

自律協調分散システム Noah における協調機構

6F-8

宮崎 一哉, 小塚 宏, 佐藤 文明, 福岡 久雄

三菱電機(株) 情報システム研究所

1 はじめに

多数の計算機がネットワークを介して接続された分散環境において、システムの運用・管理を柔軟かつ容易にすること、サービスを高信頼化し容易に利用できるようにすること、分散アプリケーションの構築を容易にすることを旨とし、分散システム構築基盤として Noah (Network oriented applications harmony) を開発している [1]。Noah は分散環境において自律的な計算主体 (エージェントと呼ぶ) が協調しながら処理を進めていく『自律協調分散技術』に基づいており、タブルスペース型の通信部、エージェントの監視 / 制御部、そして協調処理部から構成される。

協調処理部は、分散環境においてサービスの要素を提供する自律的なエージェントが協調動作することによりエンドユーザが要求するサービスを遂行するようなアプリケーションソフトウェアを開発するためのツールと実行モジュールや環境を提供する。

本稿では、Noah の協調処理部の概要を述べる。

2 フィールドとエージェントのモデル

大規模な分散環境において資源を管理したり、適切なサービスを利用したり、また分散の特長を活かした高信頼なアプリケーションを構築することは非常に難しい。Noah ではエージェントとフィールドに基づく独自のシステム構成モデルを採用することによってこの問題に対処している。Noah が採用するフィールドとエージェントからなるシステムの基本構成モデルの概念図を図 1 に示す。

エージェントは分散環境上のサービスの要素や資源を提供する計算主体である。エージェントはフィールド内に配置される。エージェント間の情報交換は基本的にメッセージプールを介したメッセージ交換によって行われる。メッセージプールを与えるのはフィールドである。メッセージがメッセージプールに投入されると、そのフィールドに所属するエージェントが投入されたメッセージを読んだり取り出したりすることができる。情報交換

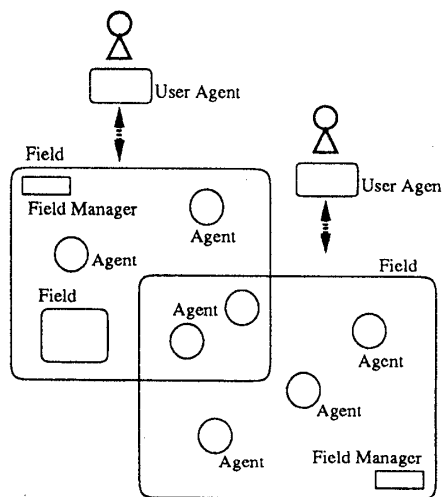


図 1: フィールドとエージェントの基本構成モデル

の仕方としてはこのようなマルチキャスト通信によるほかに、受け手となるエージェントを直接指定してメッセージを送付することもできる。

フィールドはフィールド同士を階層化したり複数のフィールドを重ねる重なりを持たせて設定することができる。フィールドは内部に所属するエージェント間の通信を媒介するメッセージプールとして働くほか、次に挙げる役割を持つ。

- 複数のエージェントをグループ化し、階層構造の上位のフィールドに対してエージェントとして振舞う
- グループ内部のエージェントに対してポリシーを規定する

前者は Object-Field モデル [2] が提唱する場の性質と同様である。後者は従来のモデルには見られない Noah の特徴である。

ポリシーは、サービスの要素や資源の集合、サービス要素の組み合わせであるアプリケーション、あるいはそれらが導入される組織等の特徴づけるものである。ポリシーがエージェント間の連携の仕方、動作するエージェントの選択基準やエージェントの行動決定基準、資源へのアクセスの権限などを規定したり影響を与えたりする。フィールドによって組織やアプリケーションに独自のポリシーを設定することができる。

Cooperation Mechanism for an Autonomous Decentralized System : Noah

Kazuya MIYAZAKI, Hiroshi KOZUKA, Fumiaki SATO, Hisao FUKUOKA

Mitsubishi Electric Corp.

