

ソーシャルネットワーキングを指向した Web カレンダの開発

鈴木 政親 早川 栄一
拓殖大学 工学部 情報工学科

Masanori.Suzuki@os.cs.takushoku-u.ac.jp, hayakawa@cs.takushoku-u.ac.jp

本報告では、複数ユーザによるグループであるソーシャルネットワークを指向した Web カレンダの開発について述べる。SNS や Web サービスなどの近年のネットワーク上でのコミュニケーションの増大により、Web 上でスケジューリングを行う頻度が向上している。このような要求に対して、本システムでは、ユーザが複数のグループ上で予定調整やタスク管理といったことを容易にする、Web カレンダの開発を行った。本システムの特徴としては、(1)全体として合意の取りやすい予定調整ツールの提供、(2)カレンダーと統合したタスク管理ツールの提供がある。これによって、ソーシャルネットワークに対するスケジューリングが容易になった。また、このためのプラットフォームとしてマルチカレンダーインタフェースを持つ、カレンダーシステムを Ruby on Rails で開発した。これによって、柔軟なカレンダーインタフェース開発基盤を提供した。

Development of a Web Calendar System for Social Networking

Masanori Suzuki, Eiichi Hayakawa
Department of Computer Science, Faculty of Engineering, Takushoku University

This report presents the development of a Web calendar system for social networking that consists of tiny groups. Increasing network communications on SNS or Web services, the Web calendar system that supports on-line scheduling or task management was developed. The features of the system are following: (1) usable scheduling tool for floating schedules, (2) task management tool integrated the calendar. The platform of the calendar system was developed by Ruby on Rails that utilizes multi-calendar interface. Flexible and powerful calendar platform is available on the system.

1. はじめに

近年の mixi などのソーシャルネットワーキングシステム(SNS)や、Google や Yahoo!による Web サービスの提供により、複数のユーザから構成されるグループ間で情報をやり取りすることが容易になった。また、情報は Web 上に留まらず、携帯電話などを通じて、実際の生活にも大きく影響を及ぼしている。仕事上での予定調整や会合の日程決定なども、従来のメールベースから、Web 上で提供されるこれらのサービスを利用したものへ変化しつつある。

このような利用形態の変化に対して、いろいろなカレンダーアプリケーションが提供されている。カレンダーアプリケーションは、従来のカレンダーに近いユーザインタフェースをユーザに提供する。このようなカレンダーインタフェースはユーザになじみのある直感的なインタフェースを提供できるというメリットがある。

その一方で、複数ユーザによるソーシャルネットワークの構築と、その中の情報交換が進んでいる。特に、Web 上でのコミュニケーションを通して、実際のイベントへの予定調整や、タスク管理をグループごとに行うことが増加している。ここでは、実世界と Web 世界とで連携し、一人が複数のグループに加入して、その中でそれぞれ役割を持っている点が

特徴的である。複数のグループに加入していることが増加すると、それに合わせて約束や日程の調整といったスケジューリングが必要になってくる。

今後、ソーシャルネットワーク化が進むにつれて、複数のグループや、利用場面に応じたスケジューリングは増大する。しかし、従来のカレンダーアプリケーションでは、カレンダーの延長に留まっていることから、このようなニーズに対応しきれていない。

本研究では、ソーシャルネットワークを応用のターゲットとし、これに適した Web カレンダーアプリケーションの開発について述べる。特に、カレンダー上で重要になる予定調整およびタスク管理ツール、マルチカレンダーについて述べる。さらに、複数の機能を実装しやすい、拡張可能な Web カレンダーの構成について述べる。

2. 問題分析

ソーシャルネットワークとは、ネットワーク上での複数ユーザによるグループから構成されるユーザ同士のつながりのことである。ソーシャルネットワークでは、実際のイベントや行事などの日程を決定するために、ユーザ同士で予定を調整してスケジューリングしていくことが頻繁に行われる。スケジューリングアプリケーションとしては、Microsoft

Outlook や Apple の iCal, Mozilla の Sunbird, Lightning[1]などに代表されるスケジューリングアプリケーションなどがある。これらは、単体で動作するアプリケーションであり、PC にインストールが必須である。デスクトップ上での利用がメインであり、多様な環境下での利用を想定していない。

