

Padrões Arquiteturais e de Projeto para a Modelagem de Usuários baseada em Agentes

Ismênia Ribeiro de Oliveira, Rosario Girardi

Universidade Federal Maranhão (UFMA) – GESEC/DEINF
Av. dos Portugueses, s/n, Campus do Bacanga – São Luís – MA – Brasil
ism_oliveira@yahoo.com.br, rgirardi@deinf.ufma.br

Abstract

Most software systems do not have the ability to satisfy heterogeneous user needs. Each user has particular knowledge level, needs, abilities and preferences. In that context, there is a need for building systems adaptable to each user type or group of users with common characteristics. Because of its advantages to approach software complexity, the agent-based development paradigm is being used for the development of applications that can adapt to different types of users. Pattern reuse in this development paradigm makes possible an optimization in the time and cost of development and an increase in the quality of software. This article approaches user modeling as a mean of producing adaptation effects in a system and proposes solutions for frequent user modeling problems in the form of agent-based architectural and detailed software patterns.

Resumo

A maioria dos sistemas de software não tem a habilidade para satisfazer as necessidades heterogêneas dos seus usuários. Cada usuário tem níveis de conhecimento, necessidades, habilidades e preferências bastante variadas. Nesse contexto surge a necessidade de construir sistemas que se adaptem a cada tipo de usuário ou grupo de usuários com características comuns. Por causa de suas vantagens para abordar a complexidade do software, o paradigma de desenvolvimento de software orientado a agentes está sendo utilizado para o desenvolvimento de aplicações que se adaptam a diferentes tipos de usuários. A reutilização de padrões no desenvolvimento de sistemas multiagente possibilita uma otimização no tempo e no custo do desenvolvimento e um aumento na qualidade do software. Este artigo aborda a modelagem de usuários como forma de produzir efeitos de adaptação em um sistema e propõe soluções recorrentes para problemas recorrentes de projetos de modelagem de usuários na forma de padrões arquiteturais e de projeto baseados em agentes.

1. Introdução

É possível adaptar sistemas de acordo com o conhecimento ou experiência do usuário, histórico de ações anteriores, propriedades cognitivas (estilo de aprendizado, personalidade), objetivos e planos (intenções) do usuário, seus interesses e preferências.

Um importante componente dos sistemas interativos adaptáveis é a habilidade para modelar os usuários do sistema. A modelagem de usuários é essencial em sistemas que tentam adaptar seu comportamento aos usuários para interagir de forma mais inteligente e individualizada.

Um grande desafio encontrado na modelagem de usuários é tratar a heterogeneidade dos usuários de software. Uma solução para este problema seria projetar e implementar a modelagem dos usuários e a adaptabilidade dos sistemas através de utilização da abordagem de agentes. Seriam usados agentes especializados e

individualizados para cada usuário ou grupos de usuários, como agentes de interface, agentes de modelagem e agentes de adaptação.

Um agente é uma entidade autônoma, que possui um sistema interno de tomada de decisões, agindo sobre o mundo e sobre outros agentes que o rodeiam e por fim, que é capaz de funcionar sem a necessidade de algo ou alguém para guiá-lo. A analogia feita com agentes no mundo real nos leva a conceituar um agente como uma entidade ativa, sempre ao lado do usuário e que possui conhecimentos específicos sobre um determinado domínio. De posse de bases de conhecimento e de mecanismos de raciocínio, os agentes devem ser capazes de reconhecer situações em que devem se ativar, sem que o usuário perceba, ou seja, de forma transparente ao usuário [32].

Segundo Wooldridge e Jennings [32], a construção e a especificação da estrutura e funcionamento de um agente genérico pode ser realizada segundo três tipos de arquiteturas:

- **Arquiteturas deliberativas:** segue a abordagem clássica da Inteligência Artificial, onde os agentes contêm um modelo simbólico do mundo, explicitamente representado, e cujas decisões (ações) são tomadas via raciocínio lógico, baseadas em casamento de padrões e manipulações simbólicas;
- **Arquiteturas reativas:** a arquitetura reativa é aquela que não inclui nenhum tipo de modelo central e simbólico do mundo e não utiliza raciocínio complexo e simbólico. Baseia-se na proposta de que um agente pode desenvolver inteligência a partir de interações com seu ambiente, não necessitando de um modelo pré-estabelecido;
- **Arquiteturas híbridas:** a arquitetura híbrida mistura componentes das arquiteturas deliberativas e reativas com o objetivo de torná-la mais adequada e funcional para a construção de agentes.

Essas considerações feitas sobre agentes nos levam a crer que a modelagem de usuários seria melhorada e facilitada caso fosse implementada com agentes inteligentes, além da melhora na eficiência e eficácia do sistema, através dos mecanismos de adaptação.

Este artigo aborda a modelagem de usuário como forma de produzir efeitos de adaptação em um sistema. A partir desse estudo são propostas soluções recorrentes para problemas recorrentes de projetos de modelagem de usuários, ou seja, padrões arquiteturais e de projeto especializados na modelagem de usuário e adaptação de sistemas.

A **Figura 1** mostra, em uma rede semântica, o relacionamento entre os padrões para esses sistemas. O padrão *Sociedade multiagente para sistemas adaptativos* justifica a escolha da abordagem de agentes para sistemas adaptativos. A organização dos agentes no sistema multiagente é descrita pelo padrão arquitetural *Camadas multiagente para sistemas adaptativos/adaptáveis* que distribui os agentes em camadas conforme as tarefas que serão executadas pelos mesmos. O detalhamento dos agentes que compõem as camadas é descrito pelos padrões *Interface, Modelagem e Adaptação*. Os padrões *Modelagem e Adaptação*, por sua vez, são derivados do padrão *Deliberativo*. Já o padrão *Interface* é derivado do padrão *Reativo*.

