

既存クラウドサービスの活用による業務改善 —Web アプリケーションに依らない申請システムの構築—

佐々木 宏孝¹⁾

1) 自然科学研究機構 岡崎統合事務センター

hsasaki@orion.ac.jp

Improving Business Processes by Leveraging Existing Cloud Services: Building an Application System without Relying on Web Applications

Hiroataka Sasaki¹⁾

1) National Institutes for Natural Sciences, Okazaki Administration Center.

概要

従来、紙媒体で行っていたネットワーク利用希望者の申請に対し、専用の Web アプリケーションを開発することなく、Google Form、GAS と Go を活用して申請システムを構築した。本稿では、学習コストや開発費用といった課題を回避し、既存の IT 資産を最大限に活用して業務効率化を実現した手法と、その導入効果について報告する。

1 はじめに

岡崎統合事務センターのユーザ管理には、オンプレミスの Active Directory (以下、AD) を利用している。従来の手法では、ユーザが手書きで作成した紙媒体もしくは電子ファイルの申請書を基に AD へのデータ登録を行っていたため、入力ミスに起因するログイン不能といった問題が発生していた。この課題に対し、専用の Web アプリケーション開発も検討された。しかし、多大な開発コスト・学習コスト、そして専門技術者の確保といった点で、現実的な選択肢ではなかった。この『Web アプリ開発の壁』を乗り越えるため、既存のクラウドサービスを工夫して組み合わせることで、低コストかつ迅速に課題解決を図ることを目的とした。

自然科学研究機構岡崎 3 機関(分子科学研究所・生理学研究所・基礎生物学研究所) および岡崎統合事務センター (以下、岡崎 3 機関等) で導入した Google Workspace のコアサービスである Google Form を利用することで、フロントエンドを簡易に構築できる点に着目した。システム構成図を図 1 に示す。

2 構築について

2.1 Google Form

Web アプリケーションの構築でのデメリット

である学習コストの高さを解消するため、GUI で入力フォームを設定可能な Google Form を採用した。入力情報のチェック処理は、Google Form に搭載されているチェック処理で問題なく行えた。入力データに関しては、フォーム送信時に JSON ファイルを作成[1]し、Google Drive に格納した。

また、岡崎 3 機関等のセキュリティポリシーでは Google のサービスを利用する際には多要素認証が必須のため、Google Form も本人確認性が高い。

2.2 オンプレミス

Google APIs[2] を利用し、Google Drive に格納している JSON ファイルを取得し、データベースに登録を行う。その後、データベースの登録内容を AD に対して登録・更新を行う。

また、取得した JSON ファイルは、クラウド上にある Spread Sheet に JSON ファイルのファイル ID を記載する。JSON ファイルのファイル ID が Spread Sheet に記載済みの場合は、ファイル ID が記載済みの JSON ファイルの取得処理はスキップする。

2.3 クラウドサービス

Google Workspace 以外のクラウドサービスでは、Google Cloud Platform (以下、GCP) 上のサービスアカウント[3]を利用した。GCP からサービスアカウントを発行したことにより、個人アカウントでのログインを行わずに Google Workspace 上のリ

ソースが利用可能になる。そのため、オンプレミス上で動作しているバッチ処理は、多要素認証の影響を受けずに指定した時間に処理を行える。

2.4 セキュリティ

クラウド利用で、問題となるセキュリティには、以下の懸念があった。

- サービスアカウントの乗っ取り
- 担当者アカウントの乗っ取り

まず、サービスアカウントには、JSON ファイルが格納されているフォルダへの読み取り権限、並びに Spread Sheet の編集権限[4]のみを与えている。そのため、サービスアカウントの乗っ取りが起きた場合でも、甚大な被害が発生しないようになっている。例として乗っ取りが起きた場合を考える。新規ユーザアカウント申請時に生成される JSON ファイルには、申請者のメールアドレス、利用者の氏名、身分、所属、メーリングリスト、利用開始日、利用者のユーザ ID が記載されている。しかし、この JSON ファイルが格納されている URL の一部はランダムで生成されているため、JSON ファイルの取得は困難である。

次に、担当者アカウントは、岡崎 3 機関等のセキュリティポリシーに基づいて、多要素認証を用いているため、パスワードのほかにトークン情報の取得及び操作が別途必要になるためアカウントの乗っ取りは容易ではない。

