**ServiceNow Oktaプラグインデプロイメントガイド**

Oktaオーケストレーションアクティビティパック ー インストールガイド

**Okta Inc.**

301 Brannan Street San Francisco, CA 94107

### 概要

ServiceNowオーケストレーション向けOktaアクティビティパックは、ServiceNow管理者にOktaインフラストラクチャの様々な自動化オプションを提供します。このソリューションに含まれるオーケストレーションワークフローアクティビティを使用して、OktaコントロールとアクティビティをServiceNow内のビジネスワークフローに統合し、ITオペレーションをコントロールできます。OktaコントロールをServiceNowに埋め込むことで、ユーザーは馴染みのあるServiceNowサービスカタログからアクセスおよびアプリケーションをリクエストすることで、ITはServiceNowフレームワーク内から承認、ドキュメント化、および監査コントロールを利用してシームレスかつ自動的に応答できます。公開されているワークフローアクティビティを使用すると、ほとんどまたは全くコーディングを行わずにOkta APIロジックを追加できます。

本書には一般的な顧客シナリオに基づくワークフローの例と説明が含まれていますが、このアクティビティパックの真の能力は、顧客独自のワークフローでオーケストレーションアクティビティを使用し、Oktaとの統合を完全に制御可能にすることです。管理者は、制御ポイント（および今後の拡張）を活用することで、既存および将来のワークフロー内でOktaを活用するために独創的な方法を発掘できます。

このアプリを利用することで、Oktaをサービス指向ITアーキテクチャのアプリケーションエンタイトルメントおよび強制レイヤーにすることができます。ServiceNowはユーザーおよびIT動作の中央制御ハブとして機能します。一方、Oktaはセキュリティを強化するほか、アプリケーションアクセスおよび許可を適用し、ユーザーの生産性を向上します。

### アクティビティ パックを活用したソリューションの例

Oktaアクティビティパックでは、ServiceNow管理者がOktaでグループおよびユーザーを制御できます。OktaマスターグループはOktaでアプリケーションをユーザーに割り当てるプライマリメソッドであり、この層は極めて強力なコントロール機能を提供します。Okta + ServiceNowの共同顧客は、このレベルのコントロールが有用な使用例がいくつもあると述べています。中でも、

* 新入社員のオンボーディング

新入社員の割り当てタスクを取り扱うServiceNowのワークフローが拡張され、Okta経由でアプリケーションプロビジョニングを含めることができるようになりました。営業担当者にとってラップトップのようなITアセット要求、バッジのような設備要求、さらにSalesforceのようなアプリケーションアクセスリクエストがすべて1つのワークフローでバッチ処理されるようになりました。Okta管理者は、Oktaでのグループセットに一致するロールセットを確立し、OktaアクティビティパックがServiceNowのワークフローで利用できるようになりました。

* ジョブ変更

このユースケースは基本的に新入社員のオンボーディングの使用例と同じです。ジョブ変更には通常、新しいアプリケーションセットが求められます。ServiceNowのワークフローはOktaでユーザーグループメンバーシップを操作し、アプリケーションアクセスレベルを変更するよう構築できます。

* セルフサービスのグループメンバーシップ／セルフサービスのアプリリクエスト

OktaのグループメンバーシップはServiceNowのカタログ項目に直接公開できます。ユーザーは特定のプロジェクトまたは許可変更の一環として、グループへメンバーシップを要求できます。オプションとして、

管理者は直接グループメンバーシップアクティビティを使用してユーザーのセルフサービスによるアプリ要求をサポートできます。ここで、カタログ項目は所定のアプリをリクエストするユーザーに公開され、そのリクエストの結果がグループ追加アクティビティ（またはアプリ未割り当てのグループ削除アクティビティ）をキックオフします。ServiceNowでこれらのリクエストとフルフィルメントを追跡することで

監査証跡とコンプライアンス追跡がはるかに容易になり、組織はセルフサービスの業務からメリットを享受できます。

* ユーザーの一時停止

ServiceNowは、ユーザー決定の中央レポジトリとして使用されることがよくあります。ユーザーは高リスクプロファイルの発展途上国に出張中であるかも知れません。ServiceNowのワークフローを利用して、標準Oktaライフサイクル操作を介したユーザーのアプリケーションアクセスを一時停止し、機密の企業資産を保護することができます。また、そのユーザーが通常業務に戻ったら一時停止を解除できます。ServiceNow内で、これらすべてのタスクを1つの場所から実行できることから、企業の管理者にとってリスクとコンプライアンス要件に対する迅速かつ容易なコントロールが可能になります。

あらゆるワークフローで使用されるサポート対象アクティビティで、これらすべてのシナリオが可能となり、管理者はOkta APIメソッドを直接呼び出すスクリプトを構築する必要がなくなります。

### Oktaアクティビティパック機能の説明

アクティビティパック機能内のデータフローを把握しておけば、アーキテクチャに関する決断を行う上で役立ちます。Oktaアクティビティパックはカスタムアクティビティ一式をServiceNowオーケストレーションモジュールに公開します。これらのアクティビティは、ServiceNowとOktaとの間で情報を交換するビルトインデータバスを持ち、

ServiceNowのワークフローからのOktaグループおよびユーザーの操作を許可します。

ワークフローアクティビティとOkta APIの通信を可能にするには、該当する操作をすべてサポートするため、いくつかの依存関係を設定する必要があります。

* ServiceNowのユーザー (sys\_user) とOktaユーザーの一貫したマッピングが必要です。このマッピングはsys\_userテーブルに新しいフィールド (x\_okta2\_actpack\_okta\_id) を追加することで完了します。以下に定義するメソッドを使用してOktaと同期すると、このフィールドに関連するユーザーのOkta IDが格納され、そのユーザーオブジェクトで実行されるすべてのアクティビティで利用できるようになります。デフォルトでは、ServiceNowユーザーからOktaユーザーへのユーザー検索にはメールアドレスの値を使用します。
* Oktaでグループ関連の変更を加えるには、ServiceNowがOkta IDと一緒に利用できるOktaマスターグループを認識する必要があります。

### インストールの前提条件

* + アクティビティパックのインストール前に、ServiceNow Orchestrationプラグインがインストールされている必要があります。
    - https://docs.servicenow.com/bundle/istanbul-it-operations-

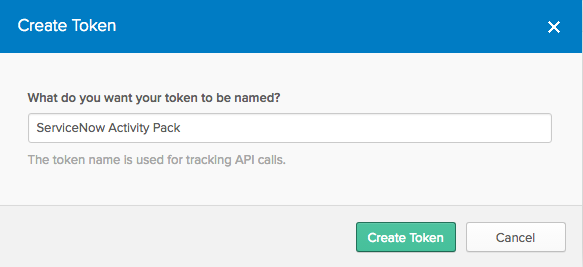
management/page/product/orchestration/task/t\_ActivateOrchestration.html

* + ストアからアプリを取得します。これにより企業インスタンスにインストールすることができます。
    - https://store.servicenow.com/sn\_appstore\_store.do#!/store/application/78d76bbd0f1d2600299f06a ce1050e20
  + 企業がストアからアプリケーションをアクティベートすると、登録済みの企業インスタンスにインストールできるようになります。
  + Oktaアクティビティパックのインストール場所となるServiceNowインスタンスから、［System Applications（システムアプリケーション）］->［Applications（アプリケーション）］モジュールを開きます。［Downloads（ダウンロード）］タブから、Oktaアプリケーションをインストールします。

### 構成

ServiceNowストアからOktaアクティビティパックを取得した後、基本的なセットアップが必要になります。

1. Okta APIトークンの取得
   1. スーパー管理者としてOktaにログインします
   2. 管理者アプリ（メインUIの右上にある［Admin（管理者）］ボタン）に進みます
   3. ［Security-API（セキュリティ-API）］にアクセスします
   4. 新規トークンを生成します
      1. トークンに名前を付けます。名前は参照用です。
      2. ［Create Token（トークンを作成）］をクリックします。



* + 1. トークンをメモ帳（または同様のアプリ）にコピーします。いったん［OK］をクリックすると、このトークンを取得することはできなくなります。トークンを紛失した場合は新しいトークンを要求する必要があります。

