■ 対応内容

「設備管理」アプリのデフォルトのレコード一覧において、絞り込み条件などアプリの構成や使い方を大きく変更することなくボトルネックを解消することを目的としました。

そのため、性能カスタマイズオプションの中から、レコード一覧に関する3つの性能カスタマイズオプションを適用しました。

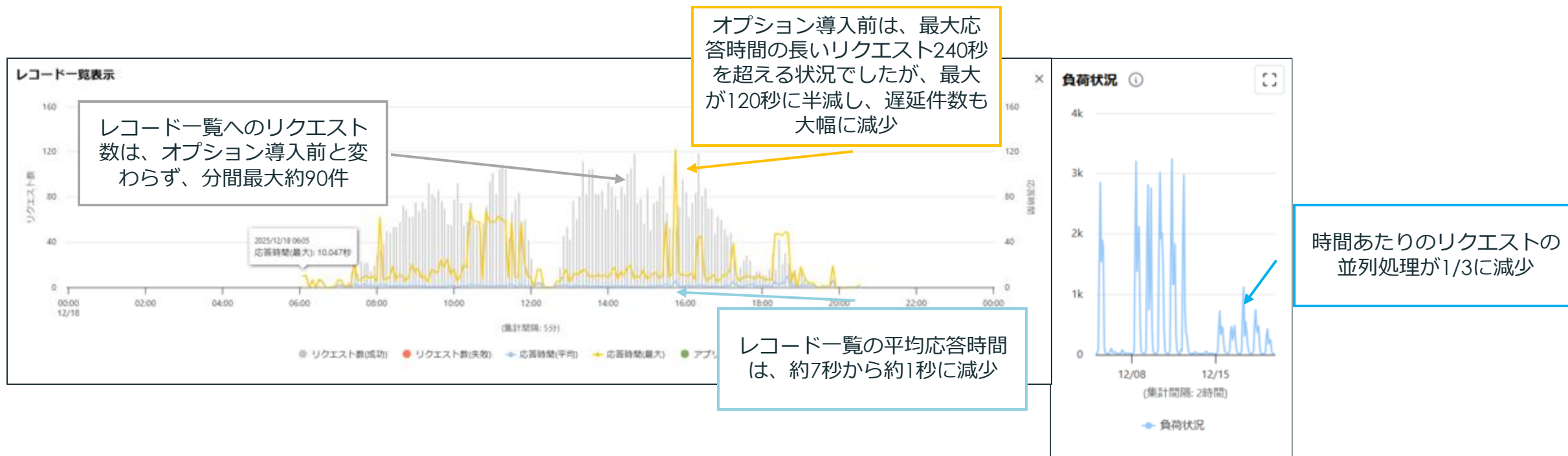
- レコード登録時のレコード番号連番保証設定を解除する
- 一斉大量書き込み用非同期レコード登録REST APIを有効化する
- グラフ表示用のデータ取得リクエストをリードレプリカに対して実行する
- レコード一覧表示用のデータ取得リクエストをリードレプリカに対して実行する
- レコード一覧表示時にデータを手動で取得する
- 関連レコード一覧のデータを手動で取得する
- レコード一覧表示時のレコードの総件数を手動で取得する
- レコード詳細画面の「次/前のレコードへ移動」時にキャッシュを利用し、データの絞り込み処理を省略する
- 複数選択、チェックボックス、ユーザー選択、グループ選択、組織選択フィールドの絞り込み時に実行する内部クエリを切り替える

# オプション導入結果

性能カスタマイズオプション適用後、レコード一覧表示の遅延が大幅に改善されました。

「設備管理」アプリのレコード一覧の平均応答時間は約7秒から約1秒に短縮され、システム全体の負荷も大幅に減少しました。

また、「設備管理」アプリのレコード一覧へのアクセスは、負荷状況が減少したことで、240秒を超えてしまうリクエスト数が大幅に減少しました。



## レコード一覧の複雑な絞り込みがシステム負荷を生む理由

kintoneアプリのレコード一覧に複雑な絞り込みを設定すると、データベースの処理に時間がかかります。また、アクセス数が増えることで、データベースで同時に実行される処理も増加し、レコード一覧表示の遅延やシステムの高負荷につながります。

