

タスクメタファの定式化

5Q-3

小笠原 直人*, 佐藤 究*, 布川 博士**, 白鳥 則郎*

*東北大学大学院情報科学研究科

**宮城教育大学理科教育研究施設

1. はじめに

ネットワーク上に存在する様々なサービスを利用しつつ生産的活動をしなければならない現在の計算機利用環境において、こうした分散環境のための、ユーザにとって理解しやすいユーザインタフェースの必要性は高まっている。

我々の研究では現在分散環境を構成する系のうち

(1)人-ネットワーク-コンピュータ

(2)人-ネットワーク-人

の2つの系を対象とし、この分散環境をメタファーにより類推することで利用方法を獲得することが可能な認知的インタフェースの構築を目的としている。

現在まで、「人-ネットワーク-コンピュータ」の系においてサービスの機能概念、利用概念を構造写像を用いることにより表現し[3]、その実現システムとしてメタファーネットワークDoReMi[1][5]を構築してきた。またメタファーエキスパートユーザに対して[2]メンタルモデルの変化に応じてメタファーの構造等をカスタマイズできる環境、サービス記述環境を提供してきた。

しかし「人-ネットワーク-人」の系を考えると、コミュニケーションを行う場となるものに対応するメタファー（例：学校、会議室等）は共有されるべきものであり、上で述べたサービス記述環境ではメタファーの構造をカスタマイズした場合、コミュニケーションするユーザ間で見えるメタファーが異なるという現象が生じる。上の2つの系からなる分散環境においてメタファーエキスパートユーザにも適応可能なUIであるためには、共有されるメタファーの構造を変更せずに、メンタルモデルの変化に対応できる仕組みが必要である。

この問題を解決するため我々はタスクメタファーという概念を導入した。本稿では、このメタファーを提供するユーザインタフェースにおける抽象的なタスクの表現であるタスクメタファーの定式化を行う。

2. タスクメタファー

2.1 タスクメタファー

タスクメタファーとはユーザが行う行動（タスク）を目標領域としてメタファーをマッピングする提供手法である。

メタファーを用いて提供される分散環境上で、ユーザが行いたいタスクは、それを実行するためのメタファー世界での一連の操作の系列として表現される。

タスクメタファーではこの操作の系列をシステム上でのコマンドの系列へ変換し（このコマンドの系列をプロジェクトと呼ぶ）、プロジェクトにメタファーをマッピングしてユーザに提供する。

プロジェクトはユーザのタスクに対するメンタルイメージをシステム上での表現である。このプロジェクトにメタファーをマッピングしユーザに提供することにより、分散環境全体のメタファーを変更することなく、メタファーエキスパートユーザに適応するUIに必要な機能、具体的には

- (1)操作の煩雑さの解消
- (2)実用性を重視したメタファーの提供
- (3)個々のユーザのメンタルモデルの反映の実現が可能になる。

タスクメタファーは例えば「都市（分散環境全体を表現するメタファー）の中で、手紙をとってくる（タスクを表現するメタファー）ことを代理人（タスクの実行主体を表現するメタファー）」といった形でメタファーの世界で表現される。

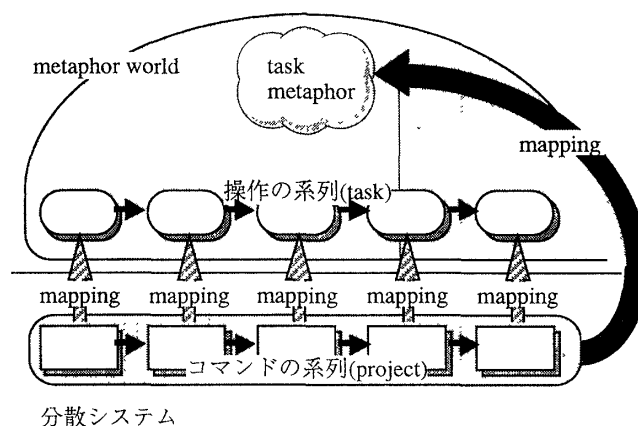


図1：タスクメタファー

Formalize of Task metaphor

Naohito Ogasawara*, Kiwamu Sato*,

Hiroshi Nunokawa**, Norio Shiratori*

* Graduate School of Information Sciences, Tohoku University

**Research Institute of Science Education, Miyagi University of Education

2.2 タスクメタファーでのタスク

上で述べたタスクメタファーを定式化するためにまずタスクについて分析する。タスクは次のような特徴を持つ。

- ・タスクは、初期状態から目標状態への状態遷移で表現される。ユーザは各状態において状況を認識し、システムに対し何らかの操作をし次の状態に遷移するという行動を繰り返し目標状態へと遷移する。
- ・一つのタスクはいくつかのサブタスクからなる場合がある。この場合タスクはいくつかのサブタスクを合成したものとして表される。
- ・一連の操作の系列であるタスクはそれを抽象化し、別のタスク中の一操作として用いられる場合がある。