Neste artigo são apresentados os padrões: *Sociedade multiagente para sistemas adaptativos, Camadas multiagente para sistemas adaptativos, Deliberativo, Reativo e Modelagem*. A Tabela 1 apresenta, de forma sucinta os padrões propostos, classificando-os segundo a dependência ou independência do domínio da aplicação e segundo o tipo de padrão.

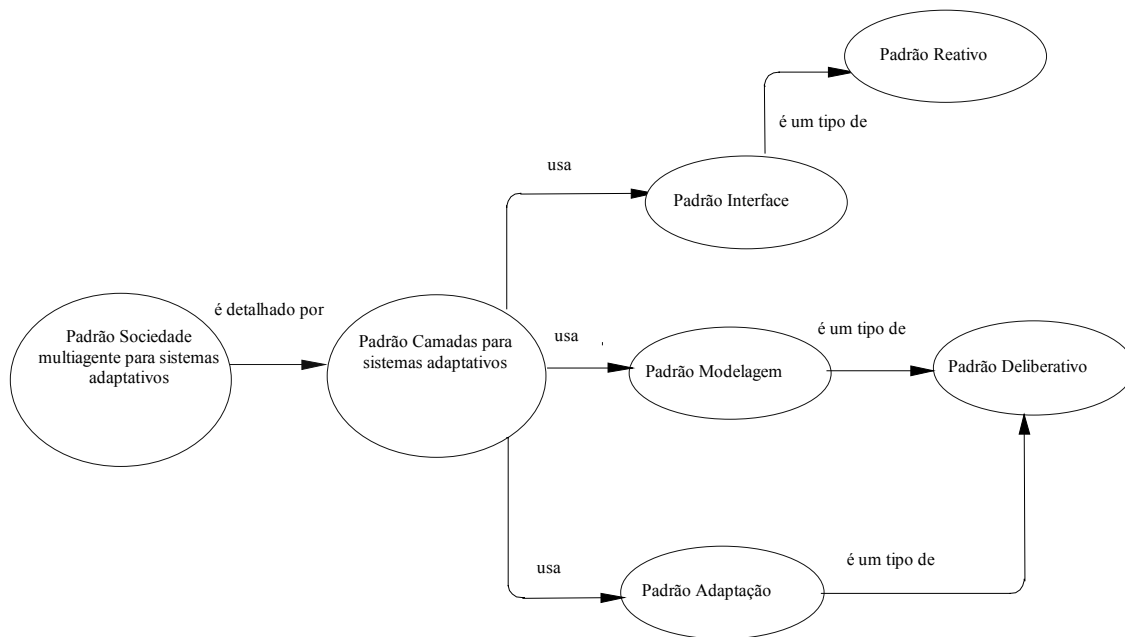


Figura 1 Rede semântica mostrando o relacionamento entre os padrões propostos

Tabela 1 Padrões para o projeto de sistemas adaptativos/adaptáveis

Padrões baseados em Agentes			
Classificação	Problema	Solução	Padrão
Padrões Dependentes de Domínio	Padrão Conceitual	Como estruturar uma comunidade de agentes para construir modelos de usuários e se adaptar a diferentes tipos de usuários ou grupos de usuários?	Camadas Multiagente para Sistemas Adaptativos/Adaptáveis
	Padrão Arquitetural	Como construir e organizar sistemas que modelam características e preferências de usuários de forma a adaptar-se às suas necessidades?	Sociedade multiagente para Sistemas Adaptativos/adaptáveis
	Padrões de projeto detalhado	Como criar e manter modelos de usuários em sistemas adaptativos?	A solução envolve a criação de um agente responsável por construir e manter modelos de usuários.
Padrões Independentes de Domínio		Como projetar um agente para que possa raciocinar sobre um problema de forma que o possibilite atingir metas pró-ativamente?	Deliberativo
		Como projetar um agente para apenas reagir a estímulos do ambiente no qual está inserido?	Reativo

2. Padrão Sociedade multiagente para sistemas Adaptativos

Contexto

Os sistemas onde o usuário pode iniciar, propor, selecionar e produzir a adaptação ou também deixar que o sistema execute algumas dessas funções, são chamados de sistemas adaptáveis. Os sistemas que executam todas as funções acima citadas, de forma automática, são chamados de sistemas adaptativos. O objetivo geral de tais sistemas é prover seus usuários com conteúdo atualizado, subjetivamente interessante, com informação pertinente, num tamanho e profundidade adequados ao contexto e em correspondência direta com o *modelo do usuário*.

O modelo de usuário é uma fonte de conhecimento que contém informações, explícitas ou implicitamente adquiridas, de todos os aspectos relevantes ao usuário para serem utilizados no processo de adaptação de uma aplicação de software.

As principais áreas de aplicação dos sistemas adaptativos são:

- Comércio eletrônico [31];
- Interfaces adaptativas [20];
- Turismo [13];
- Recuperação de informação [23] [22];
- Educação [3].

Problema

Como construir e organizar sistemas que modelam características e preferências de usuários de forma a adaptar-se às suas necessidades?