その一方で、Web インタフェース上で動作する Web カレンダーがある。例えば、Google カレンダー[2] や Yahoo!カレンダー[3]は、このような Web カレンダーの例である。これらは Web 上で利用可能であることから、PC の環境に依存せず多様な環境下で利用することが可能である。そのことから、ソーシャルネットワークが必要とする実際のイベントや行事に対するスケジューリングに対して大きなメリットとなっている。これらを見てみると、次のような特徴が存在する。

- 複数のカレンダーの参照が可能になっている。
- カレンダーへの参照や共有をある程度制御することができる。
- Ajax を使っていて、ユーザが使いやすい。
- API が公開されていて、他の Web アプリケーションと組み合わせ利用できる。

しかし、これらの既存の Web カレンダーでは、次の問題点がある。

- 共有は参照だけであり、編集ができないので、細かな予定調整が必要なソーシャルネットワークには向かない。
- 予定調整は、決定までにグループ間での調整や意見交換が必要とされる。これはカレンダーとは別のメディア、例えばメールや掲示板などに頼らざるをえない。また、予定の再調整の場合は、スケジュールの編集しかできないことから、大きな時間がかかってしまう。
- API は公開されていて、マッシュアップは可能だが、UI を作り直さないといけない。サービス基盤を作るのには不向きである。
- ユーザインタフェースが煩雑である。特に、予定作成の手間がかかる。
- 複数のデバイスに適した画面構成が難しい。特に、携帯電話向けのインタフェースは、使いやすいものではない。

3. 設計方針

3. 1 対象ユーザ

本システムでは、ソーシャルネットワークといわれる、複数ユーザによるグループワークのコミュニティを対象とする。また、幅広いユーザに対応可能のように、Web をベースとしたカレンダーアプリケーションを提供する。

3. 2 設計方針

上記問題を解決するために、次の設計方針を立てた。

(1) カレンダーデータを利用したスケジューリングを

支援

カレンダーインタフェースは、ユーザにとって馴染みがあり、使いやすいインタフェースの一つである。PC を利用しない場合でも、手帳やカレンダーを用いることで、このようなインタフェースの利用には習熟している。本システムでは、カレンダーをベースとしたスケジューリングを支援する環境を作成する。

(2) Web カレンダーをプラットフォームとして、マルチカレンダー環境を実装

通常の手帳では難しい、複数のカレンダーデータを同一の画面上に表示する環境を提供する。また、ソーシャルネットワークを意識して、グループ単位で複数のカレンダーを保持できるようにする。これによって、グループごとにアクセス制御をしながら、複数のカレンダーを利用することが可能になる。

(3) 予定調整ツールによる予定決定までの高速化

従来、ソーシャルネットワーク内での予定調整は、メールや掲示板などの手法を用いるのが一般的であるが、この方法ではネットワーク内の関係や、あいまいな予定を考慮した予定調整が難しい。そこで、他人の動向を把握しつつ予定調整が可能な予定調整ツールを提供することで、最終的な予定決定を高速に行えることを可能にする。また、予定に伴って発生する作業に対して、タスクという単位で管理できるようにする。これをカレンダーと統合することで、ユーザは一貫した操作で作業を行うことができる。

(4) 拡張可能かつ単純なシステム構造

カレンダーの開発基盤として、Ruby on Rails (以下 Rails) [4,5]を採用し、プログラムの見通しをよくするとともに、複数のモジュールやバックエンドのデータベースを容易に利用可能にし、拡張しやすい構造とした。

4. 設計

4. 1 全体構成

全体構成を図 1 に示す。

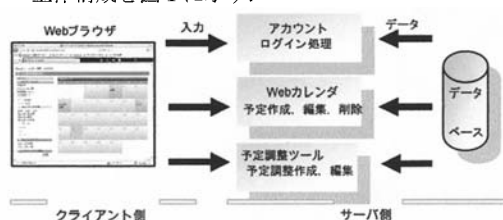


図 1 全体構成

予定管理とカレンダー管理が可能である Web カレンダーをプラットフォームとして、ソーシャルネットワークでの予定調整を可能とする予定調整ツール、ユーザ間で作業をタスクとして保存し共有可能なタスク管理ツールを実装する。また、本システム

では画面制御を Ruby プログラムで行い、それらを RHTML 文で HTML 中に埋め込んでいる。これにより、データベースからのデータを直接 RHTML で処理することができる。これによって、画面出力の変更などはページの修正だけで可能であり、全体の把握がしやすくなり、画面制御の仕様変更を容易にすることができる。