上記により、本システムの構成により岡崎統合事務センターの AD に不正な Windows アカウントが作成は容易ではない。

2.5 期間

開発初期段階では技術の習熟を要したため、最初の機能開発には約 320 時間を要した。しかし、以降の機能追加においては、開発プロセスの標準化や基本コンポーネントの再利用が可能となったため、1 機能あたり約 30 時間へと工数を大幅に短縮できた。

3 導入後について

3.1 成果

本システムを導入した結果、冒頭の「1. はじめに」で挙げた入力ミス等に起因するログイン不能問題は解消された。また、副次的な成果として以下項目も確認された。

- ユーザ問い合わせの減少
- 担当者の作業負担の減少
- 申請者の作業負担の減少

- 登録内容の揺れの減少
- バグフィックスや機能追加が容易

申請内容を AD に自動登録することにより、AD に登録の揺れがなくなった。さらに、担当者は申請内容に問題ないかの確認作業のみになり、担当者の作業負担が減少した。

副次的な効果として、繁忙期では他のトラブル対応に人的リソースを割く事ができるようになった。

また、本システム導入後、ユーザから以下の声があった。

- 申請書のフォーマットを探す手間が減少
- Google Form のチェック処理を利用し、申請情報の確認作業が効率化された

上記の 2 点により、担当者だけではなくユーザ側にもメリットがあった。

3.2 費用

本システムの人的リソースは約 440 時間程度であり、追加費用は発生していない。仮に同等の Web アプリケーション機能を外部委託するとサーバ構築並びにシステム開発で 60 万円程度必要になり、保守に年間 60 万円程度かかると見込まれる。また、組織で独自で構築する場合は、サーバの構築を含んだ学習コストの高さを考慮すると本手法の費用対効果は高いと考えられる。

参考文献

- [1] <https://developers.google.com/apps-script/reference/forms?hl=ja>
- [2] <https://pkg.go.dev/google.golang.org/api>
- [3] <https://cloud.google.com/iam/docs/service-account-overview?hl=ja>
- [4] <https://developers.google.com/workspace/drive/api/guides/ref-roles?hl=ja>

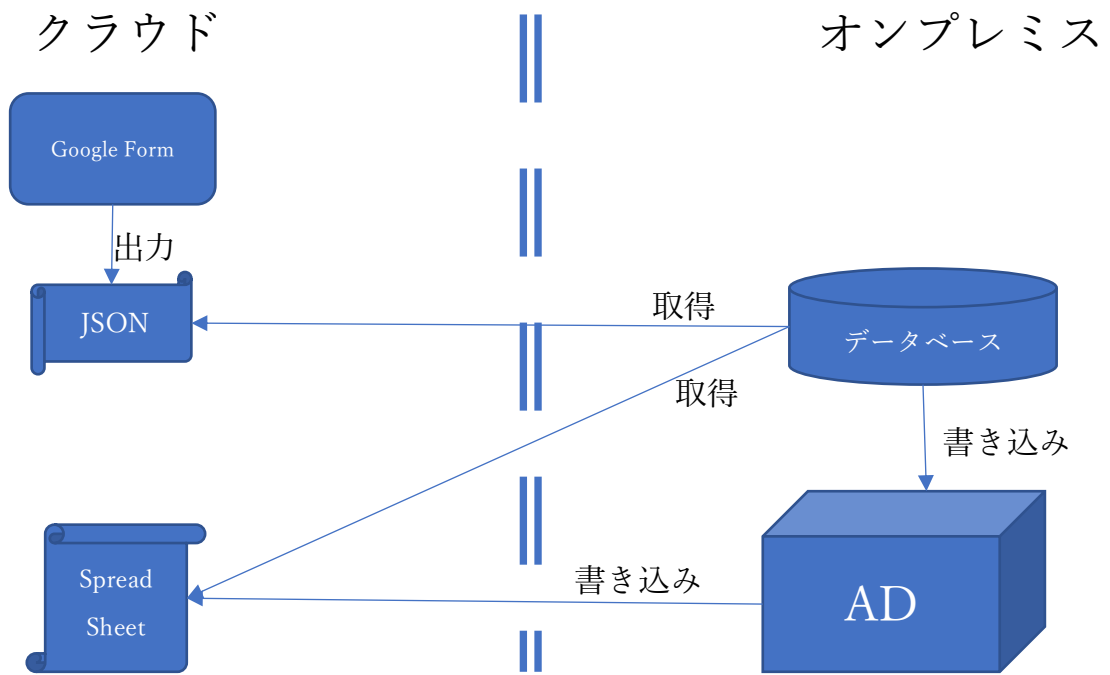


図 1