Forças

As características de autonomia, habilidade social, reatividade, pró-atividade e capacidade de aprendizagem, próprias dos agentes, provêm mecanismos apropriados para atingir os objetivos dos sistemas adaptativos/adaptáveis que utilizam técnicas de modelagem de usuários. Frequentemente, os agentes estão aptos a adaptar-se aos seus ambientes e exibem algum grau de inteligência, apesar dessas características não serem obrigatórias.

As características dos agentes podem ser associadas com os principais objetivos da utilização de modelos de usuários em sistemas adaptativos, refletindo assim as razões para a utilização da tecnologia de agentes em tais sistemas, como pode ser visto na Tabela 2.

Tabela 2 Padrões para o projeto de sistemas adaptativos/adaptáveis

Objetivos dos sistemas adaptativos/adaptáveis	Características dos agentes
Melhorar a eficiência e eficácia da interação (facilitar e acelerar interações)	Reatividade – através da interação com o ambiente, os agentes reagem rapidamente a mudanças no ambiente e apresentam ao usuário o que ele quer ver de forma a refletir o seu perfil.
Tornar os sistemas complexos mais usáveis	Habilidade social - os agentes cooperam e interagem com outros agentes para resolver tarefas complexas tornando os sistemas mais simples do ponto de vista do usuário
Ajudar o usuário a encontrar informação	Autonomia e pró-atividade – o usuário pode delegar tarefas (objetivos) para que os agentes de software as executem autonomamente.
Prever o comportamento futuro do usuário	Capacidade de aprendizagem - os agentes podem aprender com o comportamento do usuário e fazer inferências sobre o seu comportamento esperado.

Solução

Modelar a aplicação como uma sociedade de agentes. Seriam usados agentes especializados para cada usuário ou grupos de usuários, como agente de interface, agente de modelagem e agente de adaptação.

O processo de modelagem de usuários e o processo de personalização de aplicações podem ser divididos em três tarefas maiores que podem ser executadas por diferentes componentes do sistema:

- A *aquisição* consiste na identificação de informações sobre usuários, como características, preferências, necessidades e fazer essas informações acessíveis ao componente de adaptação da aplicação;
- A *representação* consiste na representação dos conteúdos dos modelos de usuários e modelos de adaptação, apropriadamente, em um sistema formal, permitindo o seu acesso e posterior processamento e formular suposições sobre usuários e/ou grupos de usuários, seus comportamentos e seu ambiente;
- A *produção* consiste na geração da adaptação do conteúdo, da apresentação, da modalidade e da estrutura, baseados nos modelos de usuários, nos modelos de adaptação e nos modelos do domínio.

As tarefas de aquisição e produção podem ser atribuídas ao agente de interface, a tarefa de representação e manutenção de modelos de usuários pode ser atribuída ao agente de modelagem e o agente de adaptação pode ser responsável pela representação e manutenção de modelos de adaptação, como pode ser visualizado na Figura 2. Durante o processo de modelagem de usuários, de acordo com a Figura 2., o agente de interface colhe dados sobre o usuário; esses dados são processados e representados em um modelo de usuário pelo agente de modelagem. Na execução do processo de adaptação, o agente de adaptação produz modelos de adaptação baseados no modelo de usuário, esses modelos são utilizados pelo agente de interface para personalizar a aplicação.

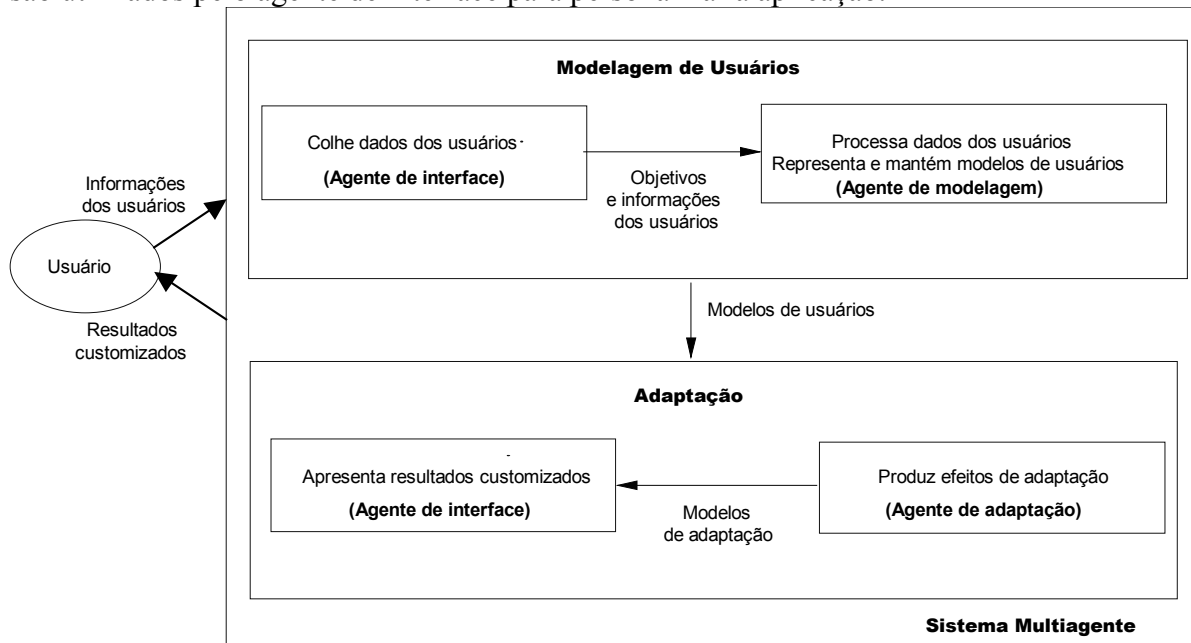


Figura 2 O processo da modelagem de usuários e adaptação de sistemas utilizando agentes

Padrões relacionados

A solução arquitetural, descrita neste sistema de padrões, para estruturar os agentes dos sistemas adaptativos/adaptáveis é descrita no padrão chamado Camadas multiagente para sistemas adaptativos/adaptáveis.

