

Agentes de Software: Conceitos e Tecnologias

Alberto Silva*, José Delgado**

Instituto Superior Técnico

INESC - Instituto de Engenharia de Sistemas e Computadores

Grupo de Sistemas e Serviços Telemáticos

Rua Alves Redol, 9

Apartado 13069 - 1000 Lisboa

SUMÁRIO

Com a explosão da Internet, a expectativa de computação global centrada na rede com aplicações diversas como o comércio electrónico, organizações virtuais, administrações públicas electrónicas, etc., os agentes de software tornaram-se um tópico de investigação intensa. Descreve-se neste capítulo o que são agentes de software e o seu contexto geral segundo basicamente a perspectiva e visão de duas comunidades científicas. Por um lado a comunidade da Inteligência Artificial Distribuída, e por outro a comunidade dos Sistemas Distribuídos. Apresentam-se propostas de definição de agentes e referem-se os atributos que de alguma forma permitem identificá-los e caracterizá-los. São apresentados os principais modelos de arquitecturas de agentes e referidas algumas das suas aplicações conhecidas, nomeadamente modelos de agentes intencionais, reactivos e híbridos; de interface com o utilizador; móveis; colaborativos; e de informação.

Por fim faz-se uma crítica da viabilidade dos agentes de software no contexto referido e referem-se expectativas futuras.

1 INTRODUÇÃO

À medida que se assiste, por parte da generalidade das organizações, a um esforço notável de mudança para sistemas cliente-servidor (vid. Capítulo 3), encontra-se em marcha um novo paradigma de **computação baseada na rede** (*network centric computing*). Neste modelo, conteúdo (dados e código), comunicação e computação convergem e a rede torna-se o que o computador fora outrora [Gil+95]. Um dos novos padrões de funcionamento neste modelo é a delegação de tarefas. Por exemplo um utilizador pode especificar um conjunto de tarefas e de objectivos (e.g., pesquisa, filtro e obtenção de informação; compra ou venda de

determinado produto; etc.), que a sua aplicação especializada deverá realizar. Este tipo de aplicação designa-se por **agente software**, a qual é responsável por suportar delegação de tarefas, gerir respectiva complexidade, suportar a mobilidade dos utilizadores; adaptar-se ao seu utilizador, adaptar-se ao seu meio; aprender de forma adaptativa, etc.

O conceito de “agente” é uma metáfora utilizada em inúmeras áreas do conhecimento, desde a Psicologia, Sociologia, Biologia até às Ciências da Computação.

Foi a comunidade de Inteligência Artificial, pelo facto de basear-se desde sempre em metáforas psicológicas e sociais quer como fonte de inspiração e de motivação, quer como forma de modelização do processamento (complexo) de informação, que adoptou pela primeira vez, nas Ciências da Computação, este conceito. Esta primazia é geralmente associada a Carl Hewitt pela sua proposta do sistema Actor [Hew77]. No seu modelo Hewitt propõe o conceito de objecto auto-contido, interactivo e com execução concorrente, ao qual designa por “actor”. Cada objecto encapsula o seu estado interno e pode responder a mensagens de outros objectos similares.

“Um actor é um agente computacional que tem um endereço de correio electrónico e um comportamento. Os actores comunicam por troca de mensagens e executam as suas mensagens concorrentemente” [Hew77].

O foco de interesse primordial da Inteligência Artificial Distribuída (IAD) consiste na solução de problemas considerados complexos que deveriam envolver grupos de indivíduos. Os constantes avanços tecnológicos, nomeadamente o aparecimento de máquinas paralelas potentes e principalmente a explosão das redes de computadores, vêm fomentando cada vez mais esta área de investigação. Segundo Bond [BG88] a IAD pode dividir-se em duas áreas principais: a resolução

* Telefone: 310 03 07

- Email: Alberto.Silva@inesc.pt

** Telefone: 310 02 11

- Email: Jose.Delgado@inesc.pt

distribuída de problemas; e sistemas multiagente. Há contudo a acrescentar a área da IA paralela.

Por outro lado, a comunidade de Sistemas Distribuídos (SD) tem vindo a preocupar-se com a modelização e construção de aplicações distribuídas de uma forma indistinta e pragmática. As metáforas em geral adoptadas estão associadas às linguagens e paradigmas de programação de uso geral (programação estruturada e programação orientada por objectos). As principais questões levantadas dizem respeito às (melhores e mais fáceis) formas de desenvolvimento de aplicações generalistas, sem “inteligência”, em ambientes distribuídos. Os problemas colocam-se ao nível das linguagens de programação, dos sistemas operativos e dos sistemas persistentes. As principais propostas evoluíram a partir de sistemas de comunicação de mensagens de baixo nível, mecanismos de invocação de procedimentos remotos, mecanismos de avaliação remota de código, sistemas de objectos distribuídos, até às recentes propostas de sistemas de código móvel (vid. Capítulo 3).

