

Social Scheduler: 携帯電話を用いた協調的スケジューラ

Social Scheduler: Collaborative Scheduling System with Mobile Phone

大向 一輝¹⁾ 濱崎 雅弘¹⁾
Kentaro HATORI Toshio HARADA

1) 総合研究大学院大学 (〒101-8430 東京都千代田区一ツ橋 2-1-2 国立情報学研究所 大学院
E-mail: i2k@grad.nii.ac.jp)

ABSTRACT. This paper proposes a collaborative approach for personal task management in which people can decide management of their tasks according to their environments including their subjective and multivalent judgement and human relations. In our approach the task management is modeled as an alliance of individuals. Alliance model is based on information sharing and collaboration of several persons. Users disclose their task condition and maintain to be updatable by friends. We propose emergent group discovery algorithm to control the level of disclosure. We implement client/server system called Social Scheduler on mobile phones environment and remark the advantages of our approach with some context.

1. 背景

情報化社会の進展により、個人の持つ時間や金銭、知識といった資源（リソース）管理の必要性が高まっている。リソース管理については、すでに多くの組織内でグループウェアやERPが導入され、活用されている。しかしながら、異なるシステムを導入している複数の組織にまたがる情報の共有に関しては、それぞれのシステムのデータ構造が統一されていないなどの問題により、実現することが難しい。

また、これらのシステムは、グループの構成員が持つリソースを、全てグループのために利用できるという前提のもとで設計されている。そのため、構成員のリソースに関する情報は全てが公開され、グループ内で管理される。

近年では個人が複数のグループに同時に所属し、活動することは珍しくない。このような状況では、上述の理由により、異なるシステム間での連携が不可能であり、またグループ内のリソースを最大限に活用しようとするシステム間で競合が発生する。その結果、ユーザがシステムを利用することで得られる効用は極めて小さいものとなる。

一方、個人単位での情報管理の手法としては手帳の利用や、PDA (Personal Digital Assistant) およびPIM

(Personal Information Manager) の利用といったものが挙げられる。しかし、これらの手法では管理に必要な情報をユーザが全て入力しなければならず、そのためのコストは非常に大きい。また、個人のための管理手法は恣意的なものが多く、効率的なリソース管理を行うことは難しい。

さらには、グループ中心のアプローチとは異なり、効用を定義することが極めて難しい。そのため、従来より研究されていた人工知能やOR (Operations Research) の各手法がそのまま適用できることは少ない。

本研究では、以上の問題点を解決するために、個人をリソース管理の主体とし、個人同士の協調および連携を可能とするモデルを提案する。そして、スケジューリン

グ問題への適用例として「Social Scheduler」を設計し、検討を行う。

2. 目的

本ソフトウェア（以降 Social Scheduler と表記）は携帯電話上で動作する Java アプリケーション (i アプリ) を含むスケジューリングシステムである。従来の同様のソフトウェアと比較して、以下のような特徴を持つ。

- ・インターネットを介したスケジュール情報の保存機能および公開機能
- ・ユーザ同士の情報共有によるスケジューリング支援機能
- ・TODO 情報の統合によるスケジュール提案機能
- ・携帯電話の画面に適したインターフェイス

Social Scheduler を使用するユーザ同士は認証やタスクの依頼によって随時グループを形成する。Social Scheduler はグループ内のスケジュール決定を迅速に行うことを目的としている。

基本的な操作手順を以下に述べる。

1. ID の登録

提案システムを利用する複数のユーザを区別できるように、ID とパスワードの登録および管理を行う。Web サーバ上の CGI を用い、クライアント端末からのインターネット接続によって ID とパスワードを発行する。それと同時に、サーバ側にユーザのデータ保存領域を確保する。

2. 相互認証

サーバに登録されたユーザ同士が情報共有を行うためには、その前段階として相互認証が必要となる。これはクライアントソフトウェアの機能として実装し、相手のIDを指定するとその相手にメッセージが送信される。相手は認証するか否かを決定し、その結果がサーバに登録される。全ての認証情報はサーバに格納される。また、ユーザ自身の認証情報はクライアント側でも保持する。一連の動作により、サーバ上では認証ネットワークが形成される。

3. タスクの依頼

提案システムを利用するユーザ同士がタスクを依頼す

る場合には、クライアントソフトウェアの機能を使用する。依頼者はタスク情報を記述し、認証済みリストから相手を選択し、サーバに送信する。サーバは依頼関係データベースにこれを記録し、相手にタスク情報を送信する。相手はこれを受理するかどうかを決定する。この結果は再びサーバに送信される。この一連の動作により、サーバ上では依頼関係ネットワークが形成される。

