

kintone大規模運用のための パフォーマンス設計ガイド

01 アクセス権の設定について

アクセス権：注意が必要な設定パターン

What：何が起きるか

- アクセス権の設定によっては、評価に時間がかかり、レコード一覧表示や一括取得・更新などに影響が生じる場合があります。
- また、一括更新などの処理においても影響を受ける場合があるため、下記の設定パターンには注意が必要です。

設定	リスク
レコード/フィールドのアクセス権を設定している	アプリ単位より評価コストが大きい
アクセス権の対象者に多数のユーザー/組織/グループを指定している	設定数によりアクセス権の評価コストが大きくなる

How：設計時の予防策

- アクセス権の事前評価は「レコード数 × アクセス権の設定数 × アクセス権をもつユーザー/組織/グループの数」の分だけ実行され、その結果がデータベースに保存されます。下記を減らすことでアクセス権評価の負荷を軽減できます。
 - アクセス権の設定数
 - アクセス権をもつユーザー/組織/グループの数
 - レコード数

補足

- レコード/フィールドのアクセス権が多い場合は、スペース、アプリのアクセス権で代用できないかご検討ください。
- レコード/フィールドのアクセス権で、アクセス権の対象者に多数の「ユーザー/組織/グループ」を設定している場合、不要な設定が含まれていないかご確認ください。

アクセス権：運用後の設定変更

What：何が起きるか

- 大量のレコードを保持しているアプリやアクセスが集中するアプリにおいて、運用中にアクセス権設定を変更すると、アクセス権評価の更新処理により負荷が高まり、エラーや他のアプリ操作への影響が発生する可能性があります。

How：設計時の予防策

- 運用後に頻繁なアプリ更新が発生しないような設計を意識してください。
 - 個人ごとではなく、組織やグループ単位でアクセス権を設定することで、人事異動時もグループのメンテナンスだけで対応でき、個別の権限変更作業を減らせます。

参照ページ

- [kintoneレコード操作の遅延を防ぐアクセス権設定](#)
- [アクセス権の事前評価とアプリの更新時間について](#)

アクセス権のチェックポイント

- アプリのアクセス権で代用できるレコード/フィールドのアクセス権を設定していないか
- アクセス権を設定している「ユーザー/組織/グループ」で不要な設定が含まれていないか
- 運用後にアクセス権の設定変更が頻繁に必要な設計になっていないか

02 一覧画面表示時の絞り込みについて

レコード一覧画面の絞り込み：使用するフィールドの種類

kintoneアプリのレコード一覧画面で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。

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。一方で、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、問題はありません。

What : 何が起きるか

- kintoneでは絞り込みに使うフィールドの種類によって処理速度が異なります。

条件の種類	速度目安	補足
数値・日付	速い	インデックスが効きやすく、効率的に処理できるため
ドロップダウン・ラジオボタン（単一選択系）	速い	値が固定されるため、比較しやすい
チェックボックス、複数選択、ユーザー選択、組織選択、グループ選択、作業者、関連レコード	遅い	複雑なクエリが実行されるため、処理が遅延しやすくなります。処理待ちが増加し、システム全体の負荷が高まる可能性があります。
テーブル内フィールド全て	遅い	

補足

- 遅くなるフィールドを組み合わせることで、より一覧画面の表示が遅くなる可能性があります。
- なお、レコードの一括取得APIでqueryパラメーターを指定する際にも同様の注意が必要になります。

レコード一覧画面の絞り込み：使用するフィールドの種類

How：設計時の予防策

- 複数選択系のフィールドを使用している場合は、下記代用をご検討ください。
 - レコード一覧を表示するときの初期値（デフォルトの一覧）は、**単一選択系フィールドを利用した構成**とする
 - 選択式のフィールドをレコードの絞り込みに利用する場合は、可能なら単一選択系フィールドを利用する。
 - 初期値（デフォルトの一覧）に遅くなるフィールドを利用しない
 - 複数選択系フィールドを利用する場合は、他の単一系フィールドを利用したレコードの絞り込み条件を追加する
※レコード一覧の絞り込み処理が速くなる場合があります
- ユーザー選択、組織選択、グループ選択、作業者については、下記代用をご検討ください。
 - カスタマイズにより、レコードの追加・更新時にユーザー選択フィールドの値を文字列フィールドへ転記し、その文字列フィールドを用いて絞り込みを行う。

レコード一覧画面の絞り込み：絞り込み数の多さ

What : 何が起きるか

- 複雑な絞り込み条件を設定している場合も同様に一覧画面の表示が遅くなる可能性があります。

条件の種類	速度目安	補足
LIKE 検索（〇〇を含む/含まない）	遅い	全文スキャンが走りやすい
OR条件の複合	遅い	条件が増えるほど重くなる

- 複雑な絞り込み条件の利用を制限するものではありませんが、レコード一覧の初期表示（デフォルト一覧）には使用しないようにしてください。
- なお、カスタマイズでqueryパラメーターを指定する際にも同様の注意が必要になります。