Usos conhecidos

Bylund [6] e Baldassin [2] propõem o uso de agentes para oferecer serviços personalizados aos usuários, distribuindo tarefas entre esses agentes

3. Camadas Multiagente para Sistemas Adaptativos/adaptáveis

Contexto

Nos sistemas adaptativos/adaptáveis, baseados na tecnologia de agentes, é preciso projetar soluções para organizar os agentes de maneira que possam satisfazer as necessidades e preferências de seus usuários.

Problema

Como estruturar um sistema multiagente apto a adaptar-se às necessidades de diferentes tipos de usuários ou grupos de usuários?

Forças

A organização dos agentes em camadas, segundo a similaridade de responsabilidades tem todas as vantagens (flexibilidade, facilidade de manutenção, de entendimento e de extensão) descritas pelo padrão camadas [5].

Solução

A solução (Figura 3), baseada no uso de técnicas de adaptação e modelagem de usuários, envolve a definição de um critério para a divisão das responsabilidades entre agentes distribuídos em camadas, de acordo com o padrão camadas descrito por Silva Júnior [29], uma extensão do padrão Camada proposto por Buschmann [5]. Os sistemas adaptativos/adaptáveis que usam a modelagem de usuários são compostos basicamente por uma camada de interface, uma camada de modelagem e uma camada de adaptação, sendo que podem ser adicionadas novas camadas de acordo com o tipo de serviço que o sistema adaptativo/adaptável irá prover.

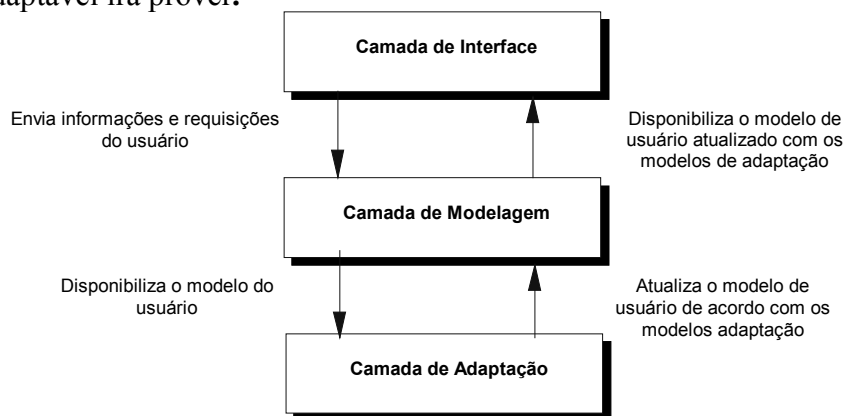


Figura 3. Padrão camadas multiagente para a modelagem de usuários

Um grupo de agentes está associado a cada camada. Cada camada é estruturada como segue.

Camada de interface

A camada de interface consiste em um grupo de agentes de interface onde cada agente está associado a um usuário ou grupo de usuários com necessidades similares para auxiliá-lo(s) em suas tarefas. As principais responsabilidades dos agentes dessa camada são:

- Apresentar uma interface inicial, caso o usuário ainda não possua um modelo de usuário, ou personalizada, baseando-se no modelo de usuário corrente;
- Receber requisições dos usuários;
- Colher informações sobre usuários, tais como características dos usuários, comportamento durante o uso do computador ou do sistema, através do monitoramento do uso do computador ou requisitando essas informações diretamente ao usuário;

- Caso seja adotado um sistema de retroalimentação, fazer perguntas a respeito dos resultados para possíveis melhoras na eficácia do sistema;
- Apresentar informações aos usuários de forma personalizada (respostas e explicações).

Camada de modelagem

A camada de modelagem consiste em um grupo de agentes de modelagem. Cada agente de modelagem provê serviços para um agente de interface. As principais responsabilidades dos agentes dessa camada são:

- Processar a informação provida pelo usuário, através de um agente de interface e atualizar ou criar modelos de usuários, se eles ainda não existirem, de acordo com esta informação;
- Formular metas ou inferir suposições sobre usuários ou grupo de usuários com base nos modelos de usuários;
- Reformular consultas, tarefas e metas dos usuários de acordo com os modelos de usuários e disponibilizá-los para o agente de adaptação correspondente.

Camada de adaptação

A camada de adaptação consiste em um conjunto de agentes de adaptação. Cada agente de adaptação provê serviços para o agente de modelagem. As principais responsabilidades dos agentes dessa camada são as seguintes:

- Construir, representar e manter modelos de adaptação, de acordo com os modelos de usuários;
- Atualizar os modelos de usuários de acordo com os modelos de adaptação.

Padrões relacionados

- Para o projeto dos agentes da camada de interface pode ser utilizado padrão Interface.
- Para o projeto dos agentes da camada de modelagem pode ser utilizado o padrão Modelagem.
- Para o projeto dos agentes da camada de adaptação pode ser utilizado o padrão Adaptação.