4. 情報共有

認証を行った相手のスケジュールは、協調タスクビューアの機能を利用して随時クライアントソフトウェアで閲覧することができる。ただし、閲覧できるスケジュールはあらかじめ次に述べるようにフィルタリングされている。手順としては、クライアントソフトウェアにおいて認証済みの名前リストから複数人を選択すると、サーバは認証関係を判断し、それに応じてクライアントにデータを送信する。また、相手のスケジュールとユーザ自身のスケジュールを重ね合わせ、共同作業のようなタスクの入力を支援することも可能である。

5. フィルタリング

サーバ上では依頼関係ネットワークをもとにグループを同定する。ここでのグループとはグラフ理論における部分グラフが完全グラフであるようなものを指す。ユーザは個々のタスクを公開するかどうかを決めるが、その際に自動フィルタリングの設定を行うと、グループ内のメンバーのみにタスク情報を公開する。逆に、グループの範囲を超えた情報公開は行わない。

3. 開発内容

(1) システム構成

Social Scheduler のシステム構成を図 1 に示す。

Social Scheduler はクライアント・サーバ型のシステムである。インターネット上のサーバと携帯電話間においては TCP/IP 等を用いた通信が不可能であることから、株式会社 NTT ドコモが提供する通信センター (i モードセンター) を介し、サーバ・i モードセンターでは TCP/IP、i モードセンター・携帯電話間では独自プロトコルを利用している。アプリケーションプロトコルとしては HTTP を利用している。HTTP は両通信プロトコル上でシームレスに動作するため、システムの開発にあたっては通信プロトコルの差異を考慮する必要はない。また、操作によっては E-mail による通信を利用している。現在の携帯電話システムにおける E-mail はインターネットメールと互換性があるため、こちらもシステム間の差異を考慮する必要はない。



図 1 システム構成

各クライアントはインターネット接続が可能な携帯電話である。これらの端末は Java アプリケーションのダウンロードおよび実行が可能である。本プロジェクトでは株式会社 NTT ドコモが提供する端末 (ソニー株式会社製 SO504i, 日本電気株式会社製 N504iS, N504iS, および松下電器産業製 P504iS) を利用した。ソフトウェア開発環境としてはサン・マイクロシステムズ株式会社が提供する Java 2 Micro Edition Connected Limited Device

Configuration (J2ME CLDC) および J2ME Wireless SDK for DoJa を利用した。

サーバは常時インターネットに接続されている。ハードウェアは一般的な PC サーバを用いた。オペレーティングシステムには Debian/GNU Linux 3.0 を使用し、Java によるアプリケーションサーバ実行環境としてサン・マイクロシステムズ株式会社が提供する Java 2 Enterprise Edition (J2EE) 1.2, Java 2 Standard Edition Runtime Environment (J2SE JRE) 1.4, および Apache Project が提供する Tomcat 4.0 を利用した。

(2) データ構造

a) タスクの種類

すべてのタスク情報はサーバ上のデータベースと携帯端末内に格納されている。両者はユーザが明示的に指定することで同期をはかることが可能である。本システムではタスクを個人タスクと協調・依頼タスクの 2 種類に分類している。個人タスクはそれを保持するユーザ自身にのみ関係のあるタスクであり、協調・依頼タスクは複数のユーザに関係するタスクである。後者のうち、協調タスクとはタスクの実行者の中に依頼者を含むものであり、依頼タスクとは依頼者を含まないものである。あるユーザによって協調・依頼タスクが発行されると、サーバはタスク情報からそのタスクに関連のあるユーザリストを取得し、それぞれの個人タスクデータベースに新たな個人タスクとして追加する。協調・依頼タスクを依頼されたユーザが同期を行うと、サーバ上の個人タスクデータベースから携帯端末に協調タスクの情報がダウンロードされ、タスクの依頼が通知される。

b) 個人タスクのデータ構造

- UID : (例 : 20021221160553i2k)
個々のタスクにユニークにつけられた ID 番号。
TIMESTAMP (後述) + 乱数 + 依頼者の ID から構成されている。
- TITLE : (例 : 研究室ミーティング)
タスクの名称。Shift-JIS で記述する。
- START : (例 : 2003/02/16 09:00)
タスクの開始時間。YYYY/MM/DD HH:mm の形式で記述する。
- END : (例 : 2003/02/16 18:00)
タスクの終了時間もしくは締切時間。書式は START と同様である。
- PRIORITY : (例 : 5)
タスクの主観的な重要度。1~5 の整数で記述する。5 が最重要である。
- TIMESTAMP : (例 : 2003/01/23 15:22)
タスクが発行された時間。書式は START と同様である。
- TODO : (例 : false)
タスクの種類の判別フラグ。true であれば TODO タスク, false であれば Schedule タスクである。
- WORKLOAD : (例 : 30)
TODO タスクの仕事量。整数で記述する。単位は時間である。
- ACCEPT : (例 : 21)
協調・依頼タスクにおいて他のユーザが受理しているかどうかを示すフラグ。
- SYNC : (例 : true)
タスクがサーバ上のデータベースと携帯端末内で同期されているかどうかを示すフラグ。true であれば同期済み。情報が変更されると false になる。
- PUB : (例 : -1)
他ユーザにこのタスクを閲覧させてもよいかどうかを示すフラグ。-1 は完全非公開, 1 は完全公開, 0 はスマートフィルタリング適用を示す。
- GROUP : (例 : 1)
タスクの種類の判別フラグ。-1 が個人タスク, 1 が依