4. 2 予定調整ツール

予定調整ツールは、ソーシャルネットワークでの予定調整を行うツールである。予定調整ツールでのフローを図 2 に示す。

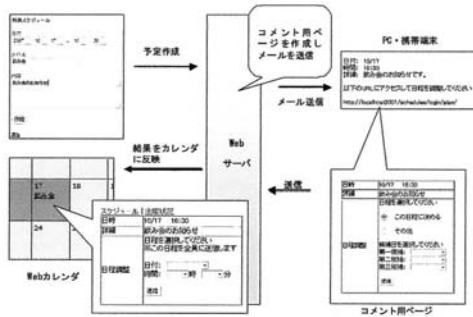


図 2 予定調整ツールでのフロー

本ツールでは、予定調整用のスケジュールを作成し、作成したスケジュール内でユーザ同士が意見を交換できるようにしている。予定調整データはソーシャルネットワークの各ユーザにメールで送信される。ユーザは送られたメールに添付されている URL から意見を送るためのコメントページにアクセスを行い、都合のよい日を選択する。ここでの意見は Web カレンダー上に反映されユーザ全員が見ることが可能である。これによって、他のユーザのスケジュールや都合がよい日時などの出席状況をグループ間で把握することができるようになる。結果として、日程調整までにかかる時間と手間を減少させ、予定決定を高速化することができる。

4. 3 タスク管理ツール

タスク管理ツールは、作業をタスクとして保存することで、作業内容を複数ユーザで共有できるようにするツールである。タスク管理ツールについて、図 3 に示す。

ソーシャルネットワークでは、プロジェクトや作業計画を管理するためにタスクという単位で管理している。タスクを使い計画の進行を確認しながら作業をするが、既存のシステムでは作業に必要なデータを別のフレームワークで管理していることから、作業とそのデータとの関連性を意識しにくくなっている。

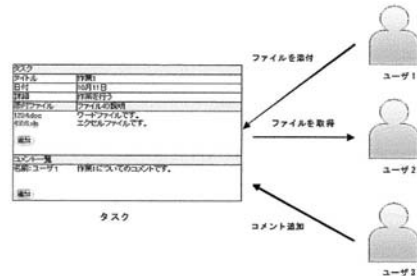


図 3 タスク管理ツール

本システムでは、カレンダー上にあるスケジュールデータを利用したタスク管理を可能にしている。このために、次の三つの機能を実装している。

(1) コメントの添付

ネットワーク内の他のユーザから意見をもらい、修正することを可能にするために、タスクに対してコメントを添付できる機能を提供する。

(2) ファイルデータの添付

タスクデータに対して、文書ファイルなどのファイル類を添付できるようにし、タスクに多くのデータを残せるようにする。これによって、ソーシャルネットワークの全ユーザが容易にファイルデータを利用できるようになり、タスク単位での作業を容易にする。

(3) カレンダー上でのタスク管理

既存のシステムではタスクを管理することができる機能はあるが、カレンダー上にタスクのデータを表示することができない。そこで、カレンダー上にタスクデータを表示して、ユーザのスケジュールと一緒に表示することを可能にした。これによって、自分のスケジュールの状況とタスクの作業内容の把握が容易になり、計画全体を管理しやすくなる。

4. 4 Web カレンダー

本システムでは、Web カレンダーをソーシャルネットワークのための機能を実装するプラットフォームとして提供している。ユーザが複数のカレンダーを所持できるマルチカレンダー機能を提供する。既存のカレンダーでは、スケジュール情報だけが表示可能であり、タスクデータや予定調整の結果などの情報をカレンダー上に表示できない。本システムでは、これらのデータをカレンダーとして扱い、一つのカレンダーの中に合わせて表示することを可能にする。

マルチカレンダーの例を図 4 に示す。



図4 マルチカレンダー

ここでは、タスクデータ、予定調整、スケジュールをすべて一つのカレンダーの中で表示する。それぞれは個別の情報であるが、色を変えて表示することで、ユーザはカレンダーを切り替えずに内容を把握することができる。

4. 5 ユーザインタフェース

本システムのユーザインタフェースでは、従来の3ペイン型の表示形式をベースに、より把握しやすいインタフェースを提供している。本システムの例を図5に示す。



図5 本システムのスクリーンショット

カレンダーは、上位に(1)ヘッダメニュー、左側に(2)メニュー一覧、右側に(3)カレンダーの三つの部分から構成されている。ヘッダメニューでは、どのページでも共通の操作について表示している。メニュー一覧では、スケジューリング、カレンダー管理、グループワークなどの機能が表示されている。これらを独立して、左側に表示することで、グループ作業などを頻繁に行う場合に、メニューやリンクをたどらずにすむという利点がある。右側には、月単位でのカレンダーを表示している。詳細情報は、各日をクリックすることで表示できる。