アプリのレコード一覧に設定された絞り込みが、1秒未満で処理が完了するような設定であれば、分間アクセス数が90であっても問題はありません。

一方で、アプリのレコード一覧に設定した絞り込みの処理に10秒以上かかる場合には、同時実行される処理数が増加し、システムに大きな負荷を与える可能性があります。

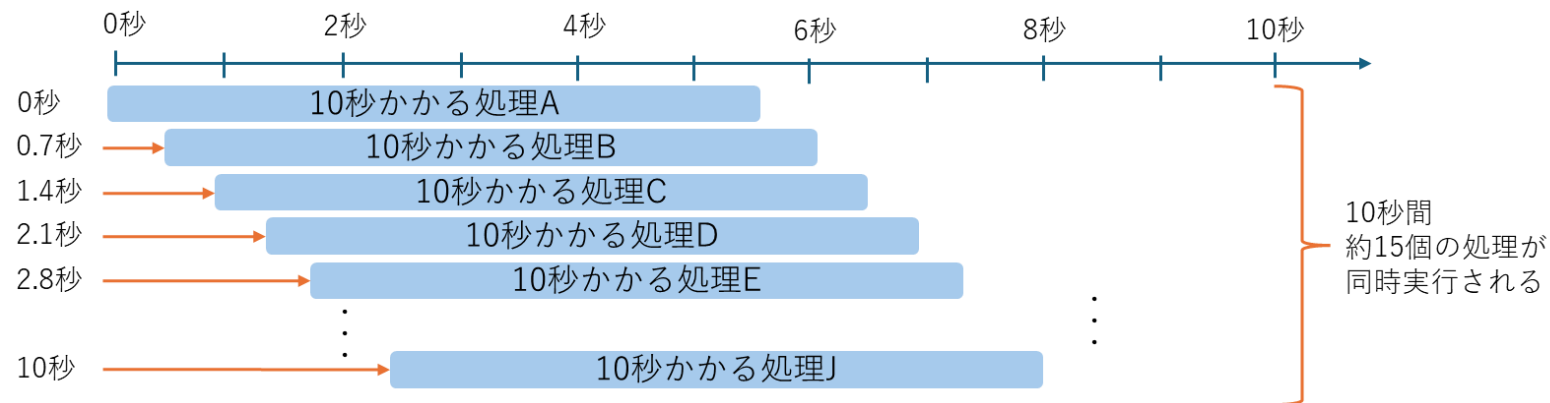
分間90回アプリのレコード一覧へアクセスする場合、秒間では約1.5回のアクセスが発生している計算になります。

このとき、1回の処理時間が10秒の場合、同時実行数の目安は次のようになります。

- $1.5 \text{ 処理/秒} \times 10 \text{ 秒} = \text{約}15 \text{ 処理}$

つまり、常に約15個の処理が同時に実行されている状態になります。

# レコード一覧の複雑な絞り込みがシステム負荷を生む理由



時間軸で見ると、以下のように処理が開始されます。

- 0秒：処理A 開始
- 0.7秒後：処理B 開始
- 1.4秒後：処理C 開始

それぞれの処理は10秒間終了しないため、結果としてデータベースには約15個の処理が積み重なった状態になります。処理の重なりによりデータベースの負荷が高まり、システム負荷増大の要因となります。

kintoneでは、信頼性と実績を持つリレーショナルデータベースを採用していますが、製品特性上、レコード件数が大規模になる場合には、レコード一覧の絞り込みなど事前の設計および運用面での配慮が重要となります。

## レコード一覧の複雑な絞り込みがシステム負荷を生む理由

---

参考となりますが、kintoneの性能に関する詳細は、kintone SIGNPOSTの「性能上の考慮点と改善策」および、エンジニア向けに詳細な検証結果を公開している「kintoneの性能」シリーズもあわせて参照してください。

### ▼kintone SIGNPOST 性能上の考慮点と改善策

<https://kintone.cybozu.co.jp/kintone-signpost/guide/performance.html>

### ▼cybozu developer network パフォーマンス

<https://cybozu.dev/ja/id/24631ddfbc022f852aa6e0f4/>

## 結論

---

エンタープライズ向けに提供している「ワイドコース」では、リクエスト数や応答時間などのメトリクスを可視化できるため、システムのボトルネックを把握することができます。

また、性能カスタマイズオプションを適用することで、kintoneアプリのレコード削除やアプリ設定の見直しといった運用変更を行うことなく、既存の業務構成を維持したまま大幅なパフォーマンス改善を実現できる点が大きな特長です。

業務プロセスやデータ蓄積方針を変更せずに性能課題へ対応できるため、現場への影響を最小限に抑えながら、エンタープライズ規模の利用にも耐えうる安定した運用基盤を提供することができました。