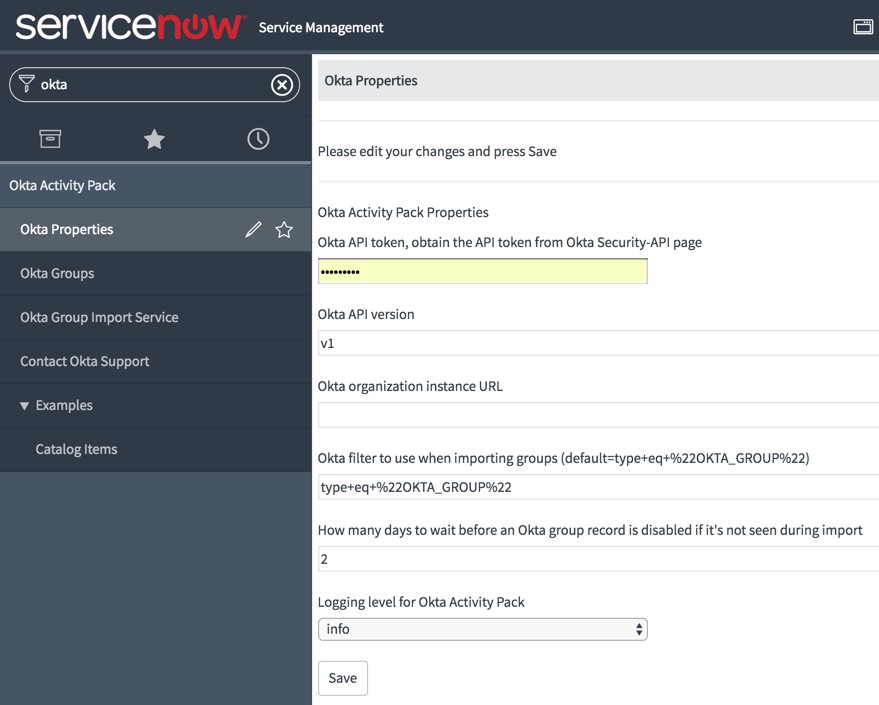
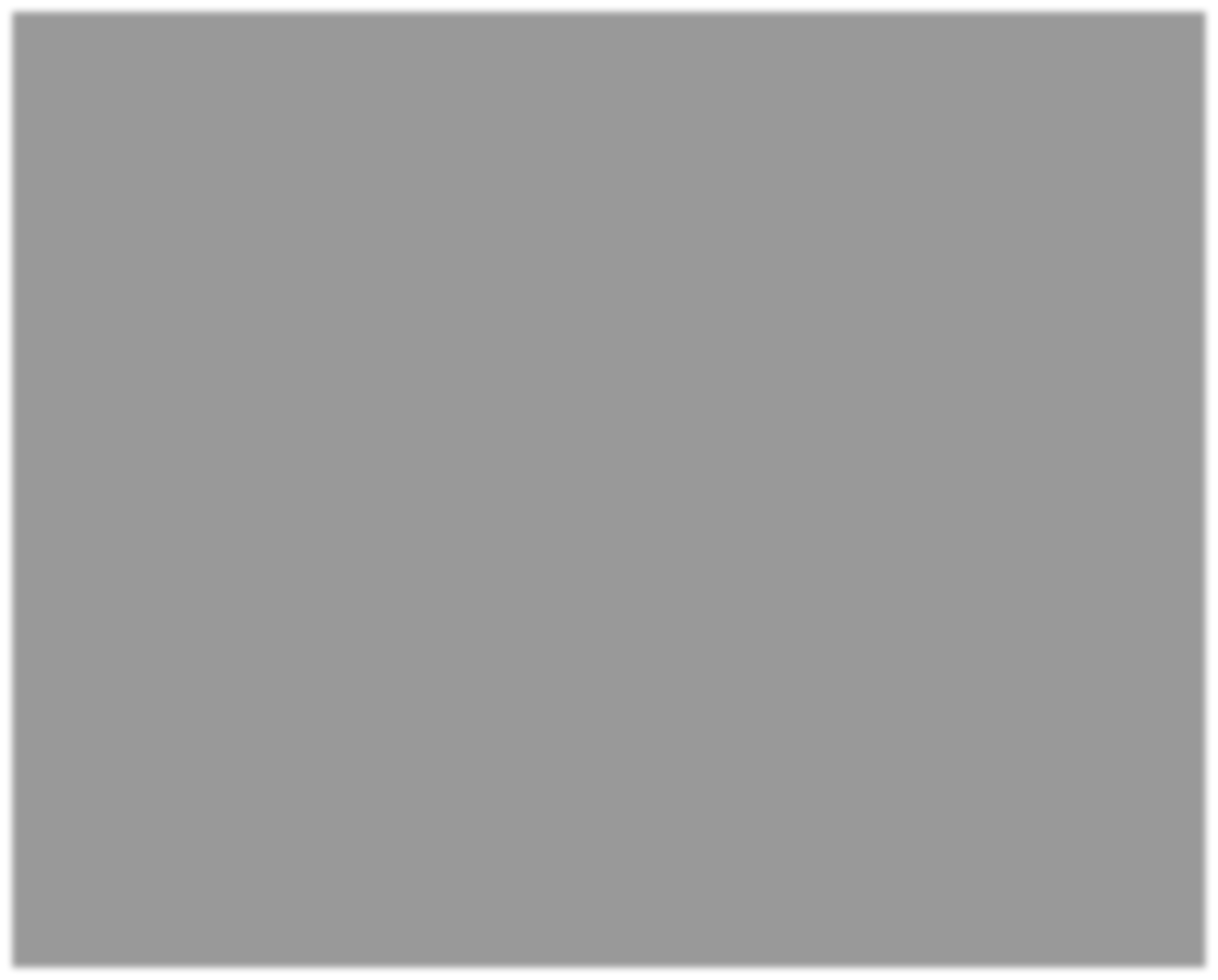
1. 管理者ロールのユーザーとしてログインしているServiceNowから、［Okta Activity Pack（Oktaアクティビティパック）］->［Okta Properties（Oktaプロパティ）］モジュールを開きます
   1. フォームが編集可能でない場合は以下のメッセージが表示されます：

「This record is in the Okta Orchestration Activity Pack application, but Global is the current application.（このレコードはOkta Orchestratonアクティビティパックアプリケーション内にありますが、Globalが現在のアプリケーションです。）To edit this record click here.（このレコードを編集するには、”ここ”をクリックしてください）」 “here（ここ）” のリンクをクリックします。これは、すべてのServiceNow認定アプリケーションの標準的なプラクティスである独自の保護対象アプリケーションスコープ内にあるOktaアプリケーションと関連しています。

1. プロパティフィールドをアップデートします：
   1. Okta APIトークン：Oktaトークン作成からコピーしたトークン値を貼り付けます
   2. Okta APIバージョン：Oktaサポートから別途指示がない限り、デフォルト値“v1”のままにします
   3. Okta organization URL：OktaインスタンスのURL。例： https://company.okta.com
   4. インポートするグループのOktaフィルター ：デフォルトのフィルター値にしておくことを推奨します。このフィルターは

Oktaマスターグループのインポート中に使用されます。値が間違っているとOktaグループアクティビティに関連する機能に直接影響します。

* 1. Oktaグループレコードを無効にするまでの日数：Oktaマスターグループの削除の反映を尊重するために、自動スケジュールされたインポートでこの日数の間グループが表示されない場合、OktaグループのServiceNow表現が無効になります。デフォルト値の”2”を使用することを推奨します。
  2. ロギングレベル：デフォルト値は“info（情報）”であり、最小限のトランザクションイベントデータがServiceNowアプリケーションログに記録されます。詳細なトラブルシューティングには、値を“debug（デバッグ）”に設定できます。このようにすると、すべてのOktaアクティビティパックのトランザクションで大量のアクティビティログが発生することに留意してください。実際のインスタンスでは慎重に使用し、トラブルシューティングが完了したら直ちに値を“info（情報）”に戻してください。



### データの依存関係

前述したように、アクティビティパックはユーザーおよびグループの両方のOkta IDデータに依存します。ServiceNowグループレコード (sys\_user\_group) の使用との競合を避けるため、アクティビティパックは新規テーブルを作成し、プロパティページで定義されたフィルターの条件を満たすOktaマスターグループに関するデータを格納します。

以下に、ServiceNowにインポートされ格納されるこれらのOktaメタデータの詳細を定義します。

#### Oktaマスターグループ

必要なOktaマスターグループメタデータを保存するため、新しい表 (x\_okta2\_actpack\_okta\_group) がプロビジョニングされています。この表は、［Okta Activity Pack（Oktaアクティビティパック）］->［Okta Groups（Oktaグループ）］モジュールからアクセスできます。この表は

Oktaをマスターとするグループのみをリスト化し、Active Directoryまたはその他のソース（O365あるいはBoxなど）をマスターとするグループは

リスト化しません。OktaでOktaマスターグループがServiceNowからアプリケーションまたはポリシー割り当てを実行することを前提としています。

この表の自動入力には以下のコンポーネントが使用されます：

* Okta Import Groups Scheduled Job – デフォルトでは、インスタンスのタイムゾーンで毎日午前6時にジョブが実行されます。［System Definitions（システム定義）］->［Scheduled Jobs（スケジュールされたジョブ）］モジュールに進み、“Okta Import Groups”というジョブ名を検索することで頻度を変更できます。また、［Execute Now（今すぐ実行）］フォームボタンを使用することでいつでもジョブをジョブフォームから手動で実行できます。
* Okta Group Importサービスステージングテーブル (x\_okta2\_actpack\_okta\_group\_import) – スケジュールされたジョブが、変換マップで処理されるようインポートセットのステージングテーブルに自動入力
* Okta Group Import変換マップ – ターゲットOktaグループレコード (x\_okta2\_actpack\_okta\_group) の作成およびアップデートを管理する標準ServiceNowインポートセットの変換マップ

#### ユーザー属性

sys\_userテーブルには、ユーザーのOkta IDを格納するために使用する新規フィールド (x\_okta2\_actpack\_okta\_id) があります。このIDはユーザーがOktaアクティビティで使用する前に自動入力済でなければなりません。サンプルカタログ項目では、Okta ID値を持たないユーザーはフィルターで除外され、カタログ変数で選択することはできません。新規

カタログ項目を構築するか、他の方法でOktaアクティビティを利用する場合、“x\_okta2\_actpack\_okta\_id”値を持つユーザーのみ使用するようにしてください。