予定作成インタフェースを図6に示す。予定作成インタフェースでは、リストボックスを使用しないインタフェースを提供している。これは、予定作成時には選択項目が多く、リストボックスは余計に手間がかかってしまうという問題があるからである。特に、複数ユーザでのスケジューリングのように、

設定項目が多い場合には、これはデメリットとなる。そこで本システムでは、リストボックスを設けず、ユーザが簡単に入力することで、迅速な予定作成を可能にする。

また、所持しているカレンダーに登録されている予定の一覧をカレンダー別に閲覧することができる。スケジュール一覧機能を提供している。これにより、マルチカレンダーの欠点である、スケジュールがどのカレンダーに登録されているかが分かりにくいという問題を解決する。また、カレンダーごとの色分けが可能であり、それをメインのカレンダーに反映させることができる。これによって、カレンダーとスケジュールの内容とを対応づけて把握することが容易になる。

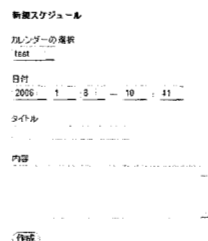


図6 予定作成インタフェース

4. 6 カレンダー共有

本システムでのカレンダー共有の流れを図7に示す。

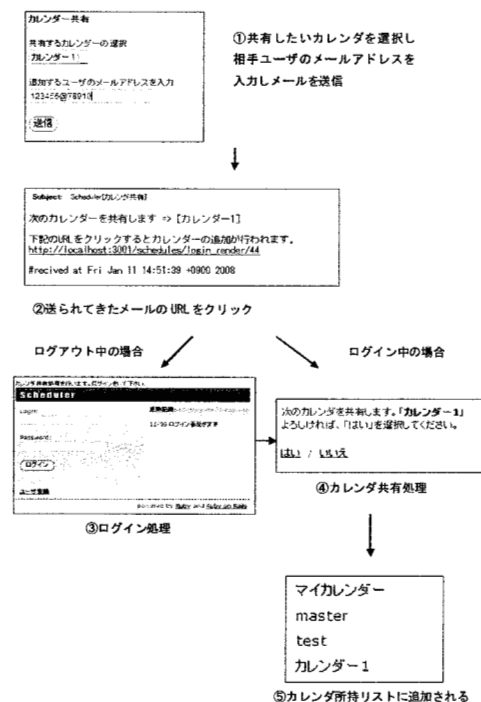


図7 カレンダー共有のフロー

ソーシャルネットワークでは、多くのユーザとのスケジュール共有が必要である。また、ソーシャルネットワークでのグループは、1 ユーザが複数のグループに参加していることが一般的であり、このグループはプロジェクトや計画が終了した段階で解散を行う。このため、カレンダーは、複数のユーザでの共有や削除が容易でなければならない。

本システムでは、カレンダー共有にメールを用いることが可能となっている。メールに添付した URL から共有処理を行うことが可能である。また、カレンダー所持リストから削除することで、カレンダーを削除できるようにした。この場合でも、所持リストからの削除であり、スケジュールそのものは削除されないことから、予定共有と削除が安全かつ容易に可能である。

5. 実現

5. 1 開発環境

本システムは、Windows/Linux 上で動作する Ruby で記述されている。前述のとおり、アプリケーションプラットフォームとしては、Ruby On Rails を用いた。また、RHTML を含む HTML、および JavaScript によるユーザインタフェース支援を含んでいる。全体では 1798 行である。

また、Rails のアプリケーションは、サーバ負荷が高く、応答性に問題がある場合がある。本システムでは、Apache および Mongrel を利用し、負荷分散をするように構築した。ユーザからのリクエストをサーバ側で動的コンテンツと静的コンテンツとに分類し、静的コンテンツについては Apache で、動的コンテンツについては Mongrel 上で動作する Rails を使うように設定した。これによって、システムにかかる負荷を分散させることができ、ソーシャルネットワークが想定する多くのユーザの使用でも、応答性が低下することを防いだ。