頼タスク、0 が協調タスクを示す。

- FROM : (例 : ham)
タスクの依頼者の ID.
- MEMBER : (例 : ham:i2k:akatsuka)
タスクの実行者の ID. 協調・依頼タスクでは個々の ID をコロン区切りで記述する.
- MEMO : (例 : 国立情報学研究所にて)
タスクに関する備考. Shift-JIS で記述する.
- PROGRESS : (例 : 15)
TODO タスクにおける進捗度. 0~100 の整数で記述する. 単位は%である.

c) 協調・依頼タスクのデータ構造

協調・依頼タスクのデータは各ユーザのタスクデータに中にも保持されるが、それとは別にサーバ上でも一括管理される。パラメータは以下の通りである。なお、個人タスクデータと同様のパラメータについては説明を省略する。

- LOCK : (例 : true)
タスク情報を変更できるかどうかを示すフラグ. true であれば変更不可, false であれば変更可である.
- METHOD : (例 : -2)
タスクの状態を表すフラグ. 0 は全てのユーザに承認された状態, その他は承認待ちであることを示す.
- NUM : (例 : 3)
タスクに関連のあるユーザの数. 整数で記述する.
- NAME : (例 : ham)
タスクの実行者の ID. 複数人いる場合には次の STATE とともに繰り返し定義する.
- STATE : (例 : -1)
タスクの受理状態を表すフラグ. -1 は未受理, 0 は受理, 1 は拒否を示す.

d) その他のデータ構造

タスク情報の他にサーバが管理するデータとして以下のものが挙げられる。

- ユーザリスト
ユーザリストには本システムの登録ユーザの情報が格納されている。ユーザリストのデータ構造は次の通りである。
 - ID : (例 : i2k)
システム上のアカウント. 英数半角文字で記述する.
 - PASSWORD : (例 : ****)
システムの認証を受けるためのパスワード. 携帯端末で利用しやすいよう半角数字 4 桁で記述する.
 - E-MAIL : (例 : ngrid@docomo.ne.jp)
携帯端末に付与された E メールアドレス. 管理者からの情報配信等に用いる.
- 認証ユーザリスト
認証ユーザリストはそれぞれのユーザが保持し、ユーザ間認証が行われた場合にエントリーが追加される。データ構造は次の通りである。
 - ID : (例 : ham)
相手のユーザ名.
 - TIMESTAMP

(2) クライアントの実装

クライアントソフトウェアは主にカレンダービュー、タスクの登録・変更画面、協調タスクビュー、およびユーザマネージャの 4 種のインターフェイスから構成されている。本節ではそれぞれのインターフェイスが備える機能について述べる。また、それぞれのインターフェイスに付随するメニュー項目についても述べる。

a) カレンダービュー

カレンダービューはクライアントソフトウェアのデフォルト画面として起動処理の終了後に表示される (図 2)。カレンダービューは携帯端末の小さな画面に合わせて様々な情報を効率よく表示することを目指して設計

されている。表示は上部と下部に大きく分けられ、上部では 1 ヶ月分のカレンダーと各日付のタスクの有無を表示する。日付の下は 6~24 時を表す時間軸を 3 段に分割し、スケジュールタスクが存在する時間に合わせて棒状に表示する。日付の横には TODO タスクの有無を表示し、赤は TODO タスクの締切が存在する日、青は TODO タスクを行わなければならないと予想される日を表している。また、紫は本日であることを表している。カーソルキーで上部のカレンダー上のフォーカスを変えることができる。フォーカスが最上段にある場合にさらに↑を押すと前月に、フォーカスが最下段にある場合にさらに↓を押すと次月に移動する。また、数字キーの「2」「4」を押すと前月に、「6」「8」を押すと次月に移動することも可能である。「5」を押すとフォーカスが本日に戻る。フォーカスされた日付にタスクが存在する場合には、画面下部にタスク名と時間が表示される。ここでは文字の色とプレフィックスによってタスクの種類を判別することが可能である。白文字のタスクは個人タスク、青文字のタスクは依頼タスク、赤文字のタスクは協調タスクを示す。また、タスク名の前に「TD」と記載されたタスクは TODO タスク、その他のタスクは Schedule タスクを示す。タスク名の前に「*」が記載されたタスクはサーバとの同期がなされていないタスクであることを示す。

フォーカスの日付にタスクが 1 つのみ存在する場合には決定キー、2 つ以上の場合にはカーソルでタスクを選択して決定キーを押すとタスクの詳細データ (後述) を見ることができる。