### Oktaから必要なデータを自動入力

#### 最初のOktaマスターグループインポート

スケジュールされたジョブ “Okta Import Groups”は、毎日午前6時に実行されるようスケジュールされています。いったんアプリケーションプロパティが設定されたら、このジョブを手動で実行する必要があります。

* + ［System Definition（システム定義）］->［Scheduled Jobs（スケジュールされたジョブ）］モジュールリストを開きます。
  + Okta Import Groupsレコードを検索します。
  + ［Execute Now（今すぐ実行）］をクリックしてグループのインポートを開始します。
  + 数分後、Okta Groupsモジュールを開き、グループレコードが作成されているか確認します。

#### Okta IDをServiceNowユーザーアカウントに追加

デフォルトでは、sys\_userテーブルに“Get Okta ID”ビジネスルールがあり、Eメール値を持っているが、まだOkta IDフィールド値が含まれていない状態でユーザーレコードがアップデートされると、Oktaはクエリを実行し、ユーザーのIDを見つけようとします。このフィールドはユーザーテーブルリストやフォームビューには表示されません。値を確認したい場合はこのフィールドを手動で追加できます。

Oktaで関連付けられたユーザーを見つけるクエリは、[profile.email eq *‘servicenow user email’*]です。このルールは変更しないようにしてください。お使いのデータモデルに適合しない場合、このルールを無効にし、独自のルールを作成してOktaでクエリを実行することができます。

このルールはユーザーレコードが挿入されたりアップデートされた場合のみに実行されるため、一括更新オペレーションを実行することから始める必要があるかも知れません。スケジュールされたジョブ“Okta Bulk User ID Update”を実行して、

Eメール値を持つがServiceNowでOkta ID値を持たないすべてのユーザーをクエリできます。このスケジュール済みジョブは、実行をスケジュールされておらず、必要に応じてオンデマンド実行で使用されるべきです。

**警告：APIは一度に600ユーザークエリまでに制限されています。アップデートが必要なユーザーが500人以上いる場合、Oktaサポートに連絡して別のスクリプトを要請してください。**

### 使用状況

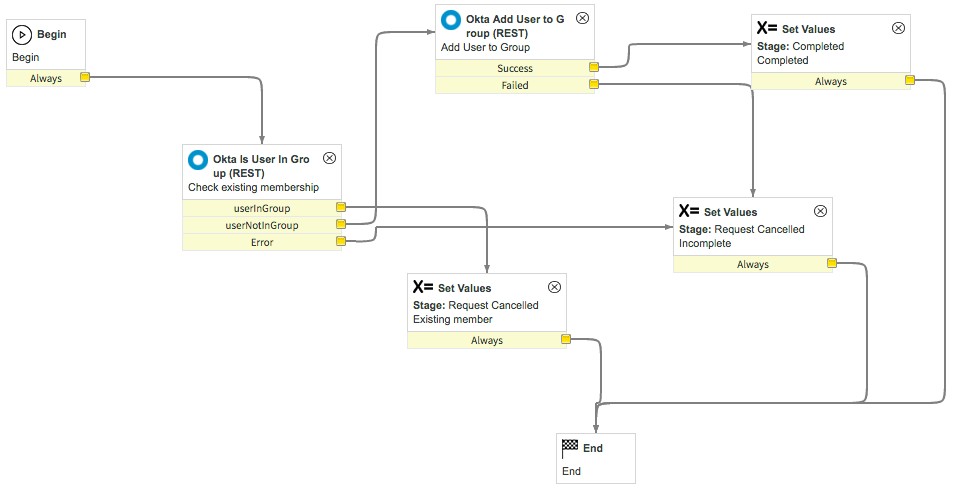
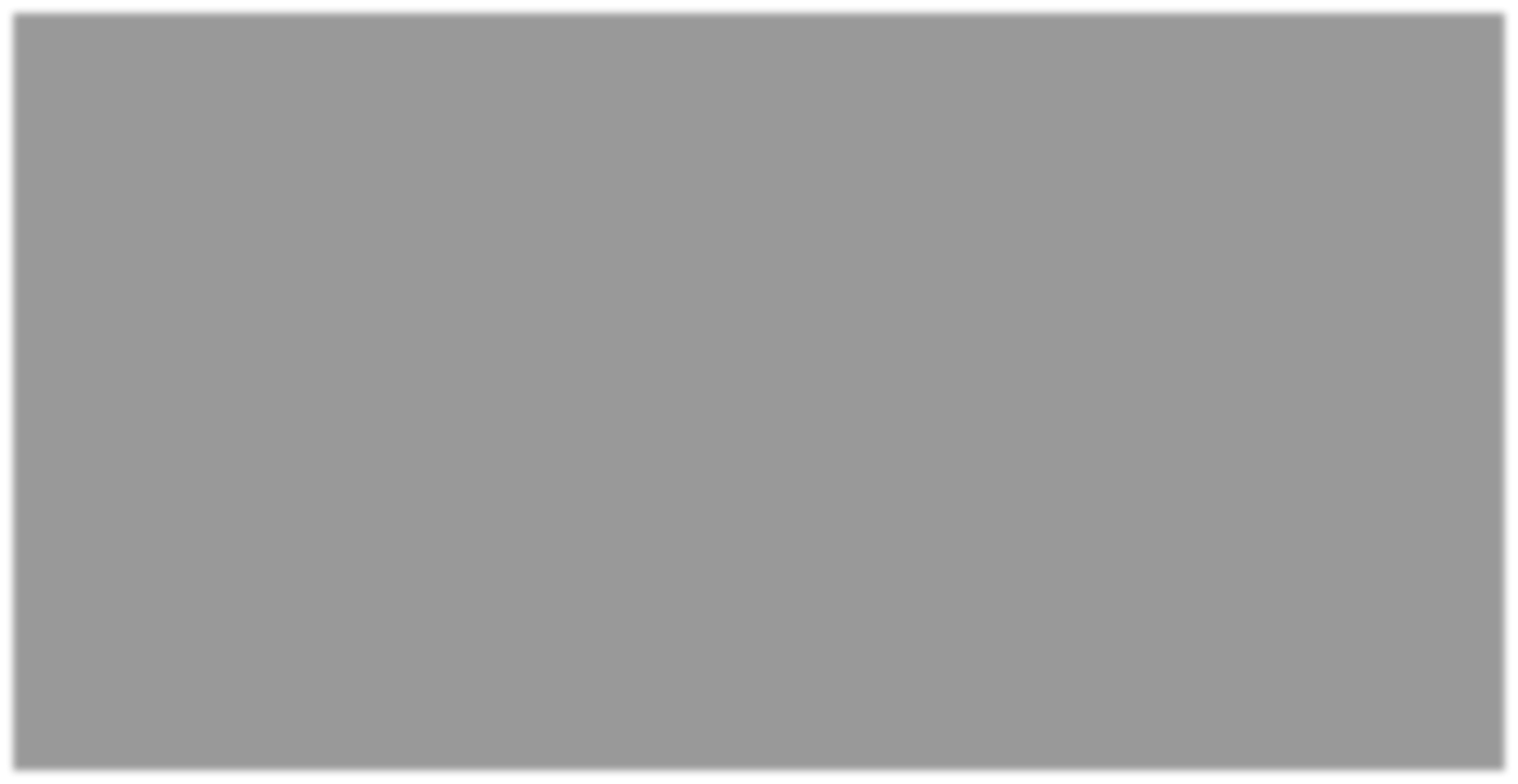
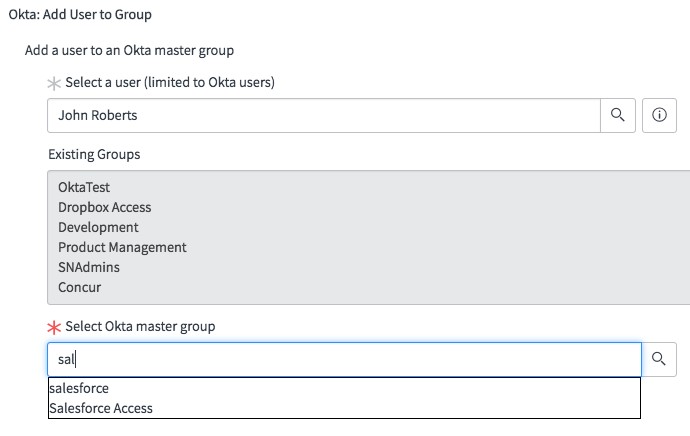
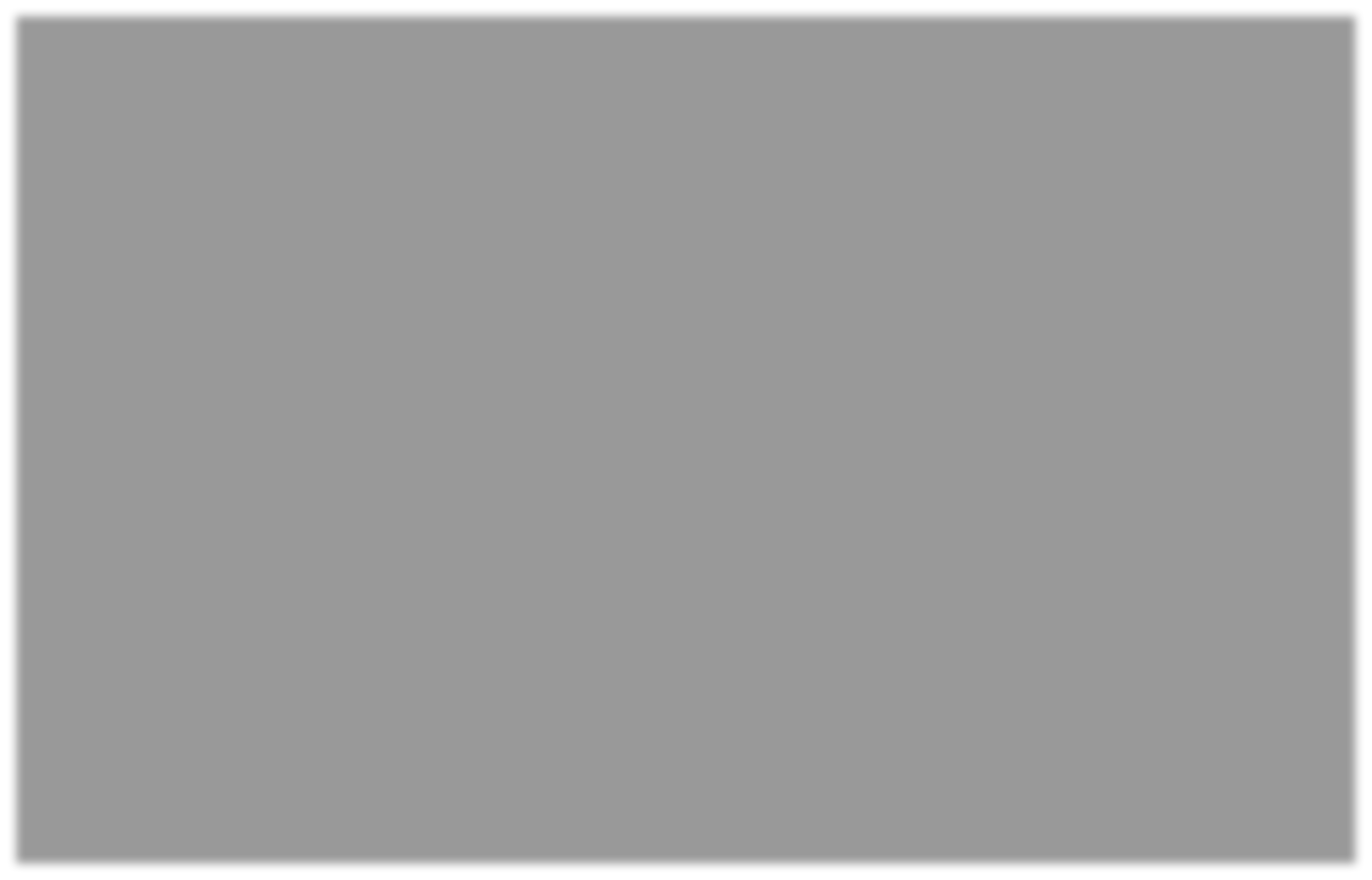
#### 要件

