

広域分散エージェントリポジトリの管理運用支援

加藤 義隆†

打矢 隆弘‡

内匠 逸‡

†名古屋工業大学 工学部 情報工学科

‡名古屋工業大学 大学院 工学研究科

〒446-8555 愛知県 名古屋市 昭和区 御器所町

〒446-8555 愛知県 名古屋市 昭和区 御器所町

1 はじめに

マルチエージェントシステムでは、複数のエージェントを連携させることによって問題を解決する。マルチエージェントシステムを効率的に開発・運用するために、エージェントフレームワークと呼ばれる枠組みが用いられる。エージェント同士が連携を行うために、DASH[1]などのエージェントフレームワークでは、ネームサーバというエージェントを検索・識別するための機能がある。しかし、現状ではネームサーバが各サブネットワークごとに設置されており、サブネットワーク外のエージェントを利用できない。本研究ではネームサーバから情報を取得し一元化することにより、別ネットワークに存在するエージェントの利用を支援する機構を提案する。

2 DASH

DASHはエージェントを保持するサーバであるリポジトリを持つエージェントフレームワークである。DASHにおけるリポジトリとネームサーバの機能について説明する。

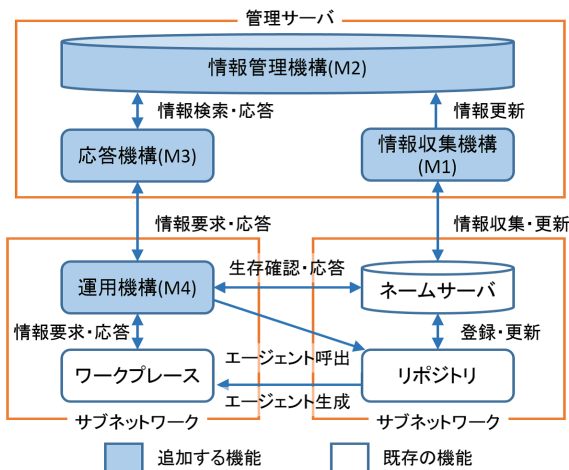


図 1: 提案機構全体図

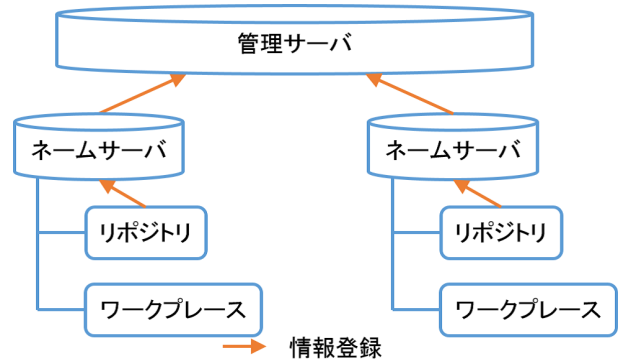


図 2: ネームサーバからの情報取得図

2.1 ネームサーバ

ネームサーバはサブネットワークごとに設置され、ネームサーバに接続されているワークスペースとリポジトリの一覧や、そこで動作するエージェントの一覧を作成する。エージェントは、ネームサーバの作成した一覧を用いて、他の動作中のエージェントを検索し、メッセージを送り協調を行う。

2.2 リポジトリ

リポジトリには、開発されたエージェントが動作可能な状態で保存されている。リポジトリはユーザのサービス要求に応じて、マルチエージェントシステムを動的に組織構成してワークスペースに生成し、サービスを提供する。

2.3 ワークスペース

ワークスペースでは、リポジトリから生成されたエージェントが動作し、ユーザへサービス提供を行う。エージェントを動作させるための環境であり、データを処理するために Java のプログラムを呼出可能である。

3 提案機構

提案機構では、各開発者が独自に開発しているエージェントを共有することで、利用できるエージェントを増やすことが可能になる。これにより、マルチエージェントシステムで解決できる問題の種類が増加する。