O aparecimento dos sistemas de código móvel e as suas características comuns com os sistemas multiagentes, agravou a confusão relativamente a uma definição consensual de agente. Tal facto é ainda significativamente agravado com a popularização massiva da Internet e da Web, com a constatação das suas limitações e com a criação de novas expectativas à volta de uma nova vaga de tecnologias de informação centradas na rede.

Todavia, esta vaga tecnológica não surge impune a críticas e contusões. Uma das principais críticas geralmente lançada diz que os agentes não são efectivamente uma nova técnica, e que tudo o que pode ser realizado com agentes, pode igualmente ser realizado nos modelos computacionais conhecidos (modelo centralizado e modelo cliente-servidor). Outra crítica diz respeito à denominada “inteligência” dos agentes: pelo facto da generalidade dos sistemas de agentes existentes e funcionais de facto não apresentar uma arquitectura particularmente complexa ou versátil, alguns críticos questionam o significado da designada “inteligência” desses sistemas. (A definição de “inteligência”, no contexto computacional, é ela própria uma questão em aberto e em discussão há vários anos sem se ter conseguido uma posição consensual e universalmente aceite).

Por outro lado, essas críticas são refutadas, principalmente pela comunidade da IAD, com os seguintes argumentos:

- Os sistemas multiagentes diferem dos outros modelos conhecidos pelo facto de permitirem soluções razoáveis, em termos de custos e de tempo envolvido, para a resolução de problemas particularmente complexos, ou cuja solução exigisse a participação de múltiplos intervenientes (e.g., um sistema de tomada de decisão em grupo, em que uma decisão final é resultado de um processo baseado em decisões parcelares de

múltiplos intervenientes, por vezes com objetivos antagónicos).

- Os agentes permitem eliminar as diferenças entre os distintos tipos de redes e SI legados e consequentemente eliminar, ou atenuar, do ponto de vista aplicacional, as fronteiras entre todas essas redes e respectivos sistemas. Ou seja, os agentes permitem, ao encapsularem a complexidade e especificidades inerentes a cada rede e sistema computacional, a concepção e construção de aplicações altamente distribuídas e heterogéneas.

Harrison, Chess e Kershenbaum analisaram [HCK95], segundo a perspectiva da comunidade de SD, as vantagens e desvantagens associadas a aplicações baseadas em agentes móveis. Concluíram que apesar dos agentes apresentarem várias vantagens (e.g., permitirem operações desconectadas, encaminhamento semântico, distribuição dos serviços básicos dos clientes) face aos modelos tecnológicos conhecidos, a generalidade das aplicações que poderiam ser concretizadas com o modelo de agentes também o poderiam ser com base nos modelos conhecidos. Contudo, referem que as suas vantagens, vistas de uma forma agregada ou em conjunto, tornam-os extremamente atractivos, comparativamente com os modelos conhecidos, para o desenvolvimento de aplicações distribuídas.

A secção seguinte aborda com detalhe a definição, ou definições, do conceito de agente, segundo diferentes perspectivas e comunidades. A secção 4.3 apresenta alguns exemplos de aplicação representativas da tecnologia dos agentes, ilustrando-se deste modo a sua versatilidade e características inerentes. A secção 4.4 descreve exemplos concretos de sistemas de agentes representativos das duas principais comunidades de investigação envolvidas. Na secção 4.5 focaliza-se a análise anterior (da secção 4.4) para o espaço computacional da Web. Por fim, na secção 4.6, conclui-se a apresentação deste capítulo com a definição da perspectiva de agente e de sistema de agentes adoptada nesta tese, e com uma síntese das principais limitações e expectativas que esta área tecnológica levanta.

2 DEFINIÇÕES E ATRIBUTOS DE AGENTES

Existem inúmeras definições e interpretações de agentes de software. Eis algumas das designações usualmente adoptadas: agentes inteligentes; agentes móveis; agentes estacionários; agentes autónomos; agentes de informação; agentes reactivos; agentes deliberativos; agentes cognitivos; softbots (*software robots*); knowbots (*knowledge-based robots*); agentes espertos; agentes colaborativos; assistentes pessoais, etc.

O facto do conceito de agente ser antes de mais um paradigma ou modelo de desenho de sistemas e de apresentar inúmeras áreas de aplicação com distintas

características, contribuiu para a impossibilidade de um consenso geral à volta de uma definição única.

Na perspectiva da IA existiu desde sempre a tentativa de definição e clarificação do que consistem os sistemas de agentes. Para essa comunidade, um agente é um sistema computacional, que para além das características básicas de autonomia, persistência, ou sociabilidade, é também associado com atributos geralmente aplicados ao ser humano. É normal em IA caracterizar-se um agente usando noções mentais, tais como: o conhecimento, as crenças, as intenções, e as obrigações [Sho93]. Outras teorias vão mais longe no processo de antropomorfismo dos agentes, ao ponto de os considerar com emoções [Bat94].