* + 不随のサンプルカタログ項目とオーケストレーションアクティビティでは、ServiceNowユーザー (sys\_user) およびOktaグループ (x\_okta2\_actpack\_okta\_group) が、一致するOktaレコードから得たIDで自動入力されたOkta IDフィールドを持つ必要があります。
  + ServiceNowで値を表示／確認する場合、Okta IDをユーザー (sys\_user) フォームとリストに追加する必要があります。このフィールドはデフォルトでは読み取り専用です。
  + ServiceNowでユーザー作成またはアップデートの操作を行うと、OktaにServiceNowユーザーレコードでEメールの一致を使用してOktaを見つけるクエリをトリガーします。オーケストレーションアクティビティで管理できるのはOkta IDを持つユーザーのみです。
    - このクエリが成功するためにはメールアドレスが一致する必要があります。Oktaユーザー名がメールアドレスの形式でない場合もあります。この場合、専門サービスによるアシスタンスが必要になるため、Oktaに連絡して支援を求めてください。
    - 一部のServiceNowでのオペレーションは、sys\_userテーブルの各レコードでアップデートイベントをトリガします。その場合、Oktaへの大量呼び出しが発生する可能性があります。個々のソースから一括で.CSVまたはLDAPアップデートを行うよりは、Oktaを使用してユーザーのプロビジョニングを行うことを推奨します。

### サンプルカタログ項目およびワークフローの使用

アクティビティパックには、Oktaアクティビティを使用して利用可能な機能を実行できるよう以下のサンプルカタログ項目およびワークフローが含まれています。これらのサンプルを現状のまま使用したり、変更したり、または独自のワークフローでOktaアクティビティを使用して独自のサンプルを構築できます。以下は各ワークフローの詳細なレビューです。

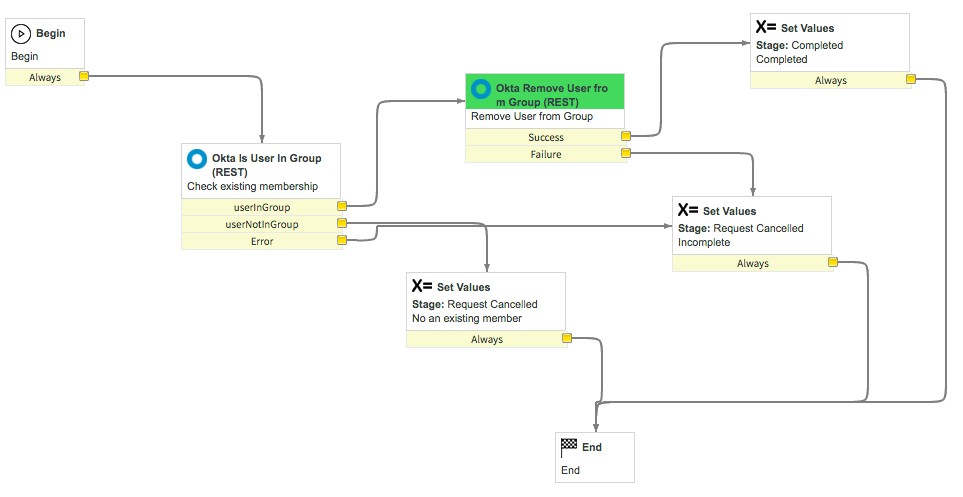
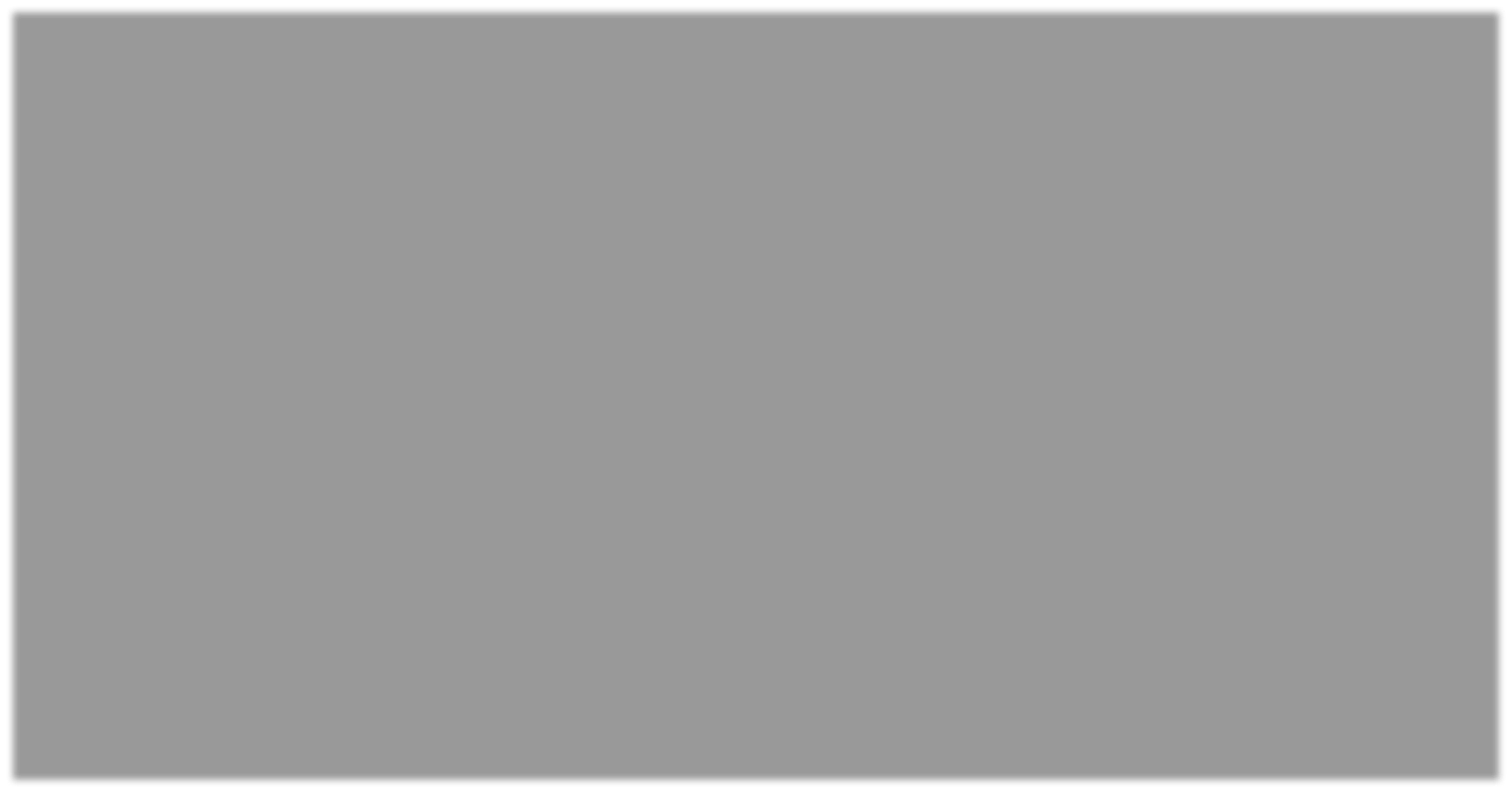
* + **Okta：Add User to Group（ユーザーグループの追加）**- Oktaユーザー (ServiceNowユーザーにリンク) をOktaマスターグループに追加するリクエストおよびオーケストレーション