5. 2 プログラム構造

本システムは、Rails が提供する MVC フレームワークに基づいて実装している。特に、この中でも本システムそのものである Controller 部分について説明する。本システムのプログラム構造を図 8 に示す。

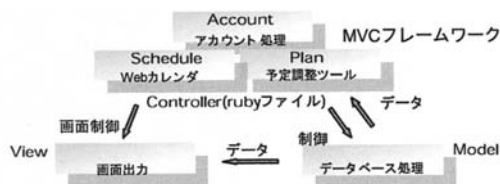


図 8 プログラム構造

本システムでは、この Controller 部分を次の四つに分類して構成するように実装した。

(1)Application Controller

アプリケーション全体に関する処理を記述する部分であり、コードの重複を避けるために用いる。ここでは、アカウントを持っているユーザだけが使用できるようにして、アクセス制限をかけている。

(2)Account Controller

アカウントに関する処理を行うコントローラである。ソーシャルネットワークでは、ユーザ名によって誰がスケジュールの作成者か、誰がタスクを管理しているのか、といったデータがよく用いられる。そこで、ユーザアカウント関連の処理を Web カレンダー処理と分離して、このコントローラで行われるユーザデータに関する処理を他のコントローラで使えるようにすることで、処理の重複を避け、ユーザアカウント関連の処理を管理しやすくする。

具体的には、ユーザ関連情報、グループ関連情報、ログイン関連情報の三つの情報を扱う。

(3)Schedule Controller

スケジュールやカレンダーに関する処理を行っているコントローラである。カレンダーでもっとも用いられているスケジュールの作成や削除の機能を、すべてこのコントローラにまとめている。具体的には、スケジュール関連クラス、カレンダー関連クラス、カレンダー共有クラスから構成される。

(4)Plan Controller

予定調整、タスクに関する処理を行っているコントローラである。予定調整関連クラスから構成される。これは予定調整にメール処理が必要なため、コントローラを分離している。

6. 実現予定の機能

現在、本システムはカレンダープラットフォームと、それを用いた二つのツールが実現した段階である。今後は、次の機能を逐次実装していく予定である。

(1)過去予定の参照ツール

多くの予定は、過去の予定の繰返しであり、ユーザはそこから逆算して求めることができる。

(2)複数のパーソナルのサポート

現在の Web カレンダーでは、他のユーザへの予定の公開制御は、大雑把な単位でしかできていない。実際に、複数のネットワークを築いた場合は、公開する予定についての制御は難しい。これに対して、どのネットワークに対して、どのような情報を提供するかを、細かく定義できるインタフェースを提供する。

(3)他のコミュニケーションツールとの統合

現在は、予定調整ツールはメールを併用することが可能になっている。しかし、メールは意見がすれ違いやすく、「相手の出方を伺って決める」場合には、

有効なコミュニケーション手段ではない。今後、確認、未確認のチェックや、重要な参加者の決定をイベントとして通知するような機構を導入する。

(4) スクリプトの添付

カレンダーでは、細かなルールを設定することが必須となるが、このようなルール設定は既存のカレンダーでは難しい。例えば、「祭日以外の毎週木曜日の5時から7時までを、11月末まで」といったスケジュールは、従来のカレンダーでは定義が難しい。このような複雑な条件のスケジュールについては、プログラムによる記述を添付することによって可能になると考えている。

また、ユーザインタフェースについては、次の2点が課題である。

- 携帯電話ユーザインタフェースの提供
- Ajax によるユーザインタフェースの向上

7. おわりに

本報告では、ソーシャルネットワークを応用のターゲットとし、これに適した Web カレンダーアプリケーションの開発について述べた。Web カレンダーの構築に必要な拡張可能なプラットフォームを Ruby on Rails で開発し、その上でソーシャルネットワークを志向した予定調整およびタスク管理ツール、マルチカレンダー機能を実現した。

今後の課題としては、6章で述べた機能の実装、および、実用化によるユーザからのフィードバックを基にした評価が挙げられる。

参考文献

- [1] Lightning プロジェクト
<http://www.mozilla-japan.org/projects/calendar/lightning/>
- [2] Google Inc: “Google Calendar BETA”
<http://www.google.com/calendar/>
- [3] Yahoo Japan Corporation: “Yahoo! カレンダー”
<http://calendar.yahoo.co.jp/>
- [4] Ruby on Rails
<http://www.rubyonrails.org/>
- [5] Ruby on Rails リファレンスマニュアル
<http://techno.hippy.jp/apid>