Uma das mais compreensíveis tipificações de agentes é a de Wooldridge e Jennings [WJ95] que define um agente segundo uma visão, ou noção, “forte” e uma visão “fraca”. Basicamente as duas visões de agentes são definidas pelos atributos que estes deverão possuir, os quais se dividem respectivamente em dois grupos: os atributos essenciais e os atributos opcionais.

A noção fraca baseia-se no conjunto mínimo de características que um agente deverá possuir em geral, tais como:

- **Autonomia**. Os agentes operam sem a intervenção directa dos seus utilizadores ou outros, e têm algum tipo de controlo sobre as suas acções e o seu estado interno.
- **Sociabilidade**. Os agentes interactuam com outros agentes (e possivelmente com os seus utilizadores) através de algum tipo de linguagem de comunicação de agentes.
- **Reactividade**. Os agentes analisam o seu ambiente, (o qual pode ser o mundo físico, um utilizador através do seu interface gráfico, uma colecção de outros agentes; a Internet; ou talvez todos eles combinados), e responder em tempo útil às alterações nele ocorridas. Estes agentes são designados reactivos e são em geral simples e fáceis de desenvolver. Baseiam-se sobre três componentes principais: percepção, acção e comunicação.
- **Pró-actividade** (ou orientação por objectivos). Os agentes não actuam apenas em resposta a alterações no seu ambiente, mas também apresentam comportamento conduzido por objectivos e são capazes de tomar iniciativa na realização de determinadas acções.
- **Persistência**. Os agentes mantêm consistentemente o seu estado interno ao longo da sua existência.

A figura 4.1 ilustra em termos esquemáticos a noção elementar de agente, designadamente a sua relação com o ambiente que o envolve. São particularmente evidentes as capacidades de persistência (estado interno), reactividade (percepções e acções), e pró-actividade (acções). A característica de sociabilidade (i.e., de comunicação entre agentes) não se encontra esquematizada e obrigaria à introdução de um novo

bloco (com a designação de “comunicação”) no esquema do agente.

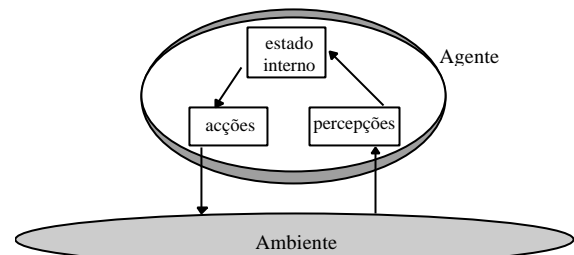


Figura 4.1: Esquema elementar de um agente inteligente.

Todavia, a noção de agente, particularmente para a comunidade de IAD, apresenta um significado mais forte do que a noção (fraca) apresentada. Para essa comunidade, um agente é um sistema computacional, que para além dos atributos apresentados, é também representado com atributos antropomórficos (tais como mentais ou mesmo emocionais), e em que se exige uma representação simbólica do ambiente envolvente, capacidades cognitivas e capacidades de aprendizagem. Eis alguns dos atributos geralmente considerados adicionais à determinação de agente:

- **Mobilidade**. A capacidade de um agente em se mover/viajar através de uma rede electrónica de modo a realizar as suas tarefas e cumprir os seus objectivos. Diz-se agente móvel caso tenha capacidade de se mover, caso contrário, diz-se estático ou estacionário. (Esta característica é essencial para os agentes vistos segundo a perspectiva da comunidade de SD).
- **Intencionalidade**. Capacidade de representação explícita dos objectivos de um agente. Estes agentes, ditos intencionais ou cognitivos, apresentam quatro componentes. Para além da percepção, acção e comunicação, apresentam ainda, capacidade de raciocínio sobre uma base de conhecimento.
- **Aprendizagem**. A capacidade de aprendizagem está intrinsecamente associada com a capacidade de manipulação e geração de conhecimento. Os agentes com capacidade de aprendizagem (e.g., assistentes pessoais, clientes de correio electrónico inteligentes) vão, à medida da sua utilização, reconhecendo padrões de comportamentos, padrões de preferências, etc., e actualizando a sua base de conhecimentos. Esta característica levanta a questão da competência e da confiança [Mae94]. Um agente é competente quando realiza de forma eficiente (eventualmente surpreendendo o seu utilizador) as tarefas previstas; e deve ser de confiança para que o seu utilizador lhe delegue tarefas continuamente.
- **Veracidade**. A assunção que um agente não comunica (deliberadamente) informação falsa. Quer seja com o seu utilizador, quer seja aos agentes com que interactua. Esta característica está também relacionada com o síndrome da competência e da confiança anteriormente referida.