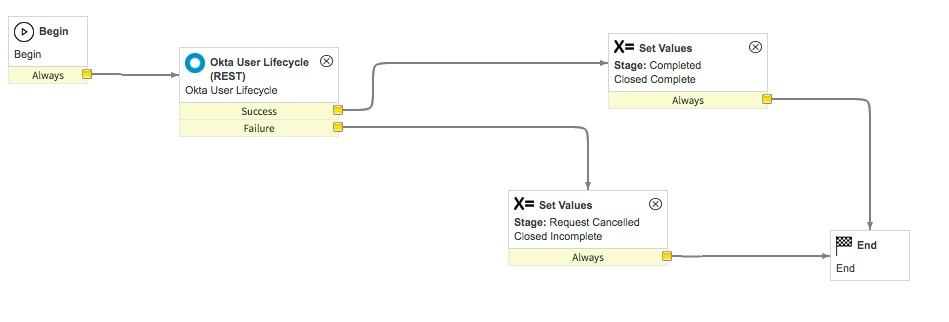
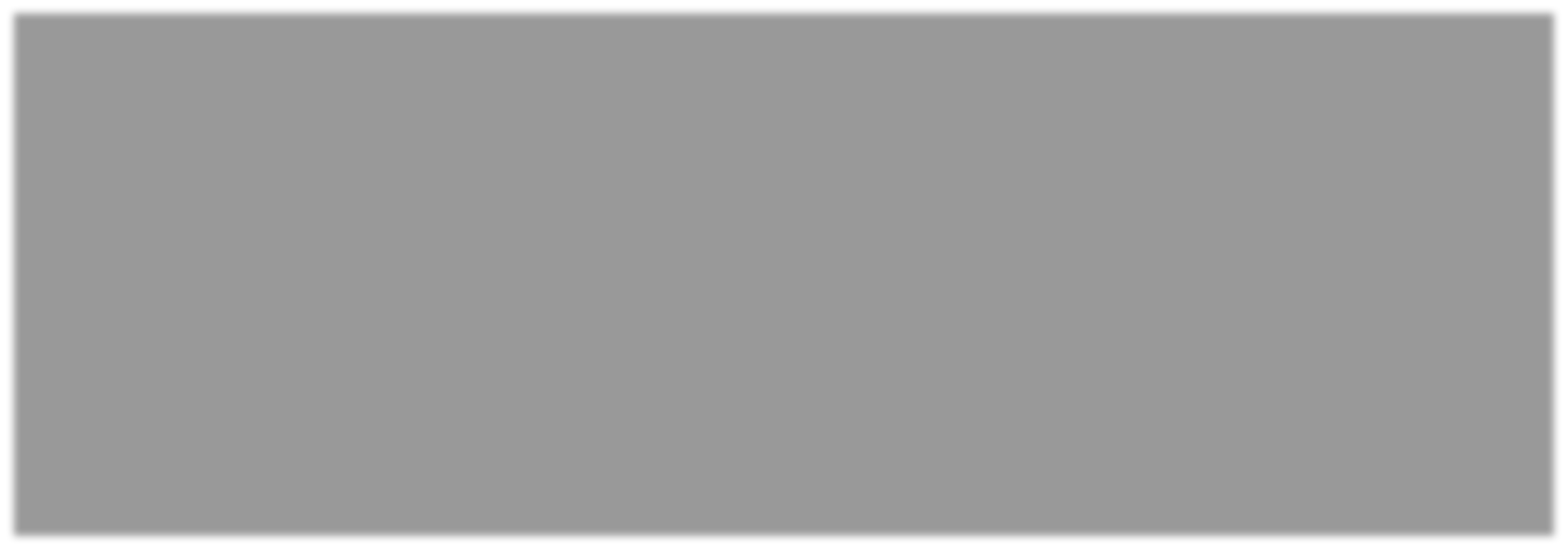
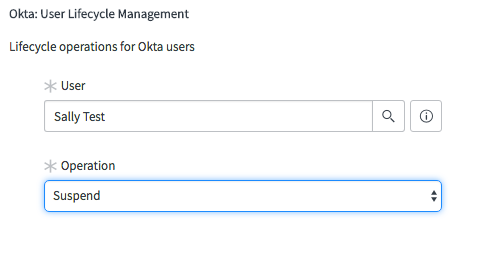
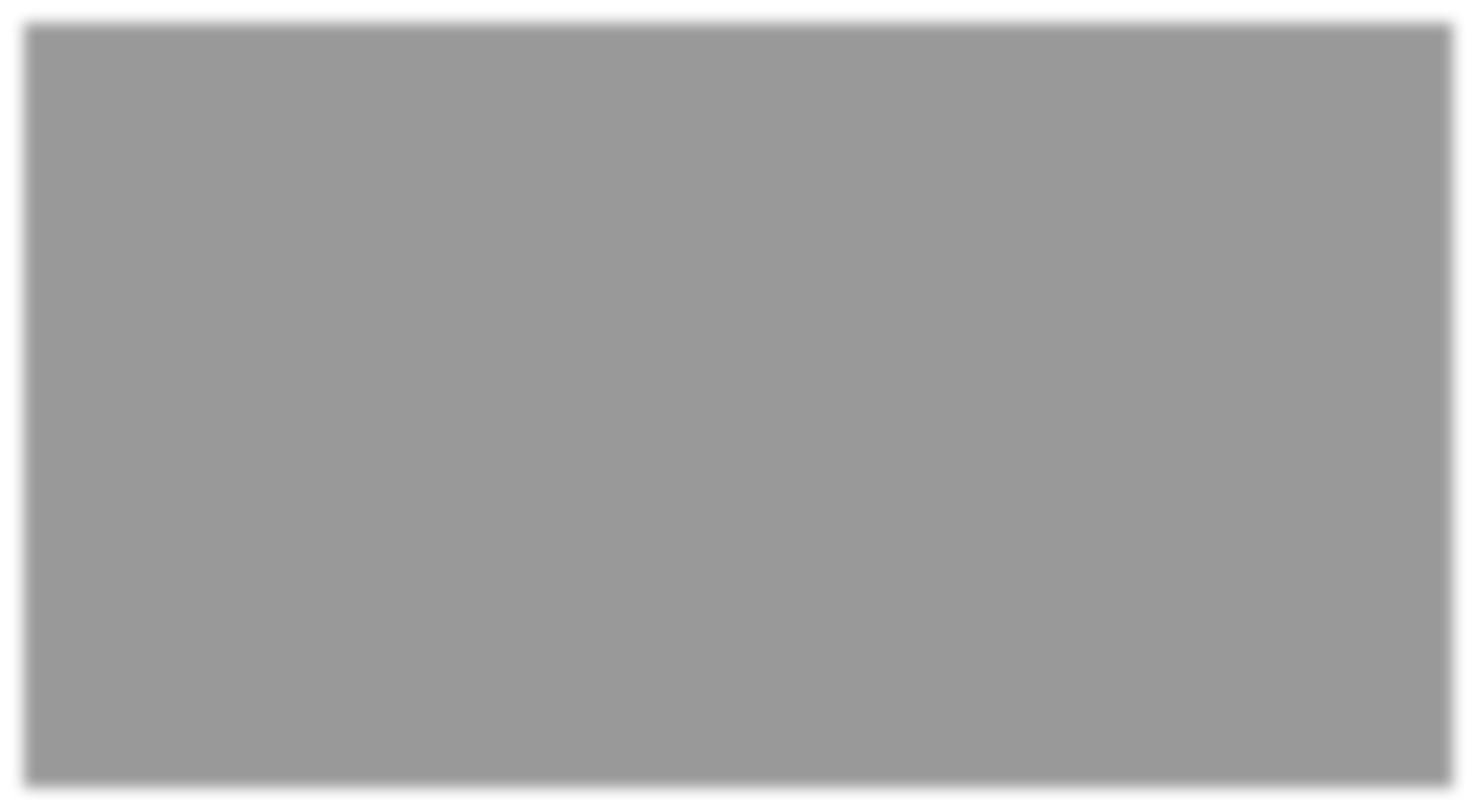
* + - 変数
      * user – Okta IDを持つユーザーに制限されたsys\_userを参照、デフォルトは現在のセッションユーザー
      * group – アクティブなグループに制限されたx\_okta2\_actpack\_okta\_groupを参照
      * existing\_groupsおよびexisting\_group\_ids – ユーザーのグループリストを追跡および表示
    - クライアントスクリプト
      * Get Okta User's Groups – ユーザー変数の変更時に、クライアントスクリプトがOktaにそのユーザーの既存グループの