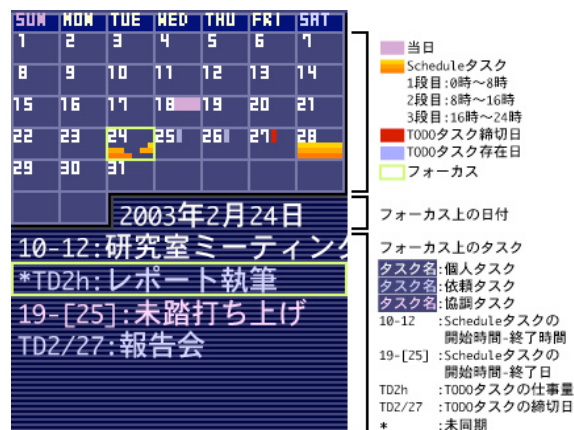


図 2 カレンダービュー

b) タスクの登録・更新画面

タスクの登録・更新画面では新しいタスクのデータ入力および既存のタスク情報の変更を行うことができる。本画面ではタスクに関する全てのパラメータを一覧することが可能であり、これをもとにユーザは個々のタスクに関してパラメータの設定を行う。タスクの登録・更新画面の例を図 3 に示す。各タスクに設定できるパラメータと表示との対応は以下の通りである。

• TYPE

タスクの種類を設定する。個人・協調・依頼の 3 種類と Schedule・TODO の 2 種類の組み合わせで計 6 種類のタスクが設定可能である。

入力項目 (A)

- 個人 : 個人で行うタスク, もしくは ngrid を利用していないメンバーとの協調作業を行う場合に設定する.
- 協調 : ngrid を利用しているメンバーとの協調作業を行う場合に設定する.
- 依頼 : ngrid を利用しているメンバーにタスクを依頼する場合に設定する.

入力項目 (B)

・ **Schedule** : 開始時間と終了時間が決定されており、その間は拘束されるようなタスクである場合に設定する。
例 : 会議・授業

・ **TODO** : 締切時間のみが決定されており、実際のタスクの実行時間がユーザの裁量に任されるような場合に設定する。例 : 論文執筆

・ **TITLE** : 入力項目 (C)

タスクの名称

・ **FROM** : 入力項目 (D)

タスクの依頼者

基本的には手動で入力を行うが、協調タスクを登録する場合は自動的に入力される。

・ **MEMO** : 入力項目 (E)

タスクに関する備考

・ **MEMBR** (Member の略) : 入力項目 (F)

協調タスクを行うメンバーの ID. 個人タスクの場合は空欄。

・ **START** : 入力項目 (G)

Schedule タスクの開始日・時刻

・ **LOAD** (Workload の略) : 入力項目 (H) (I)

TODO タスクの仕事量の見積もり。単位は時間・日・週・月から選択する。

・ **END** : 入力項目 (J)

Schedule タスクの終了日・時刻および TODO タスクの締切日・時刻

・ **PRIOR** (Priority) : 入力項目 (K)

タスクの優先度

High / Mid-High / Middle / Mid-Low / Low の 5 段階。主観的評価を行い入力する。

・ **PROG** (Progress) : 入力項目 (L)

TODO タスクの進捗度。0~100%の 10 段階

・ **PUB** (Publication) : 入力項目 (M)

タスク情報の公開・非公開

・ **all** : 認証関係を結んだ全ての人に公開

・ **smart** : グループ内のメンバーにのみ公開 (後述)

・ **off** : 公開しない

・ **REPT** (Repeat) : 入力項目 (N) (O)

定期的タスクの一括入力支援機能。繰り返し回数を数字で入力し、毎日・毎週・毎月を選択する。

・ **off** : 繰り返さない (1 回のみのタスク)

・ **days** : 毎日繰り返す

・ **weeks** : 毎週の同じ曜日に繰り返す

・ **month** : 毎月の同じ日に繰り返す

・ **mon_w** : 毎月の同じ週・曜日に繰り返す

TYPE	協調	Schedule	0
TITLE	研究室ミーティング		
FROM	i2k		
MEMO	15Fセミナー室にて		
MEMBR	i2k:ham:akatsuka		
START	2003/2/24 10:00		
LOAD	2 hours		
END	2003/2/24 12:30		
PRIOR	MidHi	PROG.	30%
PUB.	Smart	REPT.	2 weeks