- **Racionalidade.** A assunção de que um agente (racional) não aceita objectivos impossíveis de concretizar, contraditórios com os seus, ou ainda que não sejam compensadores em termos do risco/custo/esforço envolvido.
- **Benevolência.** A assunção que um agente não tem objectivos contraditórios, e que todo o agente procura realizar as tarefas que lhe foram solicitadas. Um agente é benevolente se adopta os objectivos dos outros, caso lhe seja solicitado, desde que estes não sejam incompatíveis com os seus.
- **Características mentais.** A atribuição de noções mentais - conhecimento, crenças, intenções ou desejos - aos agentes foi um passo importante na comunidade de IAD. De entre as teorias envolvidas destacam-se o trabalho de Rao e Georgeff baseado em agentes BDI (agentes com crenças, desejos e intenções) [RG95] e de Wooldrige e Jennings com agentes baseados em atitudes e pró-atitudes [WJ95].

Outras tentativas de definição e tipificação de agentes surgiram recentemente [Gil95,Nwa96,FG96]. Todas elas apresentam similaridades na identificação das características elementares (visão “fraca”), mas diferenças significativas na forma de classificação. Todas elas, no entanto, perfeitamente discutíveis.

Por exemplo, para Nwana,

“Um agente é uma componente de software e/ou de hardware capaz de actuar de forma a resolver tarefas em nome do seu utilizador” [Nwa96].

Esta é propositadamente uma definição suficientemente geral de forma a englobar inúmeros sistemas. É na particularização de agentes que Nwana os consegue tipificar e caracterizar. Na prática propõe uma tipificação de agentes segundo um espaço multi-dimensional, em que são consideradas as seguintes principais dimensões (ou critérios) de análise. Primeira, a mobilidade. Segunda a intencionalidade (vs. reactividade) Terceira, a funcionalidade, i.e., segundo as suas principais especializações (e.g., agentes (de pesquisa) de informação; agentes de apresentações; agentes de compra de produtos). Quarta, o seguinte conjunto mínimo de atributos ideais de agente: autonomia, aprendizagem e cooperação. A combinação cruzada dos referidos atributos gera, segundo Nwana, os seguintes tipos de agentes: autónomos; cooperativos; com capacidade para aprendizagem; colaborativos; de interface; colaborativos e com capacidade para aprendizagem; e espertos (quando reúnem conjuntamente os três atributos ideais). Por motivos de facilidade de apresentação e discussão, Nwana, converteu o seu espaço multi-dimensional numa lista unidimensional, com apenas sete tipos de agentes que considerou relevantes: colaborativos; de interface; móveis; de informação/Internet; reactivos; híbridos; e espertos.

Franklin e Graesser discutem com detalhe a problemática de definição e classificação de agentes. Para eles,

“Um agente (autónomo) é um sistema situado dentro e participante de um ambiente, que percebe (sente) esse ambiente e actua nele de forma a concretizar a sua própria agenda e como consequência, de forma a afectar a sua visão sobre o futuro [FG96].

Discutem taxinomia de agentes baseada quer em modelos biológicos (i.e., baseada em hierarquias de tipos), quer em modelos matemáticos (i.e., baseada em colecções de atributos).

Outra visão menos elaborada de agente, é proposta por Gilbert et al.:

“Um agente (inteligente) é uma entidade software que executa um conjunto de tarefas em nome do seu utilizador, ou em nome de outro programa, com um determinado grau de independência e de autonomia, e consequentemente, utiliza o conhecimento (os objectivos e desejos) do seu utilizador” [Gil+95].

O agente é caracterizado em termos de três características principais: agência, mobilidade, e inteligência. Agência entendida como o grau de autonomia e autoridade, e inteligência entendida como o grau de gestão de preferências, raciocínio, capacidade de planeamento e de aprendizagem patenteada pelo agente.

3 ARQUITECTURAS DE AGENTES

Como referido na secção anterior, existem distintas visões de agentes e de suas inerentes características. Apresentam-se nesta secção as mais representativas arquitecturas de agentes. Atente-se que todas elas são representativas de agentes ou de sistemas de agentes, contudo com origens e objectos distintos e focadas principalmente sobre esta ou aquela característica específica. Consideram-se para análise as seguintes arquitecturas de agentes: intencionais, reactivos e híbridos; de interface com o utilizador; móveis; colaborativos/cooperativos; e de informação.

3.1 AGENTES INTENCIONAIS, REACTIVOS E HÍBRIDOS

A IAD procurou atribuir aos agentes (ditos inteligentes) capacidades de raciocínio e de inteligência inerentes ao ser humano. Esta característica -“inteligência”- é que, segundo a IA, os distingue de outras componentes de software. Foram propostos ao longo da última década as seguintes arquitecturas principais.

3.1.1 Agentes Intencionais

Nestes sistemas o mundo é representado simbolicamente por um número limitado de elementos, que ao ser combinado permite que os agentes envolvidos possam operar sobre um conjunto simbolicamente codificado de instruções. As decisões

com vista à execução de acções são baseadas em manipulação simbólica, raciocínio lógico e reconhecimento de padrões.