リストを検索するクエリを実行します。ページにグループ名が表示され、既存グループが ‘group’ 変数の選択肢から削除されます。

* + - ワークフロー：Okta Add User to Group（Oktaがユーザーをグループに追加）
      * アクティビティ：Okta Is User in Group（Oktaがグループのユーザー）– OktaグループおよびユーザーのOkta IDを取得し、Oktaで既存のグループメンバーシップを確認します。
      * アクティビティ：Add user to Group（ユーザーをグループに追加）– OktaグループおよびユーザーのOkta IDを取得し、ユーザーをOktaグループに追加します。
  + **Okta：Remove User from Group**（ユーザーをグループから削除）- Oktaユーザー (ServiceNowユーザーにリンク) をOktaマスターグループから削除するリクエストとオーケストレーション



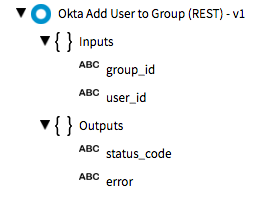
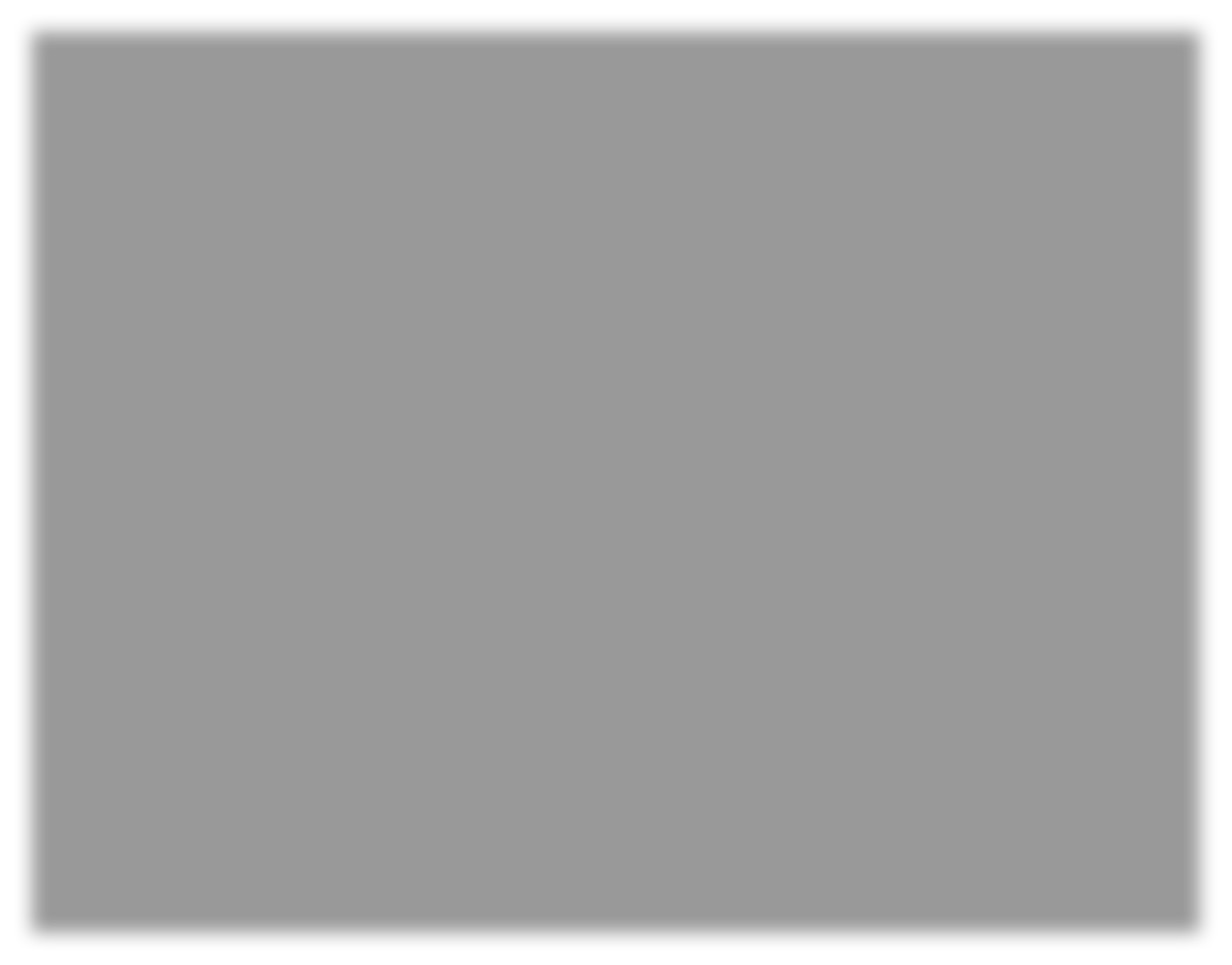
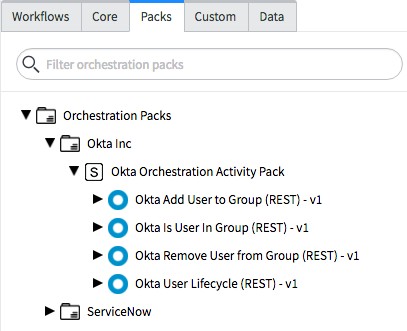
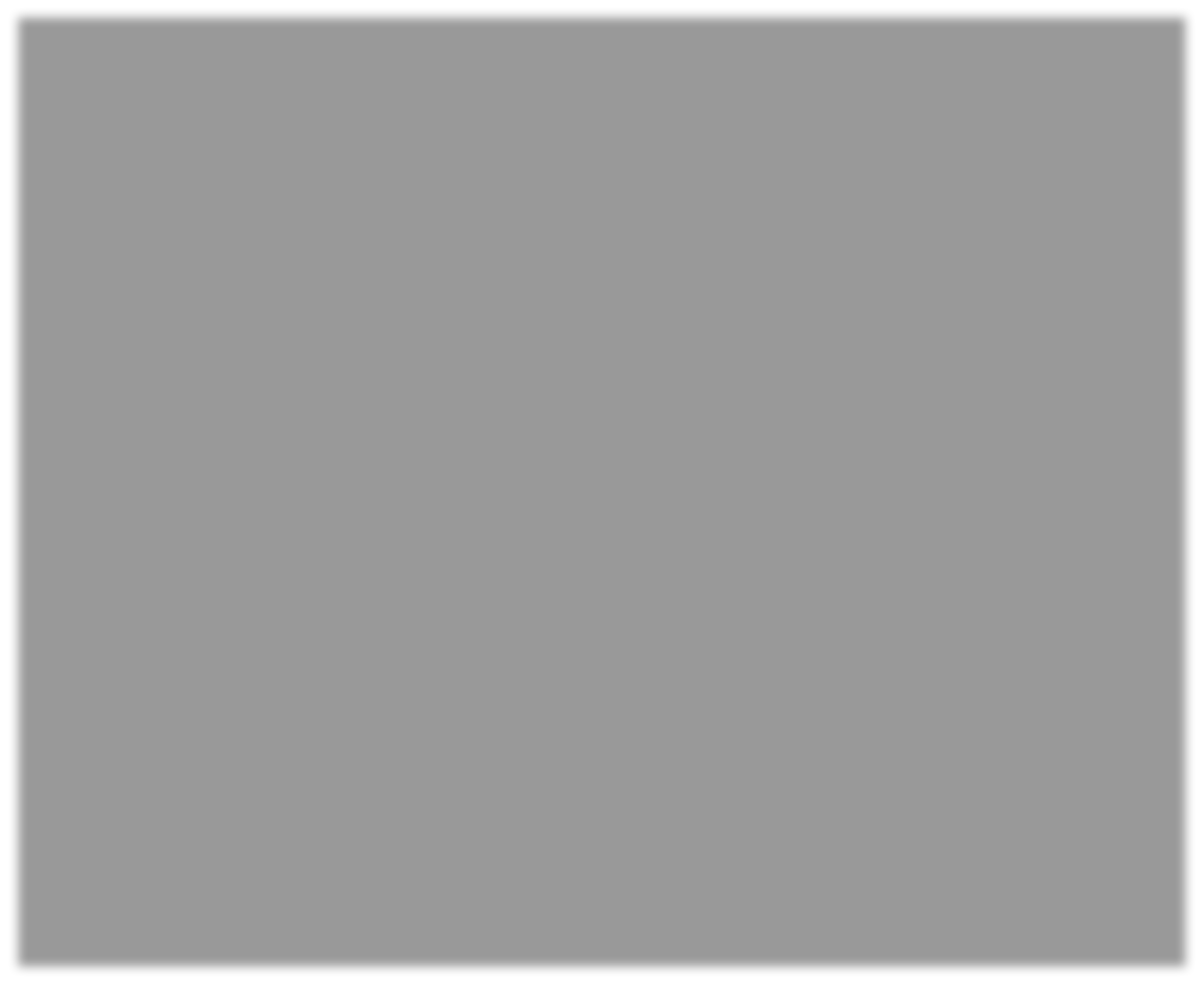
* + - 変数
      * user – Okta IDを持つユーザーに制限されたsys\_userを参照、デフォルトは現在のセッションユーザー
      * group – アクティブなグループに制限されたx\_okta2\_actpack\_okta\_groupを参照、既存のグループIDリスト
      * existing\_groupsおよびexisting\_group\_ids – ユーザーのグループリストを追跡および表示
    - クライアントスクリプト
      * Get Okta User's Groups（Oktaユーザーのグループを取得）– ユーザー変数の変更時に、クライアントスクリプトがOktaにそのユーザーの既存グループのリストを検索するクエリを実行します。ページにグループ名が表示され、既存グループが ‘group’ 変数の選択肢から削除されます。
    - ワークフロー：Okta Add User to Group（Oktaがユーザーをグループに追加）
      * アクティビティ：Okta Is User in Group（Oktaがグループのユーザー）– OktaグループおよびユーザーのOkta IDを取得し、Oktaで既存のグループメンバーシップを確認します。
      * アクティビティ：Remove User from Group（ユーザーをグループから削除）– OktaグループおよびユーザーのOkta IDを取得し、ユーザーをOktaグループから削除します。
  + Okta：User Lifecycle（ユーザーライフサイクル）- **Oktaユーザー (ServiceNowユーザーとリンク) でOktaライフサイクルアクションを実行するリクエストおよびオーケストレーション**