ポリシーの定義を持ちそれを管理するのはフィールド毎に存在するフィールドマネージャである。協調の手順の定義とその定義に基づいた協調動作の指示はフィールドマネージャに任せられ、エージェントは詳細な協調手順の定義を持たない。このため、どのフィールドに所属するか、あるいはどのフィールドにメッセージが置かれたかによって、同一のエージェントが異なった協調動作や判断を行うことになる。同一のエージェントが他のフィールドに移動することによって異なった協調動作、異なった行動決定のための評価を行なうことになる。細かな協調の定義をエージェントに埋めこむ必要がなくなるため、個々のエージェントの定義は簡潔になり、アプリケーションや組織に跨ったエージェントの再利用がしやすくなる。

3 基本動作

ユーザがユーザエージェントを介して要求をフィールドに投入する。ユーザエージェントは Noah が提供する環境とユーザとの間のインターフェースである。ユーザエージェントはユーザの特性を格納したデータベースを持ち、これを参照してユーザから与えられた要求をユーザの特性を反映した形でメッセージを生成しそれをフィールドに投入する。

メッセージがフィールドに投入される際にフィールドマネージャは自己の管理するポリシーにしたがって協調動作を指示する情報(協調情報)をメッセージに付加する。フィールドに含まれるエージェントはこの協調情報を伴ったメッセージを読み取る(図2)。エージェントは基本的にメッセージの取得、評価、行動決定、実行を行なうが、フィールドマネージャによって付加された協調情報により、メッセージ内に含まれる要求をすぐに処理して良いのか、処理をするための条件を返せば良いのか、あるいは要求をどのように評価したら良いのかなどをエージェントが知り、その内容によってエージェントが自己の振る舞いを調整し、その結果をフィールドに返す。フィールドマネージャは返された結果をポリシーに従い評価し、次の動作を決定する。

このようにメッセージが投入されるフィールドによって付加される協調情報が異なるので、同一のエージェントでもフィールドのポリシーに適応的に振る舞うことができる。

4 実装

個々のフィールドマネージャやエージェントはそれぞれ独立したプロセスとしてノード上に分散配置され、やはりノード上に分散配置された対等な関係にある複数の Noah の通信サーバ(TS マネージャ)を介して情報のやり取りを行う。フィールドマネージャにはフィールドマ

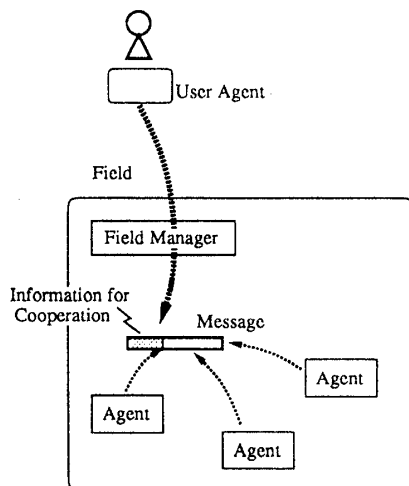


図 2: 基本動作

ネージャのテンプレートに対してスクリプト言語で記述されたポリシー定義が与えられる。また、エージェントにはエージェントのテンプレートに、自己の仕様を記述したプロファイル、行動選択のための評価基準、サービス定義などが与えられる。フィールド及びエージェントは与えられた定義を解釈実行するインタプリタとして実現される。フィールドへのエージェントの所属はフィールドマネージャが持つメンバリストで管理される。フィールドへのエージェントの出入りはこのメンバリストに対するアクセスで制御される。

5 おわりに

Noah ではフィールドをポリシーを規定する枠組みとして実現した。エージェントは所属するフィールドによって、あるいはメッセージがおかれたフィールドによって異なった振る舞いを示す。現在のバージョンにおいては、ポリシーの設定、特に評価基準の設定がやや難しい。スクリプトなどをより洗練することによりこの問題の解決を図る予定である。また、エージェントをプロファイルを参照することにより、ある程度自動的に適当なフィールドに配置することにより、システム管理を支援することも必要であると考えている。今後、いくつかの応用問題に適用のうえ評価し、機能の拡充を図る予定である。

参考文献

- [1] H. Kozuka et al. "An Architecture of Distributed Computing Environment — Noah: Network Oriented Applications Harmony," Proc. IPSJ 58th DPS (1993).
- [2] 西尾郁彦, ほか, "オブジェクトと場に基づいた協調的プログラム言語", 情報処理学会論文誌, Vol. 34, No. 12, pp.2499-2508(1993).