Usos conhecidos

A organização em camadas é típica das arquiteturas multiagente para filtragem e recuperação de informação, tais como as arquiteturas RETSINA [28], AMALTHAEA [22] e ABARFI [9], bem como as arquiteturas de sistemas Web Adaptativos, como a arquitetura genérica descrita por Sharma [27]. Este padrão é inspirado nas soluções arquiteturais propostas por tais sistemas.

4. Padrão Modelagem

Contexto

Nos sistemas adaptativos/adaptáveis, baseados na tecnologia de agentes, é preciso projetar soluções para representar os usuários através da construção de modelos de usuários que especificam suas características, suas preferências e os seus interesses.

Problema

Como criar e manter modelos de usuários que serão utilizados como referência para que a aplicação ofereça serviços personalizados aos seus usuários?

Forças

Os usuários diferem em seus níveis de conhecimento, necessidades e preferências. O tema da adaptabilidade é fornecer informação sob medida de acordo com as características dos usuários, por exemplo, adaptando o estilo de apresentação durante a interação com o usuário. Qualquer abordagem para apresentar informação sob medida envolve a criação e manutenção de modelos de usuários que devem estar aptos a refletir as mudanças nas características desses usuários no decorrer do tempo.

A Modelagem de usuários é necessária, pois através dela se pode sintetizar as características e habilidades de usuários ou grupo de usuários com o fim de oferecer serviços personalizados.

Solução

A solução envolve a criação de um agente de modelagem do tipo deliberativo, responsável por construir e manter modelos de usuários com base nas informações fornecidas por um agente de interface através da interação com o usuário.

O agente de modelagem recebe informações sobre os usuários do agente de interface, e os classifica como pertencentes a um ou mais grupos, chamados de estereótipos; introduz as características específicas de tais grupos no modelo individual de cada usuário; faz inferências de novas hipóteses sobre os usuários com base nos fatos iniciais.

Algumas das técnicas mais comuns para a representação de modelos de usuários são: redes bayesianas [21], lógica fuzzy, redes neurais, redes de Petri [18], representação baseada em lógica [25], estereótipos [34], representação vetorial [22] e ontologias [11].

Além da representação dos modelos de usuários, o agente de modelagem é responsável pela manutenção da consistência dos modelos verificando as novas informações fornecidas pelo agente de interface e/ou pelo agente de adaptação, comparando com informações anteriores. Dentre as técnicas mais comuns para fazer a manutenção de modelos de usuários podemos destacar a lógica fuzzy (teoria de conjuntos difusos), a teoria de probabilidades, algoritmos genéticos [18].

Padrões relacionados

O agente de modelagem pode ser definido como um agente do tipo deliberativo que tem suas características descritas pelo padrão Deliberativo.

Os modelos de usuários são utilizados pelo agente de adaptação para construir modelos de adaptação para cada usuário ou grupo de usuários. As características do agente de adaptação são descritas pelo padrão Adaptação

Usos conhecidos

Este padrão está sendo utilizado na construção de agentes de modelagem que utilizam técnicas de aquisição implícitas do modelo baseadas em algoritmos genéticos [30] no contexto dos projetos de pesquisa MaAE e JURÍDICAS [15].

5. Padrão Deliberativo

Contexto

Em sistemas multiagente, onde existem tarefas complexas, é preciso projetar agentes que estejam aptos a executar essas tarefas, exibindo comportamento direcionado a metas, tomando iniciativas e decisões através de raciocínio e/ou adaptando-se às mudanças no ambiente.

Problema

Como projetar um agente para que possa raciocinar sobre um problema de forma que o possibilite atingir metas pró-ativamente dentro do contexto no qual está inserido?

Forças

Tendo conhecimento, sobre o seu ambiente e usando mecanismos de raciocínio para encontrar uma solução adequada, agentes deliberativos são capazes de executar tarefas complexas.

Solução

O agente deve ser do tipo deliberativo, ou seja, ele deve possuir modelos simbólicos internos de raciocínio do ambiente no qual ele está inserido e de si mesmo. Ele raciocina sobre esses modelos para criar um plano que lhe permite atingir suas metas. O padrão deliberativo é estruturado por cinco módulos organizados verticalmente e horizontalmente: módulo de comunicação, módulo de ação, módulo de raciocínio, módulo de sensores e módulo de conhecimento. A estrutura do padrão deliberativo está representada na Figura 4.

Através do *módulo sensores*, o agente percebe do ambiente e atualiza seu conhecimento com estas percepções para refletirem o estado corrente do ambiente. Percepções são informações que o agente pode receber do mundo real, informações de recursos e interações com outros usuários. A camada sensitiva também recebe mensagens de outros agentes da sociedade.

No *módulo de raciocínio*, baseado nos seus conhecimentos, o agente elabora metas e planos para atingir essas metas.

No *módulo de ação* o agente atua no ambiente executando as ações do plano gerado pelo módulo de raciocínio. A ação executada é armazenada no módulo de conhecimento para refletir os efeitos das ações no ambiente. O módulo de ação serve o módulo de comunicação. Durante a execução de um plano, um agente pode determinar a necessidade de cooperação com outros agentes, requerendo informações ou pedindo para executar ações. Esta cooperação é suportada pelo módulo de comunicação.

No *módulo de comunicação*, o agente envia e processa mensagens.