Operations Support for Wide-area Distributed Agent Repositories
 †Yoshitaka KATO ‡Takahiro UCHIYA ‡Ichi TAKUMI
 †School of Engineering, Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya, Aichi, 466-8555 Japan
 ‡Graduate School of Engineering, Nagoya Institute of Technology Gokiso-cho, Showa-ku, Nagoya, Aichi, 466-8555 Japan

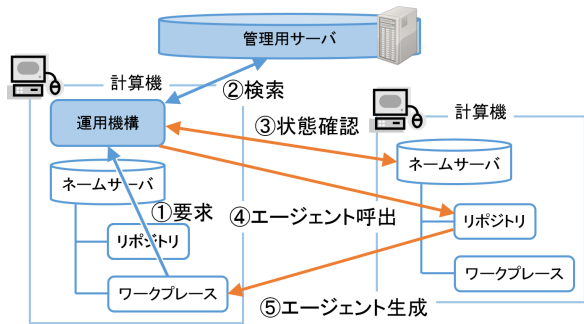


図 3: エージェント呼出の流れ図

全ネームサーバに登録されたエージェントを検索する機能を、エージェントフレームワークに用意することで、開発者が機能を用意せず利用できる。共通の機構を使用することでエージェントの蓄積を一元化する。

3.1 提案機構の設計

提案機構は以下の内部機構で構成される (図 1)。

- (M1) 情報収集機構 負荷分散のためにネームサーバを仲介してリポジトリの情報を収集 (図 2)
- (M2) 情報管理機構 情報の保存・削除・検索
- (M3) 応答機構 DASH エージェントおよび (M4) からの問い合わせを受信し検索結果を返信
- (M4) 運用機構 (M3) からの返答に応じてワークスペースにマルチエージェントシステムを生成

3.2 提案機構の動作

提案機構を利用したエージェントの登録・検索・呼出の動作を述べる。

3.2.1 エージェントの登録

ネームサーバを経由して、エージェントの情報を提案機構の管理サーバに登録する (図 2)。

1. リポジトリ起動時にネームサーバへエージェント情報の登録
2. ネームサーバから定期的に (M1) へ情報を送信
3. (M1) が (M2) の情報を更新

3.2.2 エージェントの検索

エージェントの名前や機能名などに対して文字列比較を行い、条件を満たすエージェントを検索する。

1. (M4) が (M3) にエージェントの検索を要求

2. (M3) が要求を受けて (M2) を検索
3. (M3) が検索結果を (M4) に返信

3.2.3 エージェントの呼出

提案機構を用いてエージェントを呼び出す。検索されたエージェントが存在するリポジトリに対して、エージェントの生成要求を送信する (図 3)。

1. (M4) が検索によってエージェント情報を取得
2. (M4) が返答されたりポジトリの状態を確認
3. (M4) がリポジトリにエージェント生成を要求
4. リポジトリがワークスペースにエージェントを生成

4 提案機構の試作

情報管理機構 (M2) と応答機構 (M3)、および運用機構 (M4) の試作を行った。次に、エージェントの検索機能について動作検証を行った。

情報管理機構の起動時に、テスト用エージェントデータを生成し、同機構内に保存した。次に、運用機構から応答機構に検索リクエストを送信し、応答結果を入力するテストを行った。結果は、リクエストで指定した条件に、該当するエージェントのデータが正しく出力された (図 4)。

```
c:\test>java -cp OMDR.jar TopClient
send request:
[DRSearchRequest(name,List(1)), DRSearchRequest(funcs,List(1, func))]

result:
AgentMetaData(Sampleagent1,List(func1, func2),test)
AgentMetaData(Agent1,List(1, func2),コメント)
```

図 4: 検索機能の実行テスト図

5 まとめ

各ネームサーバに接続しているリポジトリに対して、エージェントの情報を一元化し、管理運用を行う機構を提案した。今後は提案機構の実装を行い、動作検証及び性能評価を行う。

参考文献

[1] 打矢 隆弘, 原 英樹, 高垣 暁, 菅原 研次, 木下 哲夫, “リポジトリ型エージェントフレームワークの開発と評価”, 情報技術レターズ, Vol.2, pp.139-141, 2003.