Homer [VB90] e Grate [Jen92] são exemplos de aplicações de agentes segundo este modelo

Uma séria limitação dos sistemas simbólicos reside na elevada exigência de processamentos necessária para análise (da informação) “do mundo real”, construção de um plano de solução adequado e implementação da acção determinada.

3.1.2 Agentes Reactivos

Na sequência das críticas e limitações das arquitecturas de agentes intencionais foram propostos modelos baseados em arquitecturas reactivas, nas quais os agentes não possuem representações internas e simbólicas do seu ambiente, nem usam mecanismos de raciocínio simbólico. Alternativamente interactivam entre si segundo um padrão de estímulo-resposta relativamente ao estado corrente do ambiente em que se encontram integrados.

O modelo de referência desta classe de sistemas deve-se a Brooks com a designada *subsumption architecture* [Broo96]. Esta consiste na representação de uma hierarquia de comportamentos concebida para realização de tarefas específicas. Cada comportamento compete com outros comportamentos de forma a influenciar as acções finais do agente. Níveis inferiores da hierarquia representam estilos de comportamentos elementares enquanto que níveis superiores representam comportamentos mais abstractos e elaborados. O comportamento é descrito na linguagem de *subsumption* baseada em extensões às máquinas de estados finitos, o qual representa a única unidade de processamento da arquitectura, ou seja, não é adoptada representação simbólica como a IA clássica.

Inúmeras críticas têm sido dirigidas a esta arquitectura. A primeira, o facto de existirem um número reduzido de aplicações nela baseadas. Segunda, o contexto da sua aplicabilidade é ele próprio reduzido - em geral, aplicado a jogos e a simuladores. Terceira, não é óbvio o desenho e concepção de aplicações gerais com base em agentes reactivos.

3.1.3 Agentes Híbridos

O objectivo da arquitectura de agentes híbridos é a combinação das virtudes (pressupostamente existentes) de ambas as anteriores aproximações. A aproximação simbólica que permite o desenvolvimento de planos dinâmicos e a tomada de decisões sobre estes; e a aproximação reactiva baseada na rápida reacção a eventos sem a necessidade de manipulação de estruturas simbólicas complexas e apresentando uma maior adaptabilidade, robustez e desempenho.

Esta ideia foi originalmente proposta por Maes em [Mae91] e posteriormente concretizada nas arquitecturas de Touring Machines [Fer92] e InteRRaP [Mnl95]. Ambas as arquitecturas baseiam-se num modelo em camadas. Na camada superior situa-se uma

máquina de processamentos simbólico que providencia suporte aos objectivos mais complexos e abstractos do agente. Por outro lado, na camada inferior encontra-se uma máquina reactiva, responsável pelo tratamento e reacção aos eventos mais elementares que o agente deverá tratar.

Tal como nas anteriores arquitecturas, o número de aplicações existentes é extremamente reduzido e coloca-se o problema da gestão das interacções entre as diferentes camadas.

3.2 AGENTES DE INTERFACE COM O UTILIZADOR

Os agentes de interface com o utilizador, ou apenas agentes de interface, privilegiam as interacções com os (seus) utilizadores e caracterizam-se principalmente segundo a autonomia, a capacidade de memorização e a aprendizagem.

Segundo Maes um agente de interface é descrito como um:

“Assistente pessoal que colabora com o utilizador no mesmo ambiente de trabalho”
[Mae94].

Um agente de interface providencia, desejavelmente de forma pró-activa, assistência ao seu utilizador num determinado contexto. Observa e monitoriza as acções realizadas pelo utilizador e de algum modo vai concebendo o seu perfil.

Um agente com capacidade de aprendizagem baseia-se em um ou mais dos seguintes mecanismos: tratando devidamente as respostas (afirmativas e negativas) dadas pelo utilizador; observando e imitando o utilizador; recebendo instruções/formação explicitamente do utilizador; ou ainda questionando directa ou indirectamente outros agentes.

A área de investigação que trata a problemática da interacção entre o agente e o ser humano designa-se por **interacção social**. A questão principal centra-se no paradigma de conversação entre agentes (computacionais) e seres humanos. Esta questão é importante pois refere-se à visão que o utilizador tem (ou terá) sobre as características e funcionalidades dos seus agentes. O utilizador não se apercebe que os seus agentes são móveis, reactivos, cognitivos, inteligentes, etc. Apenas poderá perceber que uma ou mais aplicações realizam-lhe um conjunto de tarefas específicas, as quais comunicam consigo de forma flexível e natural.