How : 設計時の予防策

- データ量が多いフィールドに対してLIKE検索（〇〇を含む/含まない）を極力使用しない
- 複数のOR条件をAND条件でつないだ状態での運用をしない。どうしてもその状態での運用が必要な場合、可能な限りOR条件の数を増やさない条件を検討する
- 条件設定時には、絞り込みの処理が遅くならないかを確認しながら調整する
- 特に5秒以上かかる一覧に関してはアクセス数の多いアプリで適用しない

レコード一覧画面の絞り込み：アクセス数

What : 何が起きるか

- レコード一覧の表示に時間がかかる状況かつ、アプリへのアクセスが多い場合に負荷が高まります。
- 表示が遅い一覧を**ポータルに配置する場合も同様**です。ポータルを開くたびにアプリへのアクセスが発生するため、負荷が高くなる可能性があります。

How : 設計時の予防策

- 複雑な条件を使用した一覧はデフォルト表示ではなく必要時に開く運用にする
- 特に**5秒以上**かかる一覧に関してはアクセス数の多いアプリで適用しない
- 遅い一覧のアプリをポータルに張り付けず、リンク遷移に変更する

参照ページ

- 複数のレコードを取得するAPI実行時の注意点（テーブル内のフィールドについて）
- 複数のレコードを取得するAPI実行時の注意点（OR条件について）
- 快適な絞り込みを設定してみよう！

レコード一覧画面の絞り込み：チェックポイント

絞り込み条件のチェックポイント

- ☑ アクセス数の多いアプリに対し5秒以上の一覧を設定していないか
- ☑ 処理が遅くなるフィールドを条件に適用していないか
(チェックボックス、複数選択、ユーザー選択、組織選択、グループ選択、作業者、関連レコード)
- ☑ テーブルのフィールドを条件に適用していないか
- ☑ データ量の多いフィールドに対しLIKE検索(〇〇を含む/含まない)を使用していないか
- ☑ 複数のOR条件をAND条件でつないだ状態で使用していないか
- ☑ 複雑な条件を使用した一覧をデフォルトで表示していないか
- ☑ 遅い一覧のアプリをポータルに直接貼り付けていないか

03 REST APIを使ったカスタマイズについて

REST APIを使ったカスタマイズ : totalCountパラメーター

REST APIを使ったカスタマイズについても、最適なクエリ条件や実行状況でない場合に遅延やエラーの原因になります。ここでは特に大量データのアプリやアクセス数が多い環境で気を付けるべきポイントについて説明します。

What : 何が起きるか

- 複数のレコードを取得するAPIでは、totalCountパラメーターにtrueを指定すると、falseを指定した場合と比較して処理速度が低下する特性があります。
- これは、該当するレコードの総件数を取得するために、検索条件に一致する全レコードを対象とした集計処理が別途実行されるためです。

How : 設計時の予防策

- 複数のレコードを取得するAPIの絞り込みでヒットしたレコード数を数える必要がない場合は、**totalCountパラメーターにfalseを指定する**
- JavaScriptカスタマイズでREST APIを実行している場合は、ブラウザキャッシュを利用するもしくは、手動取得（ボタン押下）へ変更する
- 複数のレコードを取得するAPIでtotalCountパラメーターにtrueを指定する場合は、下記を推奨する。
 - 複数選択フィールドやユーザー選択フィールドをクエリの絞り込みの条件に使わないこと
 - クエリによる絞り込みにおいては、条件を適切に設定し、ヒットするレコード数を最小限に抑える

REST APIを使ったカスタマイズ：並列実行・チェックリスト

What : 何が起きるか

- 短時間に大量のREST APIを並列実行するとサーバー負荷が増大し、処理時間が長くなるなどパフォーマンスに悪影響が出る場合があります。
- どの程度影響が出るかは、アプリの設定や処理内容、データ量、ネットワーク環境などに大きく左右されます。
- 「kintone REST API」には同時接続数（セッション数）100件の上限があります。これを超えて並列実行すると、HTTPステータスコード429（Too Many Requests）が返ってAPIリクエストが失敗します。

How : 設計時の予防策

- アクセスが集中する業務時間を避け、**業務時間外に実行**する
- 並列にならないよう、**1件ずつ実行**されるように調整する

参照ページ

- 複数のレコードを取得するAPI実行時の注意点（totalCountパラメーターについて）
- kintone REST API同時接続数の制限値はなぜあるのか
- kintone REST API同時接続数を取得する

REST APIを使ったカスタマイズ：並列実行・チェックリスト

カスタマイズのチェックポイント

- ☑ 複数のレコードを取得するAPIでtotalCountパラメーターは必要な時以外falseになっているか
- ☑ 短時間で大量のREST APIを並列実行する設計になっていないか
- ☑ できるだけ業務時間を避け、業務時間外に実行しているか