(A) (B) (X)
(C)
(D)
(E)
(F)
(G)
(H) (I)
(J)
(K) (L)
(M) (N) (O)
□ フォーカス

図3 新規タスクの登録

c) 協調タスクビュー

協調タスクビューでは各タスクを1日ごとのガントチャート(クロノグラム)上に配置する。また、認証関係を持つユーザのスケジュール情報を取り込み、自身のスケジュールと重ね合わせて表示することが可能である。協調タスクビューの例を図4に示す。最上段に表示されたオレンジ色のバーがユーザ自身のタスクである。カーソルキーでフォーカスを移動させ、選択すると各タスクの詳細が表示される。日付の横に表示された赤の正方形はTODOタスクが存在することを示している。10時から12時までのタスクの直下に同じ長さの紫色のバーが存在する。これは認証関係を持つ他ユーザAが公開しているタスクである。紫色はこのタスクが協調タスクであることを示している。この協調タスクの右に表示された青いバーは、ユーザAがこの時間帯にタスクを保持していない(空き時間である)ことを示している。また、その下の段に存在する青いバーは他ユーザBの空き時間である。これによれば、16時から18時にかけてはユーザ自身にタスクが存在せず、かつユーザAとBの空き時間が存在する。このような場合、その時間帯に3人が関係する協調タスクを投入すると受理される可能性が最も高くなる。そこで、本ソフトウェアでは登録されたデータより、協調ユーザに指定した全てのメンバーに共通する空き時間を自動的に算出し、紫色のフォーカスとして提示する。このフォーカス上で選択を行うと、タスク登録画面に移動する。この場合、算出された時間とメンバーリストがすでに記載された状態になる。



図4 協調タスクビュー

d) ユーザマネージャ

ユーザマネージャではこれまでに認証関係を結んだ全てのメンバーリストを表示する(図5)。また、ユーザごとのタスク依頼数(Outと表記)、タスク被依頼数(Inと表記)、後述のグループ共有数、ユーザ自身が保持するグループ数、およびPageRankによる評価値が表示される。表示はこれらの値をキーにソートし直すことが可能である。また、各ユーザにフォーカスを合わせると、そのユーザに関係したタスクの一覧が画面下部に表示される。ユーザ自身にフォーカスを合わせた場合には、全タスクが表示される。このタスクリストでは、フィルタ設定により表示条件を変更することが可能である。

e) メニュー

カレンダービュー、協調タスクビュー、およびユーザマネージャでは左ソフトウェアキーから基本メニューを呼び出すことができる。メニュー項目の内容を以下に述べる。

・新規タスク

タスク登録画面に移動する。タスクのSTARTパラメータはカレンダービューおよび協調タスクビューではフォ

一カスされた日付に設定される。ユーザマネージャから呼び出された場合には本日の日付になる。

- ・協調メンバー設定

協調タスクビューにおいて公開スケジュールを閲覧したいメンバーを指定する。

- ・同期→

同期には通常の同期と完全同期の2種類がある。

- ・同期

同期が行われていない携帯端末内のタスク情報およびサーバに保存された情報を同期する。その際、他のメンバーによって登録された協調タスク・依頼タスクの情報がサーバに存在すると確認を促すダイアログが表示される。

- ・完全同期

同期が行われている、いないに関わらず、携帯端末内のタスク情報とサーバに保存されたタスク情報一致させる。通常の同期と比較して通信量が多い。主にソフトウェアのアップデートの際に利用する。

- ・環境設定

環境設定には使用頻度の少ない設定項目が収められている。

- ・協調ユーザリスト取得

新たに他ユーザと認証関係を結んだ場合に、携帯端末内に保存された認証ユーザリストを更新する。

- ・協調ユーザ追加

他ユーザと新たに認証関係を結ぶ場合に相手のIDを入力する。

- ・ユーザ設定

別のIDに変更する場合に利用する。

カレンダービュー、協調タスクビュー、およびユーザマネージャでは右ソフトウェアキーから表示メニューを呼び出すことができる。表示メニューでは、これらの画面表示を切り替えることが可能である。右ソフトウェアキーを2回続けて押すと、カレンダービューと協調タスクビューのトグル切替が可能である。

ユーザ名	Out	In	共有/全グループ	評価値
USER	→	←	GROUP	EVAL.
akatsuka	3	8	4/11	384
ham	7	6	7/11	631
coba	2	0	1/11	107
ikns	1	3	2/11	189

認証ユーザリスト
■ フォーカス

*TD2h:レポート執筆
[19]-[25]:未踏打ち上げ
TD2/27:報告会

タスクリスト

図5 ユーザマネージャ

(3) サーバの実装

サーバソフトウェアはデータベースモジュールおよびプログラムモジュールから構成されている。本節ではこれらの詳細について述べる。

a) データベース

データベースモジュールは以下のデータを格納している。

- ・ユーザ情報 (ユーザ DB)

すべての登録ユーザのID、パスワード、および携帯電話のメールアドレスを格納する。

- ・ユーザごとの認証ユーザリスト (認証ユーザ DB)

認証関係を結んだユーザの一覧を保持する。このデータベースは全ユーザの数だけ用意される。

- ・ユーザごとのタスク情報 (タスク DB)

各ユーザが持つすべてのタスクとそのパラメータを保持する。このデータベースは全ユーザの数だけ用意される。

- ・協調タスク情報 (協調タスク DB)