O *módulo de conhecimento* é um repositório onde o agente armazena informações sobre o estado corrente do ambiente através do histórico das percepções do agente e ações executadas. Esta camada contém recursos e comportamento do agente e também conhecimento sobre as habilidades dos outros agentes da sociedade.

Os módulos de sensores, de raciocínio, de ação e de comunicação têm acesso direto ao módulo de conhecimento para que possam executar suas respectivas funcionalidades.

Cada módulo pode ser construído através da composição de um grupo de padrões para prover a funcionalidade de cada módulo.

O ciclo de execução do agente consiste em: no módulo de sensores, a percepção é usada para atualizar o conhecimento do agente. No módulo de raciocínio, este conhecimento é usado para determinar, dinamicamente, as metas correntes do agente para as quais um plano de ações foi construído, de acordo com as capacidades e comportamento do agente no módulo de conhecimento. O plano é então executado no módulo de ação. Quando um plano necessita de ações que o agente não está apto a executar porque elas não

fazem parte das suas capacidades e comportamentos, o módulo de comunicação determina a necessidade de cooperação com outros agentes. Então mensagens são enviadas para agentes capazes de ajudá-lo a cumprir o plano; as mensagens de outros agentes são recebidas através do módulo de sensores, depois então o ciclo completo é repetido. O módulo de conhecimento é atualizado a cada novo ciclo pelos demais módulos.

1- Recebe mensagens e percebe mudanças no ambiente

2- Registra percepções e mensagens recebidas

3- Percepções ou mensagens recebidas

4- Consulta capacidades do agente

5- Plano de ações

6- Ações não suportadas pelo agente

7- Registra e consulta possíveis colaborações com outros agentes

8- Envia mensagens para outros agentes

9- Atua no ambiente

10- Registra ações realizadas

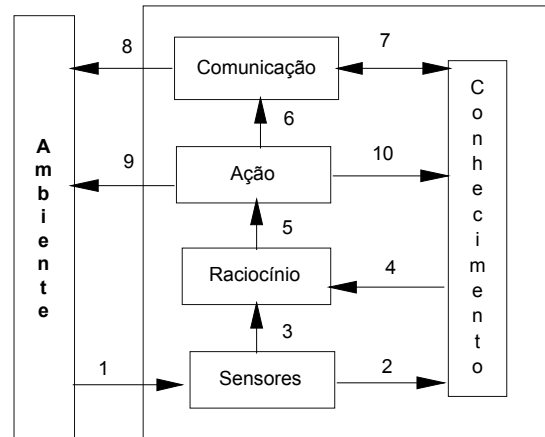


Figura 4 Estrutura do Padrão Deliberativo

Padrões relacionados

Os agentes deliberativos são amplamente utilizados em arquiteturas multiagente que utilizam raciocínio complexo [32], Kendall [17] propõe um padrão alternativo, chamado Agente deliberativo que difere do padrão proposto neste trabalho principalmente em dois aspectos. Primeiro, o padrão agente deliberativo sugerido por Kendall acrescenta as camadas de mobilidade e tradução, funcionalidades das quais fazemos abstração atualmente no nosso trabalho. Segundo, o nosso padrão propõe a base de conhecimento como camada que interage verticalmente com todos os outros módulos do agente deliberativo

Usos conhecidos

Os ambientes de desenvolvimento de sistemas multiagente, como AgentBuilder [1] e Zeus [10] [33] e JADE [16] usam diferentes arquiteturas genéricas para a construção de agentes deliberativos. Eles provêm funcionalidades semelhantes às providas pelas camadas do padrão deliberativo que nós propusemos. Porém, a forma como os módulos interagem é diferente. No agente genérico do AgentBuilder e JADE/JESS, os módulos interagem diretamente sem considerar uma hierarquia de camadas. O agente genérico do Zeus é organizado em camadas, no entanto o conhecimento do agente está distribuído em várias camadas.

6. Padrão Reativo

Contexto

Em sistemas multiagente, onde existem tarefas que demandam respostas rápidas durante a sua execução, é necessário projetar agentes que estejam aptos a reagir a estímulos do seu ambiente, mas que não utilizem raciocínio complexo e que não tenham conhecimento sobre o ambiente no qual estão inseridos.

Problema

Como projetar um agente para apenas reagir a estímulos do ambiente no qual está inserido ou a mensagens de outros agentes quando ele não tem conhecimento sobre esse ambiente e nem pode aprender a partir dele?

Forças

O uso deste padrão é indicado em aplicações que demandam respostas rápidas requerendo que os agentes reajam a estímulos e atuem no seu ambiente. Mesmo não exibindo comportamento racional, os agentes estruturados de acordo com o padrão reativo podem exibir algum tipo de comportamento inteligente através da interação com outros agentes em sistemas multiagente.

Solução

O agente deve ser do tipo reativo, ou seja, ele não deve ter um modelo interno do ambiente no qual está inserido, o agente deve agir usando um comportamento estímulo/resposta de acordo com o estado corrente do ambiente ao qual está integrado. A solução do padrão reativo estrutura um agente em quatro módulos: comunicação, ação, regras e sensores. A estrutura do padrão reativo está representada na Figura 5.