Os mais avançados esforços envolvem interfaces multimodais, ou seja, conversação verbal e não verbal baseadas em linguagem natural, expressões faciais, e outros meios [Mae94, Nor94]. O paradigma da interacção social distingue-se do paradigma de interacção clássica (padrão “pedido-resposta”), pelo facto que a interacção, para além dos mecanismos usuais de decisão e de gestão do conhecimento, baseia-se em informação obtida sensorialmente (e.g., gestos, expressões faciais, sons, visão).

Um exemplo significativo desta área de investigação foi desenvolvido por Nagoa e Takeuchi [NT95] em que um agente apresenta um interface antropomórfico com rosto e capacidades de visão, audição, e fala. Outro exemplo mais recente é o da tecnologia de agentes de software interactivos (baseados na arquitectura COM/Active-X) da Microsoft [Mic96].

A área de aplicação de agentes de interface é grande, variando desde assistentes pessoais que apoiam a utilização de um programa de gestão de recursos de tempo, (e.g. Calendar Agent [KM93]) até aos agentes de pesquisa e filtragem de informação do News da USENET (e.g. NewT [Mae94]). WebWatcher [AFJM95] e Lira [BSY95] são outras aplicações baseadas em agentes de interface.

3.3 AGENTES COLABORATIVOS/COOPERATIVOS

Enquanto que as arquitecturas referidas na secção 4.3.1 focam a atenção sobre a representação e o funcionamento interno do agente (visão interna do agente), a arquitectura de agentes colaborativos/cooperativos, conhecida genericamente por **sistemas multiagente**, foca a sua atenção sobre as múltiplas relações e interacções entre os agentes (visão macro ou social). Estas duas visões em conjunto constituem as principais propostas de teorias de agentes propostas pela comunidade da IAD.

Estes agentes são em geral entidades estáticas e consideravelmente grandes. Podem apresentar diferentes padrões de comportamento e de interacção entre si, nomeadamente cooperação e colaboração [DM90]. Estes padrões levantam respectivamente as questões de coordenação e de negociação de agentes.

- A **cooperação** diz respeito à interacção entre diferentes agentes com vista à concretização de um ou mais objectivos comuns e em que cada agente executa uma tarefa distinta. Uma das questões principais da cooperação diz respeito à garantia de **coordenação** na concretização das tarefas envolvidas. Uma das soluções clássicas baseia-se na existência de um agente supervisor [BP95], responsável pela tarefa de coordenação. Esta solução apresenta contudo a desvantagem de restringir a autonomia dos restantes agentes e de exigir que as tarefas sejam conhecidas previamente à sua realização.
- Na **colaboração**, os agentes realizam uma tarefa comum mas cada um tem objectivos distintos. Este padrão comportamental exige que cada agente tenha uma representação dos objectivos dos outros, assim como a representação de como as suas acções podem influenciar a satisfação dos objectivos alheios. Pelo facto de cada agente ter objectivos distintos, nalguns casos concorrentes, surge a questão da recompensa e da **negociação** entre todos os intervenientes. Por exemplo, em situações em existem vários candidatos para a execução de uma tarefa específica, surge a questão da selecção do melhor candidato, com base na remuneração, currículo do candidato, e

das expectativas de concretização com sucesso envolvidas. Existem diversas propostas de modelização dos comportamentos colaborativos de agentes, quer baseadas em **modelos económicos** [RZ94, Wel93] quer em **modelos sociológicos** [SDCC94].

Existe um número razoável de aplicações e arquitecturas baseadas em agentes colaborativos ou cooperativos. O simulador Phoenix [Coh+89] foi um dos primeiros exemplos de sistema multiagente concebido com o objectivo de validação das teorias de coordenação e cooperação de agentes. O sistema trata um caso real: simulação de incêndios e seu combate. Bombeiros, carros tanque e outras peças de combate a incêndio são modelizadas como agentes, e todos eles cooperam na solução do mesmo objectivo, que é a extinção do incêndio. Os diferentes agentes actuam e comunicam entre si (consoante as suas capacidades) por diferentes meios. O sistema mantém informação dos efeitos causados pelos agentes, o estado do terreno e do incêndio.

Outro exemplo é o simulador TruckWorld [FH87], que modeliza um sistema de transporte rodoviário de mercadorias. Os camiões são agentes que percorrem uma rede viária minada por bombas e outros imprevistos similares. Tem diversos objectivos, tais como abastecimento de alimentos às populações, abastecimento de combustível, carregamento e descarregamento de mercadorias. O objectivo deste sistema foi de testar teorias sobre a reactividade de agentes e identificação de características necessárias à sua vivência em ambientes complexos, dinâmicos e não determinísticos.

O projecto Pleides [SDCC94] no domínio da tomada de decisões organizacionais, ou o ADEPT [OW96] na área de reengenharia dos processos de negócios, são outros exemplos mais recentes desta classe de agentes.

A principal crítica aos sistemas multiagentes reside no facto de serem adoptados modelos baseados no paradigma do pensamento simbólico (ou seja, da IA clássica) os quais produzem aplicações/protótipos pouco flexíveis e com ineficiente desempenho. Outras críticas também dirigidas devem-se à inexistência de teorias consensuais e adequadas à especificação dos mecanismos de coordenação e negociação entre agentes.