協調タスクおよび依頼タスクに関する情報は、ユーザごとのタスク情報とは別にこのデータベースに集めて一括管理する。

b) データベース

サーバは以下に述べる8種のプログラムモジュールから構成されている。

- ・ユーザ登録モジュール

本システムを初めて利用する際にID、パスワードおよび携帯端末に付加されたメールアドレスの入力を要求する。これらのデータはユーザDBに格納する。登録が正常に行われると、携帯端末にメールを送信する。

- ・ダウンロードモジュール

クライアントソフトウェアのダウンロードページにアクセスした携帯端末の機種およびモデルを判別し、それぞれに適したアプリケーション記述ファイルを配信する。各端末はこのアプリケーション記述ファイルに沿って適切なソフトウェアを自動的にダウンロードする。

- ・認証モジュール

クライアントソフトウェアからの協調ユーザ追加リクエストを処理する。リクエストの発信元のIDと相手のIDからそれぞれの認証ユーザDBを検索し、認証関係が結ばれていなければ相手の携帯端末に特殊なURLが記載されたメールを送信する。相手がこのメールを確認および承認し、URLに接続すると認証モジュールがこれを感じ知して各認証ユーザDBにそれぞれのIDを追加する。また、クライアントソフトウェアから協調ユーザリスト取得リクエストがあった場合には、認証ユーザDBの内容を送信する。

- ・同期モジュール

クライアントソフトウェアの同期リクエストを受け取り、携帯端末内のデータとサーバ上のタスクDBを同期させる。通常の同期の場合、まず両者のタスク情報からタスクIDを検索し、どちらかに存在しないタスクがあればこれをコピーする。両者に存在してもタイムスタンプが異なる場合には新しいものを古いものに上書きする。最後に両者のタスク情報に付加された同期フラグを判別し、未同期であると見なされた場合にはすべてのパラメータをチェックし、問題があればこれを修正する。最後に全てのタスクの同期フラグを同期済みに設定する。完全同期のリクエストがあった場合には、同期フラグのチェックと同期および修正を行う。協調タスクもしくは依頼タスクが新たに登録されるかパラメータが変更されていた場合には協調タスクモジュールを呼び出す。

- ・協調タスクモジュール

同期モジュールより協調タスクもしくは依頼タスクの新規登録が伝えられた場合には、このタスクの内容を協調タスクDBに登録する。また、このタスクに関連するメンバー全員に承認するかどうかを決めるためのメッセージを送信する。このメッセージに対する反応があった場合、協調タスクDB内のパラメータを随時書き換え、全てのメンバーが承認すると依頼者に対して登録完了のメッセージを送信する。パラメータに変更があった場合にも同様の作業を繰り返す。

- ・データアクセスモジュール

クライアントソフトウェアの協調タスクビューにおいて他ユーザの情報閲覧リクエストが発せられると、認証ユーザ DB をチェックし、許可および不許可を判別する。許可された場合、データアクセスコントロールモジュールを呼び出す。同じくクライアントソフトウェアのユーザマネージャからユーザ情報取得リクエストを受けた場合には、タスク DB および協調タスク DB を検索し、必要な情報を送信する。また、データアクセスコントロールモジュールおよび PageRank モジュールを呼び出して必要な情報を受け取り、これをクライアントソフトウェアに送信する。

- ・データアクセスコントロールモジュール

データアクセスコントロールモジュールはあるユーザが他のユーザのタスク情報を閲覧してよいかどうかを判定する。まず、閲覧ユーザの ID と被閲覧ユーザの ID を受け取り、被閲覧ユーザのタスク DB から公開可能なタスク (PUB パラメータが ALL であるもの) を全て送信する。同時に非公開タスク (PUB パラメータが OFF であるもの) については拒否する。スマートフィルタリングを設定しているタスク (PUB パラメータが SMART であるもの) については、両者のグループ関係を判別する。これを行うために、協調タスク DB を検索し、全ユーザ×全ユーザの依頼関係マトリックスを作成する。このマトリックスに対して完全グラフ発見アルゴリズムを適用し、すべての完全グラフ (= グループ) についてそれを構成するユーザリストを作成する。次に、これらのグループリストの中から閲覧ユーザおよび被閲覧ユーザの両者が含まれるグループを抜き出す。そして、被閲覧ユーザのスマートフィルタリング適用タスクの中から MEMBR パラメータに含まれるユーザリストと先ほどのグループリストを比較し、グループリストのうちの最低 1 つがユーザリストを包含する場合に閲覧可能であるとしてそのタスクを送信する。

- ・PageRank モジュール

PageRank モジュールはデータアクセスモジュールからのリクエストにより全てのノードについてのリンクの濃度 (PageRank 値) を計算する。ここではデータアクセスコントロールモジュールと同様に全ユーザ×全ユーザの依頼関係マトリックスを作成し、次にこのマトリックスの固有ベクトルを求める。得られた固有ベクトルを正規化したものが PageRank 値として送信される。

