

# GreenOffice プロトタイプシステム “FRESH” の機能\*

4 R - 1

森岡展朗 高橋正司 今村二康† 翁長健治‡

†NECソフトウェア中国 ‡広島大学

## 1 はじめに

近年、オフィスを取り巻く諸々の環境は著しく変化しており、それらの環境とオフィスとの軋轢が人間に与える影響は、心身両面において極めて深刻である。この現状に対処すべく、我々は快適性と生産性を共に向上させる GreenOffice を提唱し、次世代オフィスモデルの構築を目指している。FRESH(Flexible RESources for Human working) はそのプロトタイプシステムである。

現在、我々は FRESH の開発と試用を通じて、GreenOffice の利点と問題点を検証中である。FRESH は EWS および PC による LAN 上で動作するが、将来的には公衆回線などを介して遠隔地や自宅でも利用することも考えている。開発言語には主として C++ を用い、GUI 部分はユーザインタフェース構築環境「鼎」、およびそのインタフェースビルダである「ゆず」を利用して作成した。

本稿では、FRESH の機能のうち、主にオフィス業務の実行・管理をめぐる動作について述べる。なお文中、GreenOffice 固有の用語を断りなく用いることがあるが、詳しくは末尾に掲げる参考文献を参照していただきたい。

## 2 FRESH のシステム構成

FRESH を構成する主要クラスとその機能概要を以下に挙げる。これら各クラスのオブジェクトが協調動作を行うことにより、「時間・場所・スタイルのフレキシビリティ」を実現する。また、各オブジェクトの相互関係を図1に示す。

- **Application Launcher(AL)**  
ワークの定義・起動を行い、各クラスからの情報を受信・加工する。ワーカー1人につき1つ稼働する。個人環境内における全クラスの動作、およびユーザインタフェースの制御を担う。
- **Work Status Monitor(WSM)**  
ワークの状態管理を行い、ステータスの更新を随時、AL に通知する。ワークの起動に伴って生成される。
- **Communication Supporter(CS)**  
コミュニケーションパスを確立する。他のワーカーとの通信はこれを介す。コミュニケーションを交わしている双方のワーカーに1つずつ生成される。
- **Information Radiator(IR)**  
各クラスが発する情報をワーク場ごとに収集し、さらにそれを所属するワーカーにラジエーション情報として提供する。各ワーク場に1つ存在する。
- **Public Data Manager(PDM)**  
各ワーカーに共有されるデータを一元管理する。オフィスまたは企業に1つ存在する常駐型サーバ。

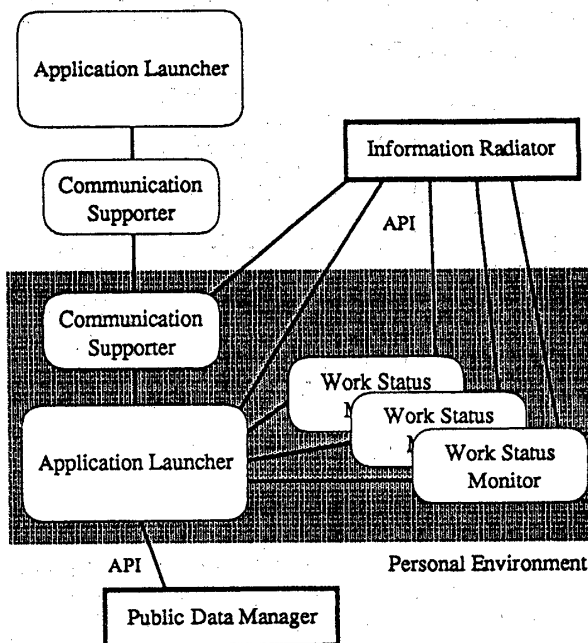


図1：システム構成

## 3 ワークの実行と管理

### 3.1 ワークの定義

起動されるワークは、その前に定義されている必要がある。FRESH においては、ワークは図2に示すような階層構造によって表現される。まず基幹となるワークがあり、それがいくつかのサブワークへ、そのサブワークがさらにまたいくつかのサブワークへと細分化されていき、最下層にはプリミティブワークが置かれる。この階層構造はワークの静的な分類とは異なり、ダイナミックな状態遷移に対応づけて記述したものである。よって図2における各ノードは、実はステータスを表現している。

プリミティブワークとは、ワークツール (WorkTool) を操作するだけで実行でき、それ以上は細分化できないワークを指す。ワークツールとはコンピュータ上の作業支援ツール群の総称であり、ワーカーはその中の特定のものをを使ってプリミティブワークを実行する。GreenOffice においては、文房具からソフトウェアまでを含む統括的な幅広い概念であるが、システムの中で表現するに際して上のように限定した。

ワークを起動する契機には、2つのケースが考えられる。まず1つは、指示や問い合わせなど、他のワーカーの依頼によって起動する場合であり、あらかじめ定義済みのワークが指定される、または自明的に選択されることがほとんどである。もう1つは、ワーカーが自発的に活動を開始したり、あるいは実行すべき目標だけが与えられ、その方法は指定されないような場合である。この場合、

\*Behavior of the GreenOffice prototype system "FRESH"

†Nobuaki MORIOKA, Masashi TAKAHASHI, Tsuguyasu IMAMURA (NEC Software Chugoku, Ltd.)