3.4 SISTEMAS DE AGENTES MÓVEIS

Agentes móveis são processos de software cuja característica principal é a capacidade de se transportarem entre diferentes nós de uma rede, ou de diferentes redes, como no caso da Internet. As arquitecturas de agentes móveis têm sido desenvolvidas pela comunidade de SD e correspondem a extensões ao paradigma clássico da computação Cliente/Servidor (vid. Capítulo 3), suportadas pelas seguintes razões/vantagens [HCK95]:

- Comunicação ponto-a-ponto (*peer-to-peer*). Uma limitação efectiva do modelo cliente/servidor reside nos comportamentos restritivos dos dois

componentes: os clientes tomam sempre a iniciativa das comunicações/computações (são pró-activos); enquanto que os servidores apenas respondem e actuam na sequência de solicitações dos primeiros (são passivos). Os agentes, em oposição, apresentam em diferentes momentos do seu ciclo de vida características de cliente e de servidor.

- **Eficiência.** O facto de um agente se poder transportar para o local onde o(s) recurso(s) exista(m) permite a redução do tráfego de rede desde que este tenha capacidade de decisão e de escolha da informação que é efectivamente relevante transferir para o local original. Este aspecto é particularmente importante num conjunto variado de aplicações e situações, designadamente sobre redes móveis (*wireless networks*), de baixo débito e de alta latência.
- **Comunicação assíncrona.** Um agente uma vez lançado para outro nó, torna-se independente do nó original. Caso este nó fique desconectado durante algum tempo, não inviabiliza a execução do agente referido. Esta característica é conveniente para utilizadores de computadores móveis que ligam, enviam o agente para a rede e desligam de seguida a sua máquina. Mais tarde esses utilizadores voltam a ligar os seus computadores de forma a testar o(s) resultado(s) do agente previamente lançado e/ou para lançar novos agentes.
- **Tolerância a falhas.** No modelo C/S o estado de uma transacção encontra-se partilhado entre cliente e servidor. Na existência de uma falha durante a execução de um pedido na rede ou no servidor, torna-se difícil recuperar o estado da transacção abortada e reiniciar-se o processo de sincronização entre cliente e servidor, pelo facto da conexão de transporte (de rede) ter-se perdido. Em oposição, os agentes móveis, por não necessitarem de manter conexões permanentes e manterem o seu estado dentro deles próprios, facilitam em geral o tratamento de falhas a diferentes níveis.

Recentemente e no contexto das redes de telecomunicações e da Internet têm surgido e continuam a surgir várias propostas de sistemas de agentes móveis. Telescript [Whi94], Aglet [IBM96], Agent-Tcl [Gra95] e Mole [SBH96] são alguns dos sistemas (quer industriais/comerciais, quer de investigação) mais conhecidos e promissores.

3.5 AGENTES DE INFORMAÇÃO

O objectivo genérico de um agente de informação é esconder ao seu utilizador a complexidade e heterogeneidade do acesso à informação envolvida, de interagir com o maior número de recursos de informação disponíveis, e de providenciar mecanismos de partilha e divulgação de informação.

O termo de “agente de informação” não se encontra ainda suficientemente maduro já que diferentes

sistemas e aplicações com características distintas adoptam essa designação.

Os exemplos mais simples desta classe de agentes consistem em processos de pesquisa de informação num espaço de procura mais ou menos conhecido (e.g., Web, News da Usenet, bases de dados e/ou de conhecimento específicas) e com base num conjunto de temas ou palavras-chave que o utilizador implícita ou explicitamente define. Informant [Inf96] é um agente de informação com procura no espaço da Web com base num conjunto de palavras-chave que o utilizador explicitamente indica. Adicionalmente o agente providencia um mecanismo de notificação de alterações de documentos na Web – o utilizador apenas tem de indicar os URLs relativamente aos documentos que pretende ser notificado. Softbot [EW94] é um agente que permite que o utilizador especifique um pedido de alto nível, com base num conjunto hierárquico de menus, tal como “*obter todos os relatórios técnicos e publicações de Alberto Silva que não se encontram guardados localmente na máquina cupido.inesc.pt*”. O softbot inspirou-se na metáfora do robot físico (suportado por comandos e aplicações usuais no ambiente Unix) com os conceitos de sensores (e.g., clientesarchie, gopher, www, netfind) e actuadores (e.g., mv, compress, mkdir). O softbot define automaticamente um plano de execução de forma a concretizar o pedido de alto nível especificado e utiliza para tal o seu conjunto de sensores e de actuadores disponíveis.