タスクメタファーでは上のタスクの特徴を表現できる能力が必要である。次章においてタスクメタファーの定式化について述べる。

3. タスクメタファーの定式化

タスクメタファーでの一局面における操作は次のように表される。

$(p) \langle \text{stage}(a, c) \rangle \rightarrow (p, r) \langle \phi \rangle$

$a \in \text{action}$: メタファー世界においてタスクを達成するために必要な行動

$c \in \text{condition}$: アクションの実行条件

$p \in \text{parameter}$: アクションの実行に必要な引数

この操作を実行すると、得られる結果はパラメータのリストに加えられる。

$(p) \langle \text{stage}(a, c) \rangle \rightarrow (p, r) \langle \phi \rangle$

$r \in \text{result}$: 実行結果

タスクは操作の系列として次のように表される。

$\text{task} = (p_1=x_1, p_2=x_2, \dots, p_i=x_i) \langle \text{stage}_1(a_1, c_1), \text{stage}_2(a_2, c_2), \dots, \text{stage}_n(a_n, c_n) \rangle \quad (i \leq n)$

このタスクがプロジェクトに変換され、メタファーがマッピングされる。

・タスクの合成

タスクは次のようにして合成される。

$\text{task}_1 = (p_1=x_1) \langle \text{stage}_1(a_1, c_1) \rangle$

$\text{task}_2 = (p'_1=x'_1) \langle \text{stage}_1(a'_1, c'_1) \rangle$

$\text{task}_1 \cdot \text{task}_2 = \text{task}_3 = (p_1=x_1 \{ (p'_1=x'_1) \langle \text{stage}_1(a'_1, c'_1) \rangle \}) \langle \text{stage}_1(a_1, c_1) \rangle \rangle$

$\text{task}_1, \text{task}_2$ から合成された task_3 は次のように実行される。

$\text{task}_3 = (p_1=x_1 \{ (p'_1=x'_1) \langle \text{stage}_1(a'_1, c'_1) \rangle \}) \langle \text{stage}_1(a_1, c_1) \rangle$

$\rightarrow (p'_1=x'_1 \{ (p_1=x_1, r_1=y_1) \langle \phi \rangle \}) \langle \text{stage}_1(a'_1, c'_1) \rangle$

$\rightarrow (p'_1=x'_1, p_1=x_1, r_1=y_1) \langle \text{stage}_1(a'_1, c'_1) \rangle$

$\rightarrow (p'_1=x'_1, p_1=x_1, r_1=y_1, r'_1=y'_1) \langle \phi \rangle$

task_2 に必要なパラメータが task_1 の実行結果に依存している場合、 task_1 の実行結果がパラメータのリストに加えられるためパラメータの引き渡しが行われる。

・タスクの抽象化

また、タスクの中のアクションとしてタスクを指定することにより、より抽象的なタスクを記述することが可能である。

アクションとして用いるタスク

$\text{task}_1 = (p_1=x_1) \langle \text{stage}_1(a_1, c_1) \rangle$

アクションとしてタスクを持つタスク

$\text{task}_4 = (p_2=x_2) \langle \text{stage}_2(a_2, c_2), \{ (p_1=x_1) \langle \text{stage}_1(a_1, c_1) \rangle \} \rangle$

このタスクは次のように実行される。

$\text{task}_4 = (p_2=x_2) \langle \text{stage}_2(a_2, c_2), \{ (p_1=x_1) \langle \text{stage}_1(a_1, c_1) \rangle \} \rangle$

$\rightarrow (p_2=x_2, r_2=y_2) \langle \{ (p_1=x_1) \langle \text{stage}_1(a_1, c_1) \rangle \} \rangle$

$\rightarrow (p_2=x_2, r_2=y_2) \langle (p_1=x_1, r_1=y_1) \langle \phi \rangle \rangle$

$\rightarrow (p_2=x_2, r_2=y_2, p_1=x_1, r_1=y_1) \langle \phi \rangle$

4. まとめ

本稿ではこのメタファーを提供するユーザインタフェースにおける抽象的なタスクの表現であるタスクメタファーの定式化について述べた。

参考文献

- [1] 小笠原直人, 佐藤究, 布川博士, 白鳥則郎: メタファーネットワークにおけるサービス記述環境の構築, 情報処理学会マルチメディア通信と分散処理研究会報告 (1995) 95-DPS-70-28, pp161~166
- [2] 小笠原直人, 佐藤究, 布川博士, 白鳥則郎: メタファーネットワークDoReMiにおけるユーザプロジェクト記述モデル, 情報処理学会第52回講演論文集 (1996), pp6-185~186
- [3] 佐藤究, 布川博士, 楠見孝, 白鳥則郎, 野口正一: 分散システムのためのユーザインタフェースメタファー, 電子情報通信学会論文誌 Vol. J79-A No.2 (1996)
- [4] 佐藤究, 布川博士, 小笠原直人, 白鳥則郎: ユーザに適應するユーザインタフェースメタファー, 電子情報通信学会論文誌B-I Vol. J79-B-I No.5 pp.353-363 (1996)
- [5] 小笠原直人, 佐藤究, 布川博士, 白鳥則郎: メタファーネットワークDoReMiにおけるタスクメタファー, 情報処理学会マルチメディア通信と分散処理ワークショップ発表論文誌 pp209~214