4. テストおよび評価

本プロジェクトでは提案システムの実証実験を行った。期間は 2003 年 1 月 26 日から 2 月 17 日までの約 3 週間である。被験者は開発者および共同開発者を含めて 12 名である。実験内容は、期間中にクライアントソフトウェアを実際に利用し、被験者自身のタスクを入力してもらうというものである。そして、期間中に開発者が用意したアンケートに答えてもらう。なお、被験者のグループ構成は、A 研究所に属するのが 3 名、B 研究室に属するのが 2 名、C 研究室に属するのが 5 名、C 研究室の O B が 3 名、D 大学に所属しているのが 6 名、D 大学の O B が 4 名、それら以外に個人的な友人関係が含まれるなど、非常に複雑なものとなっている。

実験期間中に登録された個人タスクの平均は 41.9 件であった。また協調タスク・依頼タスクは計 76 件であった。この協調タスク・依頼タスクは平均 2.8 名に関係しており、個人タスクの 31.7% を占める。また、協調タスク入力支援機能により、1 名あたり 13.3 件のタスク入力の手間を軽減している。

アンケートでは被験者がこれまでに利用していたスケジューリング支援ツールと本システムを比較するという視点からの質問を多数行った。質問内容と主な回答を以

下に述べる。

- ・タスク入力数の増減

多くの被験者が従来のツールに入力していたタスクの数とほぼ同数か、それよりも多いと回答した。とくに、関係が密なグループでは協調タスクの数が多く、そのために入力し忘れていたタスクが他のユーザによって入力されているといった場合が多いとの回答を得た。また、PDA や手帳と比較して携帯端末を持ち歩いている時間が長い場合、規模の小さいタスクを入力することが増えたとの回答もあった。

- ・Social Scheduler の特徴

被験者には本システムのマニュアルを配布していたため、機能のあらましについては理解していたが、その中でも印象に残った機能をあげてもらった。とくに回答が多かったのはスケジュールデータの共有と閲覧、協調タスクの投入機能についてのものであった。上でも述べたように、入力を忘れていたタスクが他のユーザにとって補完されるという部分がこれまでにない機能であるとして好評を得た。また、情報量の多いインターフェイス、とくにカレンダービューについても好意的な意見が寄せられた。

- ・Social Scheduler に不足している機能

この質問では携帯電話にデフォルト設定として入っているカレンダーと比較した意見があると予想していたが、実際にこれらの機能を活用しているユーザがほとんどいなかったため、回答としては協調機能の一層の充実を求める意見が大半を占めた。具体的には、協調・依頼タスクにおけるタスク実行者リストの一覧性を高めることや、依頼関係によって構築されたグループがどのようなものであるかということが見えるような仕組みが求められていた。

- ・今後も利用する意思があるか

大半 (10 名) の被験者がこの質問に Yes と答えた。ただし、多くの友人が使うようになれば、という意見が多く、本システムがコミュニティ活動を支援するものとして利用されていることが明確になった。

5. 開発成果の特徴

本システムの成果を以下に述べる。

- ・民主的意思決定

本システムが対象としているのは会社組織のようにあらかじめ上下関係が定められたグループではない。このようなグループにおいては、上位の階層に属する人間が下位の階層の人間のスケジュールをある程度自由にコントロールすることが可能であるために、意思決定までのトータルコストが非常に低い。しかしながら、階層が明確でないようなグループでは、各構成員の上下関係がないために、一旦リソースの競合が発生するとそれを解決するための交渉や妥協が必要になる。この競合解消に非常にコストがかかるために、このようなグループ形態は営利企業等では採用されないことが多い。

本システムでは、グループの構成員全てがあらかじめ自身のリソース情報を公開することを前提としている。このような環境では、他人の時間リソースの状況を考慮した上でリソースの予約をすることが可能になるために、予期しないリソースの競合を引き起こすことが少なくなる。また、競合が実際に発生したとしても、交渉のサイクルが短くなると思われる。このモデルによって競合が全く発生しなくなることはないが、フラットな組織を維持するためのコストを大きく低減できると考えられる。

- ・ボトムアップに構築されるグループの同定

本システムでは、メンバーリストの作成や管理者の決定といったグループの定義をあらかじめ行う必要はない。ユーザは、他のユーザに対して協調・依頼タスクを発行

するという操作のみを行い、そこからの関係の発見やグループの同定はすべてサーバ側が自動的に行う。これにより、ボトムアップに構築されるグループや、タスクの発生とともに生まれるアドホックなグループに対しても適切な支援が可能になる。

- ・データ入力コストの低減

本システムでは、1人のクライアントソフトウェアから他の複数のユーザの持つタスクデータに直接タスク情報を書き込むことが可能である。他ユーザはこのタスク情報を自分で入力する必要がないため、トータルで見るとデータ入力のコストは大幅に低減する。携帯電話のアプリケーションを導入する上で阻害要因となるのはデータ入力の手間であることが多い。このような協調モデルには携帯電話用アプリケーションのデメリットを打ち消す効果があると考えられる。