* + - 変数
      * user – Okta IDを持つユーザーに制限されたsys\_userを参照、デフォルトは現在のセッションユーザー
      * operation – Oktaユーザーのライフサイクルオペレーションをサポートします（詳細はOkta APIドキュメントを参照）。一般的なオペレーションはアクティベート、非アクティブ化、一時停止、停止解除、ロック解除、パスワードのリセットです。
    - クライアントスクリプト
      * なし
    - ワークフロー： Okta User Lifecycle（Oktaユーザーライフサイクル）
      * アクティビティ：Okta User Lifecycle（Oktaユーザーライフサイクル）– OktaユーザーIDおよびオペレーション名を取得し、そのOktaユーザーでオペレーションの実行を試みます。

### 利用可能なOktaアクティビティ

任意のワークフローから、［Packs（パック）］タブの下で利用可能なOktaアクティビティにアクセスできます。各アクティビティで公開された入出力を見ることができます。



#### アクティビティの詳細

*すべてのOktaアクティビティは読み取り専用モードに設定されています。コア機能を変更する必要がある場合、新規カスタムアクティビティを作成し、Okta提供のアクティビティから特定の設定とスクリプトを手動でコピーできます。*

出力された値はワークフローデータバス値を使用して他のアクティビティで利用できます。詳細については、ServiceNowワークフローのドキュメントを参照してください。

* + **Okta Add User to Group（Oktaがユーザーをグループに追加）**-OktaユーザーをOktaマスターグループに追加します
    - 関連するOkta API： https://developer.okta.com/docs/api/resources/groups.html#add-user-to-group
    - （入力）group\_id：OktaグループID。インポートされたOktaグループ、okta\_idフィールドから取得できます
    - （入力）user\_id：OktaユーザーID。sys\_user.x\_okta2\_actpack\_okta\_idフィールドから取得できます
    - （出力）status\_code：Okta APIリクエストステータスコード
    - （出力）error：Okta APIリクエストからのエラー
    - （条件）Success：status\_code == 204
    - （条件）Failed：（else条件）
  + **Okta Is User In Group（Oktaがグループのメンバー）**-OktaユーザーがOktaグループのメンバーであるか確認
    - 関連するOkta API： https://developer.okta.com/docs/api/resources/groups.html#list-group-members
    - （入力）group\_id：OktaグループID。インポートされたOktaグループ、okta\_idフィールドから取得できます
    - （入力）user\_id：OktaユーザーID。sys\_user.x\_okta2\_actpack\_okta\_idフィールドから取得できます
    - （出力）userInGroup：trueまたはfalse。ユーザーがグループに存在するかどうか
    - （出力）status\_code：Okta APIリクエストステータスコード
    - （出力）error：Okta APIリクエストからのエラー
    - （条件）userInGroup：userInGroup == “true”
    - （条件）userNotInGroup：userInGroup == “false”
    - （条件）Error：（else条件）
  + **Okta Remove User from Group（Oktaがグループからユーザーを削除）**-OktaマスターグループからOktaユーザーを削除
    - 関連するOkta API：https://developer.okta.com/docs/api/resources/groups.html#remove-user-from-group
    - （入力）group\_id：OktaグループID。インポートされたOktaグループ、okta\_idフィールドから取得できます
    - （入力）user\_id：OktaユーザーID。sys\_user.x\_okta2\_actpack\_okta\_idフィールドから取得できます
    - （出力）status\_code：Okta APIリクエストステータスコード
    - （出力）error：Okta APIリクエストからのエラー
    - （条件）Success：status\_code == 204
    - （条件）Failed：（else条件）
  + **Okta User Lifecycle（Oktaユーザーライフサイクル））**- Oktaユーザーでユーザーのライフサイクルアクションを実行
    - 関連するOkta API： https://developer.okta.com/docs/api/resources/users.html#lifecycle-operations
    - （入力）user\_id：OktaユーザーID。sys\_user.x\_okta2\_actpack\_okta\_idフィールドから取得できます
    - （入力）operation: ライフサイクルのオペレーション（Okta APIを参照）
    - （入力）send\_email：（任意）アクションがOktaからユーザーにEメールを送信するかどうかをtrue/falseで指示
    - （入力）query\_parm：（任意）クエリパラメーター（Okta APIを参照）
    - （出力）response：APIリクエストの応答文
    - （出力）status\_code：Okta APIリクエストステータスコード
    - （出力）error：Okta APIリクエストからのエラー
    - （条件）Success：status\_code = 200
    - （条件）Failed：成功しない

### トラブルシューティング

#### API、URL、トークン、およびアクセス検証用の簡単なテスト

* + Scripts（スクリプト）– Background（バックグラウンド）モジュールから以下のスクリプトを実行 var okta = new x\_okta2\_actpack.OktaRESTActivity();

var rest = okta.oktaRESTRequest("get", "users");

* + “oktaRESTRequest response status:200” を確認するログステートメントが取得されます。これはAPIテストが成功したことを意味します。ステータスが200でない場合、出力を確認して追加の情報を探します。
  + APIトークンに問題がある場合、以下のような応答が返されます。
    - **response status:401**, body:{"errorCode":"E0000011","errorSummary":"**Invalid token provided**"

#### 詳細なデバッグロギング

* + Oktaプロパティページから、ロギングレベルを‘debug（デバッグ）’に設定します。
  + ［System Logs（システムログ）］->［Application Logs（アプリケーションログ）］を確認してデバッグ詳細をリスト表示します。
  + ［System Logs（システムログ）］->［All（すべて）］を確認してデバッグ詳細をリスト表示します。

### アプリケーション コンポ―ネント リスト – 補足

この情報はServiceNowアーキテクト向けの詳細リファレンスガイドとして提供されています。通常の使用では、変更または構成手順は必要ありません。

# データモデル

## 新規テーブル

##### Okta Group Import (x\_okta2\_actpack\_okta\_group\_import)

* + 拡張元：sys\_import\_set\_row
  + 拡張可能：いいえ
  + アプリケーションへのアクセス
    - アクセス元：
    - Webサービス経由：はい
    - 構成を許可：いいえ
    - オペレーション：読み取り
  + Columns（列）
    - Description（説明）- 文字列
    - ID (id) - 文字列
    - Name（名前）- 文字列
    - Type（タイプ）- 文字列
  + ACLカウント
* 読み取り 書き込み 作成 削除

|  |  |  |  |
| --- | --- | --- | --- |
| 行 1 | 1 | 1 | 1 |
| 列1 | 1 | 1 | N/A |

##### Okta Group (x\_okta2\_actpack\_okta\_group)

* + 拡張元：N/A
  + 拡張可能：いいえ
  + アプリケーションへのアクセス
    - アクセス元：すべてのアプリケーションスコープ
    - Webサービス経由：はい
    - 構成を許可：いいえ
    - オペレーション：読み取り
  + Columns（列）
    - Active（アクティブ）- True/False
    - Description（説明）- 文字列
    - Last import (last\_import) - 日付／時刻
    - Name（名前）- 文字列
    - Okta ID (okta\_id) - 文字列
    - Type（タイプ）- 文字列
  + ACLカウント
* 読み取り 書き込み 作成 削除 行 1 1 1 1

列1 1 1 n/a

## 新規列

Okta ID (sys\_user.x\_okta2\_actpack\_okta\_id) - 文字列

* + ACLカウント

読み取り 書き込み 作成

1 1 0

# サーバー開発

## ビジネスルール

* + Get Okta ID sys\_user (async: insert,update)

## スクリプトに含まれる項目

* + OktaUtils
    - 説明：var otkautils = new x\_okta2\_actpack.Okta\_Utils(); otkautils.importGroups();
    - アクセス元：すべてのアプリケーションスコープ
  + OktaAJAX
    - 説明：AJAX関数からOkta関連のクライアントスクリプトへ
    - アクセス元：すべてのアプリケーションスコープ
    - クライアント呼び出し可能スクリプト：ACLを保有 = Yes
  + OktaRESTActivity
    - 説明：スクリプトまたはオーケストレーションアクティビティで使用するOkta RESTラッパー
    - アクセス元：すべてのアプリケーションスコープ

## スケジュールされたジョブ

* + Okta Import Groups
    - 毎日 - 06:00:00
  + Okta Bulk User ID Update
    - オンデマンド - 00:00:00