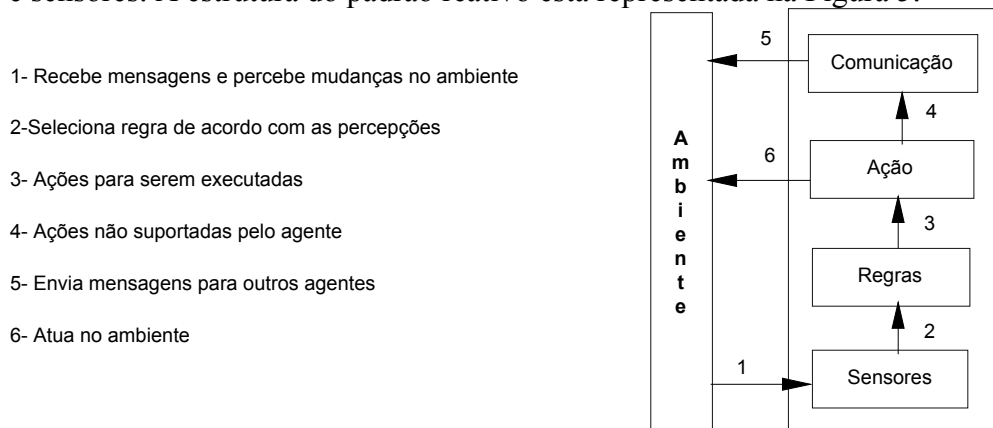


Figura 5 Estrutura do Padrão Reativo

Através do *módulo de sensores*, o agente percebe o ambiente. Percepções são informações que o agente pode receber do mundo real, informações de recursos e interações com usuários. O módulo de sensores também recebe mensagens de outros agentes da sociedade. O módulo de sensores serve o módulo de regras.

O *módulo de regras* é composto de um conjunto de regras de condição-ação da forma "**se** <condição> **então** <ação>". Baseado nos estímulos percebidos pelo módulo de sensores uma regra condição-ação é escolhida. O módulo de regras serve o módulo de ação.

No *módulo de ação*, uma ação é selecionada para ser executada.

Durante a execução de uma ação, o agente determina a necessidade de cooperação com outros agentes para requerer informações ou pedir a execução de uma ação, o módulo de comunicação suporta a cooperação com outros agentes.

No *módulo de comunicação*, o agente envia e processa mensagens.

Cada módulo pode ser contruído através da composição de um grupo de padrões para prover a funcionalidade de cada módulo.

O ciclo de execução do agente consiste em: no módulo de sensores, um estímulo é percebido. No módulo de regras, este estímulo é usado para determinar a ação que o agente irá executar no módulo de ação. Se para a realização da ação selecionada o agente necessita cooperar com outros agentes, o módulo de comunicação é ativado. Então mensagens são enviadas para os agentes capazes de cooperar; as respostas dos outros agentes são recebidas através do módulo de sensores, então o ciclo completo se repete.

Padrões relacionados

Kendall [17] propõe um um padrão alternativo, chamado Agente reativo, para estruturar agentes reativos similar ao desenvolvido neste trabalho, sendo que o o padrão sugerido por Kendall acrescenta as camadas de mobilidade e tradução, funcionalidades das quais fazemos abstração atualmente no nosso trabalho.

Usos conhecidos

Este tipo de solução para a estruturação dos agentes foi originalmente proposta na arquitetura de classificação de Brooks [4]. Kulkarni [19] criou um sistema de comportamento reativo chamado ReBa que pode disparar diferentes ações em resposta a eventos de dispositivos. Dentre os ambientes de desenvolvimento de software e frameworks multiagente que temos utilizado [1] [33] [16], apenas o JADE [16] provê uma arquitetura genérica para o desenvolvimento de agentes reativos.

7. Agradecimentos

Este trabalho é apoiado pelo CNPq.

Os autores agradecem aos responsáveis pelo processo de revisão, em particular ao Jerffeson Souza, pelas contribuições realizadas no melhoramento da descrição dos padrões propostos neste trabalho.

8. Bibliografia

- 1 “AgentBuilder User’s guide” (2003), Reticular Systems, external documentation, <http://www.agentbuilder.com/>.
- 2 BALDASSIN, Alexandre, RIZZO, Ivan, MALTEMPI, Marcus (2002) “Uma Abordagem Baseada em Agentes para Filtragem de Correspondências Eletrônicas”. Revista Eletrônica de Iniciação Científica (REIC). Vol. II, nº 4.
- 3 BEAUMONT, I. (1994) “User Modelling in the Interactive Anatomy Tutoring System ANATOM-TUTOR”. User Modelling and User-Adapted Interaction 4(1) 21 –45.
- 4 BROOKS, R (1986) “A robust layered control system for a mobile robot”. IEEE Journal of Robotics and Automation, 2 (1): 14-23.
- 5 BUSCHMANN, F., MEUNIERS, R., ROHNERT, H., SOMMERLAD, P., STAL, M. A (1996) “System of Patterns. Pattern-Oriented Software Architecture”, Wiley.
- 6 BYLUND, Markus, WAERN, Annika (1997) “Adaptation Agents: Proving Uniform Adaptations in Open Service Architectures”, In: Proc. of 3rd ERCIM Workshop on UI for All.
- 7 COSENTINO, Massimo, BURRAFATO, Piermarco, LOMBARDO, Saverio, SABATUCCI, Luca (2002) “Introducing Pattern Reuse in the Design of Multi-Agent Systems”. AITA'02 workshop at NODe02 - 8-9.
- 8 DEUGO, Dwight, WEISS, Michael, KENDALL, Elizabeth (2001) “Reusable Patterns for Agent Coordination”. Publicado como capítulo 14 do livro: Omicini, A., Zambonelli, F., Klusch, M.,Tolksdorf, R., Coordination of Internet Agents: Models, Technologies, and Applications, Springer.
- 9 DINIZ, Alessandra (2001) “Uma Arquitetura baseada em Agentes para a Recuperação e Filtragem da Informação”. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão.