- ・多様なタスクの見方を提供

従来の携帯電話用スケジューラは既存のPIM用のインターフェイスを踏襲しているものが多い。PCやPDAのような画像解像度の高いデバイスでは一度に多くの情報を表示することが可能であるため、このようなインターフェイスでも問題が生じにくい。しかしながら、携帯電話の小さなディスプレイ上ではカレンダーを表示するだけで画面が埋められてしまい、タスクの存在が認識しにくい。本システムのクライアントソフトウェアではカレンダーの日付ごとに時間軸を付加してタスクの存在を把握しやすいようにするなど、小さなディスプレイ上に多様な情報を載せることが可能である。また、協調タスクビューによって同じタスク情報を日付ごとのガントチャート（クロノグラム）として表示し直すことや、ユーザマネージャでの認証ユーザごとのタスク分類表示など、多様な情報の見せ方を提案している。ユーザはこれらの表示をもとに自らのタスク情報について新たな知見を得ることが可能になると思われる。

- ・スマートフィルタリング

本システムは、各ユーザの自発的な情報公開がなされることによってはじめて機能する。システムの有用性は情報公開へのインセンティブになりうるが、プライバシーの侵害や過剰な情報公開への危惧が大きな阻害要因となる。そこで、本システムではタスク情報のスマートフィルタリング機能を実装し、望まない情報公開や全ての認証ユーザに全ての情報を公開してしまうといった状況を回避することが可能である。実際には、タスクの依頼関係からグループを同定し、グループをまたぐような情報アクセスを禁ずる。これにより、プライバシーの侵害を最小限に抑えた上で、情報共有によるメリットを享受することが可能になる。

6. 今後の課題

今後の課題を以下に述べる。

- ・グループマネージャの実装

本システムで提案したグループ関係に基づくスマートフィルタリングは情報のアクセスコントロール手段として適切に動作することがわかった。今後はグループ同定を応用した機能を追加する必要があると考えている。とくに、得られたグループがどのような構造になっているのかをユーザ自身が知ることのできるような「グループマネージャ」が必要である。

- ・セキュア化

本システムはプロトタイプ段階であるためにタスクデータ等に関して暗号化などは行っていない。こういったデータはインターネットから直接アクセスできない領域に設置されているが、サーバ自体が攻撃を受けるような状況が起こった場合にはデータを保護することができない。本システムではユーザ同士の認証関係やタスクデー

タ等のプライベートな情報を扱っているため、非常の事態に対しても対応できるようなセキュリティをかける必要がある。

- ・複数サーバの連携

本システムは1サーバ・マルチクライアントの構成を取っている。全てのユーザのデータはサーバに集められるため、システムを導入するにあたってこの点が受け入れられない可能性がある。そこで、サーバを複数にし、それぞれのサーバが連携することで現在の利便性を確保するシステムが必要であると考えられる。また、サーバのインストールを簡便にすることも必要である。

- ・さらなる操作の簡素化

5章でも述べたように、フラットな組織内でのリソースの競合解消のためには交渉のステップが避けられない。本システムにおいてそのサイクルを短くすることが可能になったが、さらにその交渉過程をシンプルにする必要がある。そのためには、インターフェイスの改善や交渉モデルの簡素化が重要となる。

- ・大規模なテスト

今回のプロジェクトでは携帯端末10台程度でのテストを行った。開発用エミュレータ等も利用しているため実質的には15台程度であるが、より大規模な実証実験を行いグループ同定やスマートフィルタリングの性能を検証する必要がある。

- ・他キャリア端末への対応

本システムはNTTドコモのiモード端末のみに利用が限定されているが、すでに他キャリアの端末への対応の要望が多く出ている。アーキテクチャが異なるために移植は容易ではないが、ソースコードのオープン化などを視野に入れ、より多くの機種に対応したシステムを構築したいと考えている。

- ・企業との提携

半年に一度のペースで新しい機種が登場する携帯電話上のアプリケーション開発を個人ベースで継続するのは非常に困難である。単純なメンテナンスであっても画面解像度などの基本機能が向上した場合にはある程度の追従が必要不可欠である。そこで、本システムの開発を継続するにあたっては企業との提携が必要であると思われる。今後はキャリア、携帯端末メーカ、コンテンツ配信企業を問わず、アプローチを行うことを考えている。

7. 特記事項

本システムは2003年4月21日現在、<http://www-kasm.nii.ac.jp/ngrid/> においてテストサービスを提供している。携帯端末からこのアドレスに接続するとユーザ登録やダウンロードの方法に関する説明が表示される。また、ユーザ向けの簡易ドキュメントを<http://www-kasm.nii.ac.jp/ngrid/help.html> に掲示している。

8. 参加企業及び機関

本プロジェクトの遂行に際しては、プロジェクト管理組織として株式会社ジェイテックプレイヤーズの協力を得た。