# アクセス制御

## ロール

* + x\_okta2\_actpack.admin: Oktaアクティビティパック管理者ロール
  + x\_okta2\_actpack.user: Oktaアクティビティパックのコンポーネントへのユーザーレベルのアクセス権

## アクセス制御

* + OktaAJAX client\_callable\_script\_include:execute (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin,x\_okta2\_act pack.user)
  + sys\_user.x\_okta2\_actpack\_okta\_id record:read (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + sys\_user.x\_okta2\_actpack\_okta\_id record:write (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group.\* record:create (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group record:delete (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group record:read (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.user)
  + x\_okta2\_actpack\_okta\_group record:write (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group.\* record:create (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group.\* record:read (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.user)
  + x\_okta2\_actpack\_okta\_group.\* record:write (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import record:create (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import record:delete (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import record:read (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import record:write (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import.\* record:create (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import.\* record:read (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)
  + x\_okta2\_actpack\_okta\_group\_import.\* record:write (条件: No, スクリプト: No, ロール: x\_okta2\_actpack.admin)

# プロパティ

* + x\_okta2\_actpack.api\_token=: Okta APIトークン。Okta Security-APIページからAPIトークンを取得 (password2)
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin
  + x\_okta2\_actpack.api\_version=v1: Okta APIバージョン（文字列）
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin
  + x\_okta2\_actpack.deactivate\_old\_groups\_days=2: インポート中に見つからなかったOktaグループのレコード無効化までに待機する日数（整数）
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin
  + x\_okta2\_actpack.group\_import\_filter=type+eq+%22OKTA\_GROUP%22: グループのインポート中に使用するOktaフィルター（default=type+eq+%22OKTA\_GROUP%22）（文字列）
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin
  + x\_okta2\_actpack.logging.verbosity=info: Oktaアクティビティパックのロギングレベル (選択肢リスト)
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin
  + x\_okta2\_actpack.org\_instance\_url=: Okta organizationインスタンスURL (文字列)
    - 読み取り：x\_okta2\_actpack.admin
    - 書き込み：x\_okta2\_actpack.admin

# ナビゲーション

## メニューおよびモジュール（デスクトップ）

* + Oktaアクティビティパック / Oktaプロパティ: URL (引数から:):
  + Oktaアクティビティパック / Oktaグループ: レコードのリスト: x\_okta2\_actpack\_okta\_group
  + Oktaアクティビティパック / Oktaグループインポートサービス: URL (引数から:):
  + Oktaアクティビティパック / Oktaサポートに問い合わせ: URL (引数から:):
  + Oktaアクティビティパック / 例: 区切り
  + Oktaアクティビティパック / カタログ項目: レコードのリスト: sc\_cat\_item

# サービスカタログ

## カタログクライアントスクリプト

* + Get Okta User's Groups (onChange)
    - 説明：
    - 適用先：項目：Okta: Remove User from Group
  + Get Okta User's Groups (onChange)
    - 説明：
    - 適用先：項目：Okta: Add User to Group

# 統合

## インポート変換

* + Okta Group Import
    - ソース：x\_okta2\_actpack\_okta\_group\_import (x\_okta2\_actpack\_okta\_group\_import)
    - ターゲット：x\_okta2\_actpack\_okta\_group (x\_okta2\_actpack\_okta\_group)

## RESTメッセージ

* + Okta Get Groups
    - 説明：Oktaグループを取得するRESTメッセージの例
    - アクセス元：すべてのアプリケーションスコープ］
    - エンドポイント：
    - メソッド：get

# ワークフロー

## アクティビティ

* + Okta Remove User from Group (REST) 実行スクリプト
    - 説明：Oktaグループからユーザーを削除するアクティビティ
    - アクセス元：すべてのアプリケーションスコープ
    - チェックアウト有：false
  + Okta User Lifecycle (REST) 実行スクリプト
    - 説明：Oktaユーザーのライフサイクル管理オペレーション
    - アクセス元：すべてのアプリケーションスコープ
    - チェックアウト有：false
  + Okta Add User to Group (REST) 実行スクリプト
    - 説明：Oktaグループにユーザーを追加するアクティビティ
    - アクセス元：すべてのアプリケーションスコープ
    - チェックアウト有：false
  + Okta Is User In Group (REST) 実行スクリプト
    - 説明：ユーザーがグループに存在するか検証
    - アクセス元：すべてのアプリケーションスコープ
    - チェックアウト有：false

## ワークフロー

* + Okta Remove User from Group ()
    - 公開済み：true
    - チェックアウト有：false
  + Okta User Lifecycle (sc\_req\_item)
    - 公開済み：true
    - チェックアウト有：false
  + Okta Add User to Group (sc\_req\_item)
    - 公開済み：true
    - チェックアウト有：false

# その他のアプリファイルレコード

* + カタログUIポリシー：既存のグループフィールドは読み取り専用 (catalog\_ui\_policy)
  + カタログUIポリシー：既存のグループフィールドは読み取り専用 (catalog\_ui\_policy)
  + カタログUIポリシーアクション：existing\_groups (catalog\_ui\_policy\_action)
  + カタログUIポリシーアクション：existing\_groups (catalog\_ui\_policy\_action)
  + カタログUIポリシーアクション： existing\_group\_ids (catalog\_ui\_policy\_action)
  + カタログUIポリシーアクション： existing\_group\_ids (catalog\_ui\_policy\_action)
  + イメージ：okta\_logo\_sm.png (db\_image)
  + 変数：既存のグループID (item\_option\_new)
  + 変数：既存のグループID (item\_option\_new)
  + 変数：既存のグループ (item\_option\_new)
  + 変数：既存のグループ (item\_option\_new)
  + 変数：オペレーション (item\_option\_new)
  + 変数：ユーザーを選択 (Oktaユーザーに限定) (item\_option\_new)
  + 変数：ユーザーを選択 (Oktaユーザーに限定) (item\_option\_new)
  + 変数：Oktaマスターグループを選択 (item\_option\_new)
  + 変数：Oktaマスターグループを選択 (item\_option\_new)
  + 変数：ユーザー (item\_option\_new)
  + 質問の選択肢：アクティブ化 (question\_choice)
  + 質問の選択肢：非アクティブ化 (question\_choice)
  + 質問の選択肢：パスワード期限切れ (question\_choice)
  + 質問の選択肢：要素リセット (question\_choice)
  + 質問の選択肢：パスワードリセット (question\_choice)
  + 質問の選択肢：一時停止 (question\_choice)
  + 質問の選択肢：ロック解除 (question\_choice)
  + 質問の選択肢：停止解除 (question\_choice)
  + カテゴリ：Okta (sc\_category)
  + 利用可能なカテゴリ：'x\_okta2\_actpack.user' ロールを持つOktaユーザー (sc\_category\_user\_criteria\_mtom)
  + カタログ項目：Okta：ユーザーをグループに追加 (sc\_cat\_item)
  + カタログ項目：Okta：ユーザーをグループから削除 (sc\_cat\_item)
  + カタログ項目：Okta：ユーザーのライフサイクル (sc\_cat\_item)
  + カタログ項目カタログ：サービスカタログ.Okta： ユーザーをグループに追加 (sc\_cat\_item\_catalog)
  + カタログ項目カタログ：サービスカタログ.Okta： ユーザーをグループから削除 (sc\_cat\_item\_catalog)
  + カタログ項目カタログ：サービスカタログ.Okta： ユーザーのライフサイクル (sc\_cat\_item\_catalog)
  + カタログ項目カテゴリ：Okta.Okta：ユーザーをグループに追加 (sc\_cat\_item\_category)
  + カタログ項目カテゴリ：Okta.Okta：ユーザーをグループから削除 (sc\_cat\_item\_category)
  + カタログ項目カテゴリ：Okta.Okta：ユーザーのライフサイクル (sc\_cat\_item\_category)
  + 利用可能なカタログ項目：Okta：ユーザーをグループに追加.'x\_okta2\_actpack.user' ロールを持つユーザー (sc\_cat\_item\_user\_criteria\_mtom)
  + 利用可能なカタログ項目： Okta： ユーザーをグループから削除.'x\_okta2\_actpack.user' ロールを持つユーザー (sc\_cat\_item\_user\_criteria\_mtom)
  + 利用可能なカタログ項目：Okta：ユーザーのライフサイクル.'x\_okta2\_actpack.user' ロールを持つユーザー(sc\_cat\_item\_user\_criteria\_mtom)
  + システムプロパティカテゴリ：Oktaアクティビティパック (sys\_properties\_category)
  + HTTPヘッダー：権限付与 (sys\_rest\_message\_fn\_headers)
  + HTTPヘッダー：権限付与 (sys\_rest\_message\_headers)
  + HTTPヘッダー：コンテンツタイプ (sys\_rest\_message\_headers)
  + スコープスクリプトアクセス：AbstractAjaxProcessor (sys\_scope\_script\_access)
  + スコープスクリプトアクセス：JSON (sys\_scope\_script\_access)
  + フォームセクション：x\_okta2\_actpack\_okta\_group (sys\_ui\_section)
  + スタイル：x\_okta2\_actpack\_okta\_group (sys\_ui\_style)
  + スタイル：x\_okta2\_actpack\_okta\_group (sys\_ui\_style)
  + 含まれるロール：x\_okta2\_actpack.user.x\_okta2\_actpack.admin (sys\_user\_role\_contains)
  + ユーザー条件：'x\_okta2\_actpack.user' ロールを持つユーザー (user\_criteria)