

## 分散開発環境RODSにおけるユーザ・インターフェース

4N-9

春木篤幸 石井浩一 松山実 横井利彰

武蔵工業大学

## 1. はじめに

近年、ネットワーク環境が普及しているが、ハードウェアの違いなどにより、この環境を効率的に利用するソフトウェアの開発は容易ではない。こうした現状を踏まえて、現在、我々はCASEを基に、ネットワーク環境で動作するソフトウェアの開発を支援するシステムRODS<sup>[1][2]</sup>(Repetitive Optimum Dividing System)の開発を行っている。本報告ではRODSのソフトウェア記述用エディタNDFDE<sup>[3]</sup>(Network Data Flow Diagram Editor)において、特に、ネットワーク情報とアルゴリズムの関連付けを行う部分について述べ、その部分のユーザ・インターフェースについて考察を行う。

## 2. ネットワーク情報

RODSではアルゴリズムとは別に、ネットワーク情報を次の2つのリソースに分けて扱う。

## (1) ベースリソース

ネットワークで使用可能な全てのマシンの情報等、全てのプログラムで共用可能な情報

## (2) ノードリソース

各プログラム独自のネットワークの利用方法等、各アルゴリズムに密接した情報

このようなリソースの分類により、ベースリソースが様々なアルゴリズムで共用でき、ベースリソースを基にして、ノードリソースの作成が容易に行える。

## 3. リソースエディット機能

RODSではNDFD(Network Data Flow Diagram)という独自のデータ・フロー・ダイアグラムに基づき、プログラムを記述する。このNDFDの特徴は以下のとおりである。

- ・ネットワーク環境用に拡張されている
- ・表記方法に明確な規定がある
- ・フローチャート等、他の方法で制御の流れを記述することなしにコード生成を行うことができる

NDFDは、プログラムのアルゴリズムの部分のみを記述するのに使われ、各処理系独自の情報等、リソースは別形式で与えられる。このため次のようなメリットが得られる。

- ・プログラム記述の段階で処理の分割を意識しなくてもよい

- ・NDFDで作成したプログラムはリソースを変更することにより様々な処理系で使用することができる

- ・リソースの一部であるネットワーク情報を様々なプログラムで共用することができる

しかし、このように完全にアルゴリズムとリソースを分離したことにより、ノードリソースの指定が面倒なものになった。このため、NDFDの編集を行う段階でノードリソースの編集を行えるようにした。この機能により、NDFDを編集している段階では、ノードリソースとアルゴリズムを混在した形で把握することができ、保存

---

Development of NDFDE(Network Data Flow Diagram Editor)

Atsuyuki HARUKI, Kouichi ISHII, Minoru MATSUYAMA, Toshiaki YOKOI

Musashi Institute of Technology

の際にはこの2つを分離して出力することが可能となった。つまり、リソースとアルゴリズムを分離するメリットを生かしながら、問題点を改善することができた。

このリソース編集の核となる部分を図1に示す。

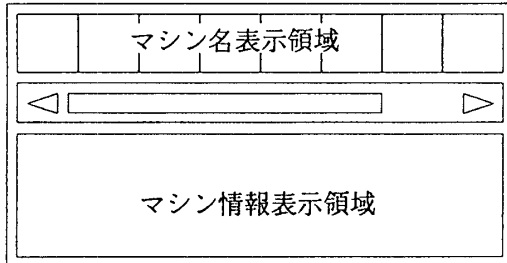


図1. リソース表示領域

実際の編集方法は、ノードをマシン名表示領域の任意の位置にドラッグすることにより編集を行う。

#### 4. ユーザ・インターフェスの改善

NDFDEはNDFDをコンピュータ上で編集するためのエディタである。NDFDEはNDFDをGUIを用いて取り扱うため、X-Window上でMotifを利用して作成されている。現在の画面構成を図2に示す。

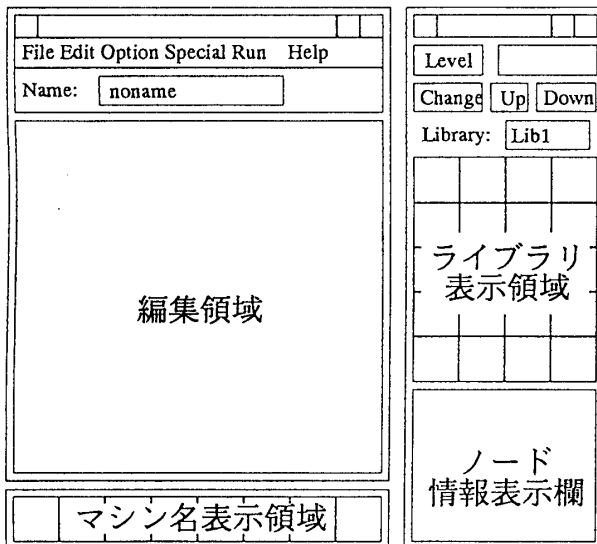


図2. NDFDEの画面構成

今回、ユーザ・インターフェスの変更を行った部分を次にあげる。

##### (1) 編集領域の自由度の向上

##### (2) リソース編集領域の付加

これらの(1)については、具体的にはノードの置ける位置に自由度をもたせオーバーラップを可能とし、データフローを表す矢印の描画をより簡単に行えるようにした。これによりアクターモデル等への対応が容易になり、またダイヤグラムの表現力が向上した。操作面では、ドラッグやショートカットキー等を採用し、より操作の自由度を増やした。さらに、こうした変更に伴って、使用する資源を90%以上節約し、倍以上のレスポンスが得られた。

#### 5. おわりに

ネットワーク情報をリソースの形で提供し、アルゴリズムと完全に分離することによりプログラミングにおいて様々なメリットが得られる。特にベースリソースとアルゴリズムの再利用が容易になり、ノードリソースもGUIの支援により、簡単に編集できるというメリットが大きい。

今後は、実際に何人かの人にこのエディタを使用してもらい、その意見を参考にしながら、より実用的なものにしていき、同時にそのユーザ・インターフェスについて研究を進めていく方針である。

#### 参考文献

- [1] 山下利夫, 松山実, 横井利彰, “ソフトウェア自動分割システムRODSの概要”, 情報処理学会第45回全国大会講演論文集(分冊5), pp. 319-320, 1992
- [2] 山下利夫, 松山実, 横井利彰, “自動分割システムRODSにおけるNDFD処理系について”, 情報処理学会第46回全国大会講演論文集(分冊5), pp. 261-262, 1993
- [3] 春木篤幸, 山下利夫, 松山実, “ネットワーク環境におけるユーザ・インターフェスの研究”, 情報処理学会第48回全国大会講演論文集(分冊5), pp. 205-206, 1994