- 10 ERICEIRA, Bruno (2001) "Desenvolvimento de Sistemas Multiagente Utilizando a Ferramenta Zeus". Monografia do Curso de Bacharel em Ciência da Computação, Departamento de Ciências da Computação, Universidade Federal do Maranhão, São Luís-MA.
- 11 FARIA, Carla, RIBEIRO, Ismênia, GIRARDI, Rosario (2003) "Especificação de uma Ontologia Genérica para a Construção de Modelos de Usuários", Anais da Terceira Jornada Iberoamericana de Engenharia de Software e Engenharia do Conhecimento (JIISIC 2003, Valdivia, Chile).
- 12 FERREIRA, Steferson Lima Costa, GIRARDI, Rosario (2002) "Arquiteturas de Software baseadas em Agentes: do Nível Global ao Detalhado", Revista Eletrônica de Iniciação Científica da SBC.
- 13 FINK, J, KOBASA, A. (2002) "User Modeling in Personalized City Tours". Artificial Intelligence Review 18(1), 33-74.
- 14 GIRARDI, Rosario (2001) "Agent-Based Application Engineering", In: Proc. of 3rd International Conference on Enterprise Information Systems (ICEIS 2001).
- 15 GIRARDI, Rosario (2002) "Reuse in Agent-based Application Development", In: 1st International Workshop on Software Engineering for Large-Scale Multi-Agent Systems (SELMAS'2002), International Conference on Software Engineering, Orlando, Florida.
- 16 JADE, Java Agent Development framework (2003). <http://jade.cselt.it>
- 17 KENDALL, Elizabeth (1997) "The Layered Agent Pattern Language", In: Proc. of Pattern Languages of Programming (PLOP'97).
- 18 KOBASA, Alfred (1999) "Personalised hypermedia presentation techniques for improving online customer relationships", GMD Report 66.
- 19 KULKARNI, A (2002) "A Reactive Behavioral System for the Intelligent Room". Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, <http://citeseer.nj.nec.com/kulkarni02reactive.html>. Accessed in May 2003.
- 20 LANGLEY, P (1999) "User Modeling in Adaptive Interfaces", In: J Kay (ed.) UM99 User Modeling: Proceedings of the Seventh International Conference Springer 357-370.
- 21 MISLEVY, RJ, GITOMER, DH (1996) "The Role of Probability-based Inference in an Intelligent Tutoring System", User Modeling and User-Adapted Interaction 5(3-4) 253-282.
- 22 MOUKAS, Alexandros and MAES, Pattie (1996) "Amalthea: Information Discovery and Filtering using a Multiagent Evolving Ecosystem", Proceedings of the Conference on Practical Applications of Agents and Multiagent Technology.
- 23 NICK, Z, THEMIS, P. (2001) "Web Search Using a Genetic Algorithm". IEEE Internet Computing, 5(2), 18-26.
- 24 OLIVEIRA, Ismênia Ribeiro de (2001) "Uma Análise de Padrões de Projeto para o desenvolvimento de Software baseado em agentes". Monografia do Curso de Bacharel em Ciência da Computação, Departamento de Ciências da Computação, Universidade Federal do Maranhão, São Luís-MA.
- 25 POHL, Wolfgang, (1999) "Logic-Based Representation and Reasoning for User Modeling Shell Systems", User Modeling and User-Adapted Interaction 9: 217-282, Kluwer Academic Publishers. Printed in the Netherlands.
- 26 RUSSELL, S, NORVIG, P (1995) "Artificial Intelligence: A Modern Approach". Prentice-Hall.
- 27 SHARMA, Amit (2001) "A Generic Architecture for User Modeling Systems and Adaptive Web Services", In: Workshop on E-Business & the Intelligent Web. (IJCAI, 2001).
- 28 SHEHORY, Onn and SYCARA, Katia (2000), "The Retsina Communicator". In Proceedings of Autonomous Agents, Poster Session.
- 29 SILVA JUNIOR, Geovanne Bezerra da (2003) "Padrões Arquiteturais para o Desenvolvimento de Aplicações Multiagente". Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão.
- 30 SOBRINHO, Antonio Carlos (2003) "Uma Análise dos Algoritmos Genéticos e suas Aplicações em Sistemas de Acesso à Informação". Monografia do Curso de Bacharel em Ciência da Computação, Departamento de Ciências da Computação, Universidade Federal do Maranhão, São Luís-MA.

- 31 STRACHAN, L., ANDERSON, J., SNEESBY, M, EVANS, M (1997) “Pragmatic User Modelling in a Commercial Software System”, In: User Modeling: Proceedings of the Sixth International Conference, UM97, páginas 189-200. Springer, Vienna, New York.
- 32 WOOLDRIDGE, Michael, JENNINGS, Nicholas (1994) “Intelligent Agents: Theory and Practice”, Knowledge Engineering Review.
- 33 ZEUS DOCUMENTATION, disponível em: <http://193.113.209.147/projects/agents.htm>, acesso em julho, 2002.
- 34 ZHANG, Xiangmin (2003) “Discriminant Analysis as a Machine Learning Method for Revision of User Stereotypes of Information Retrieval Systems”, In: Proceedings of UM'03, 9th International Conference on User Modeling Pennsylvania, USA.