‡Kenji ONAGA (Hiroshima University)

ワーカ自身が適当なワークを捜し出すか、新規に定義しなければならない。

ワークを定義するには、ワーク定義ツールを用いる。PDMが管理しているワークの雛型の中から、目的に適ったものを拾い出し、自分固有のワークにカスタマイズする。カスタマイズされた定義情報は、各自のワーク定義ファイルに保存される。PDMのデータを利用することは、単にワーク定義の指標となるだけでなく、ワーカそれぞれの作業を連動させる際に必要な整合性を図るという役割を果たしており、また、むやみなフレキシビリティによる生産性の低下を抑制する効果もある。

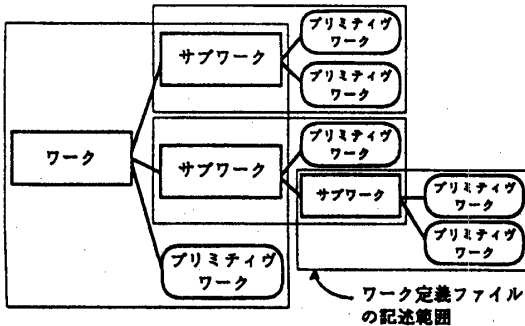


図2：ワークの階層構造

3.2 ワークの状態遷移

ワークを起動すると、必ずそれに伴ってWSMが生成される。WSMはまずワーク定義ファイルを走査し、そこに記述されている定義情報をALに返す。また、ワークのステータスを常に管理し、状態が変わればただちにALに通知する。したがって、ワーカはまず最初にワークの全体像を把握した上で、進捗度を随時確認しながらワークを行うことができる。

ワーカが定義・起動するワークには、自分自身で行う個人ワークだけでなく、他のワーカに実行を依頼するペアワークやチームワークも含まれるため、WSMは他のワーカのWSMとも通信を行う必要がある。ペアワークの状態遷移の1例を図3に示す。

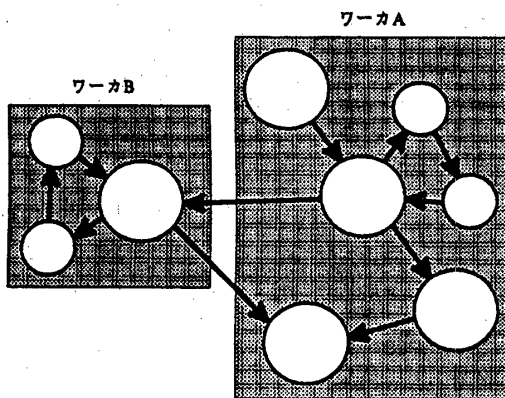


図3：ペアワークの状態遷移

ワーカAはCSを通じてワーカBにワークの起動を依頼する。このときBが起動したワークは、Aのワークから派生したサブワークと見なされる。Bのワークが進行している間、Aもまた並行して自分のワークを行う。そして、あるステータスまでワークが遷移すると、そこから先へ進むためには、Bのワークの完了が前提

となってくる。そこで、AとB双方のWSMが通信を行い、Bのワークのステータスを随時Aに伝えて、ワークのスムーズな進行をサポートしている。これにより、個人ワークだけでなく、協調ワークの定義・起動が可能となる。ただし、Aが定義するのは、Bに実行を依頼するサブワークまでである。そのサブワークがさらにどのようなサブワークを持って状態遷移するかは、あくまでBによって定義される。

4 FRESHのGUI

現在のFRESHのウィンドウイメージを図4に示す。Greennessの高いワークを行うには、少なくとも以下に挙げる3種の情報が、いつでもすぐに把握できる必要があると考え、表示枠を3つ用意した。

- 自分の関与するワークが今どのステータスにあるか。
- どのワークに関して誰とコミュニケーションパスが張られているか。
- ワーク場からいかなるラジエーション情報が流されているか。

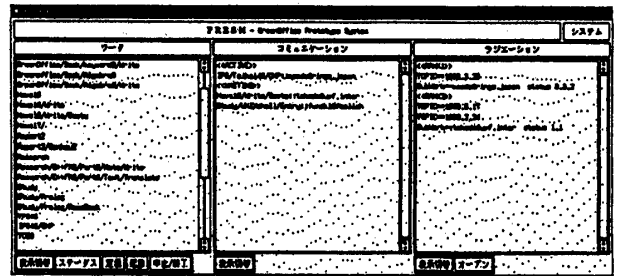


図4：ウィンドウイメージ

5 おわりに

現状では、ワーク定義ファイルへの記述はテキストエディタを用いているが、将来的にはワークの状態遷移の様子がグラフィカルに把握できるような仕組みにする予定である。さらに、個人ワークや協調ワークなどのワーク種別にも対応させたいと考えている。

また、出張移動や会議への出席など、いわば生身の作業をいかにシステムの中で表現するか、ワークを起動しながら定義していくほかないような状況、つまり、やってみなければどうなるか分からないような状況にはどう対処するか、といったことなども今後の大きな課題である。

参考文献

- [1] 翁長他「GreenOfficeの基本コンセプト」, 情報処理学会第45回全国大会講演論文集(6), 1992.
- [2] 今村他「GreenOfficeのOfficeware」, 情報処理学会第45回全国大会講演論文集(6), 1992.
- [3] 高橋他「GreenOfficeのアーキテクチャ」, 情報処理学会第45回全国大会講演論文集(6), 1992.
- [4] 森岡他「GreenOffice - プロトタイプ的设计思想 -」, 情報処理学会第45回全国大会講演論文集(6), 1992.
- [5] 片岡他「GreenOffice - プロトタイプの実装 -」, 情報処理学会第45回全国大会講演論文集(6), 1992.