Outros sistemas de agentes providenciam mecanismos de procura de informação com base na aproximação designada por **filtragem social**. Estes sistemas funcionam em contextos restritos (e.g., recomendação de filmes, músicas) e têm como objectivo a identificação dinâmica dos seus utilizadores e a criação de grupos de utilizadores com interesses comuns. Em cada grupo, a informação relevante para um determinado utilizador é sugerida ou recomendada automaticamente a todos os demais elementos do grupo envolvido. Firefly [Fir97], ou Jasper [DW95] são exemplos desta última aproximação.

4 DISCUSSÃO

O conceito e modelo agentes de software parece ser bastante adequado à concepção e construção de sistemas computacionais complexos, distribuídos ou heterogéneos.

O conceito original de agente introduzido por Hewitt há já 20 anos foi adoptado desde então pela comunidade científica da IAD a qual propôs um conjunto de modelos e teorias relativamente complexas e elaboradas, nomeadamente associadas às arquitecturas de agentes intencionais, reactivos, híbridos e dos sistemas multiagente. Contudo, e tal como sucedeu à generalidade dos trabalhos de investigação da IA, as suas propostas – baseadas em modelos de representação simbólica de conhecimento – patentearam sérias limitações de implementação, designadamente limitações ao nível da própria

capacidade de desenvolvimento das aplicações (inexistência de técnicas e ferramentas de suporte adequado), ao nível do desempenho e da escalabilidade das aplicações e protótipos desenvolvidos. A generalidade das aplicações de agentes baseadas nos modelos de representação simbólica de conhecimento da IA ou são protótipos ou simuladores em contextos restritos e limitados. Existe um número reduzido de aplicações efectivamente funcionais (por exemplo na área de controlo industrial) mas cujo desenvolvimento foi baseado em linguagens de 3ª geração (e.g., Pascal, C, Prolog) e por conseguinte extremamente complexo e demorado.

A liberalização das telecomunicações, da explosão da Internet, e o sucesso da Web à escala global veio suscitar o interesse e o esforço da comunidade de SD pelos agentes, como entidades/objectos de software caracterizados principalmente pela autonomia e mobilidade. Com abordagens, técnicas e teorias distintas espera-se, neste contexto, que a investigação e o esforço industrial nos agentes de software seja relançado e permita a concepção e construção das futuras aplicações centradas na rede.

Parece-nos contudo que a capacidade tecnológica da indústria e da comunidade de SD deverá ser com vantagem complementada pelos modelos e teorias desenvolvidas pela comunidade da IAD, designadamente no que respeita aos modelos de coordenação e negociação dos sistemas multiagentes, bem como as propostas de linguagens de comunicação dos agentes heterogéneos (e.g., KQML),

5 REFERÊNCIAS

- [AFJM95] R. Armstrong, D. Freitag, T. Joachims, T. Mitchell. WebWatcher: A Learning Apprentice for the WWW. *Working Notes of the AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments*. 1995.
- [Bat94] J. Bates. The role of emotion in believable agents. *Communications of ACM*, 37(7):122-125, 1994.
- [BG88] A. Bond, L. Gasser. An Analysis of Problems and Research in DAI. *Readings in Distributed Artificial Intelligence*, A. Bond, L. Gasser (eds.), Morgan Kaufmann, 1988.
- [BP95] P. Burkle, P. Prosser. The distributed Asynchronous Scheduler. *Intelligent Scheduling*. M. Zweben, M. Fox (eds.), Morgan Kaufmann, 1995.
- [BSY95] M. Balabanovic, Y. Shoham, Y. Yum. An Adaptive Agent for Automated Web Browsing. *Journal of Image Representation and Visual Communications*, Vol.6. 1995.
- [BTV96] J. Baumann, C. Tschudin and J. Vitek. *Proceedings of the 2nd ECOOP Workshop on Mobile Object Systems* (Linz, Austria, July 8-9, 1996). Dpunkt. 1996.
- [Car94] Luca Cardelli. Obliq: A Language with distributed scope. *Digital White Paper*, Digital Equipment Corporation, Systems Research Center, 1994.
- [Coh+89] Cohen et al. Trial by Fire: Understanding the Design Requirements for Agents in Complex Environments. *AI Magazine*. 10(3):32-48, 1989.
- [CTV96] Paolo Ciancarini, Robert Tolksdorf, Fabio Vitali. PageSpace: An architecture to coordinate distribute applications on the web. *Computer Networks and ISDN Systems*. Vol. 28, pages 941-952, 1996.
- [DM90] Y. Demazeau, J. Muller. Decentralized Artificial Intelligence. *Decentralized Artificial Intelligence*, Y. Demazeau, J. Muller (eds.), Elsevier, 1990.
- [DW95] J. Davies, R. Weeks. Jasper: Communicating Information Agents. *Proceedings of the 4th Int'l Conference on WWW*. 1995.
- [EW94] O. Etzioni, D. Weld. A Softbot-Based Interface to the Internet. *Communications of the ACM*, 37(7):72-76, July 1994.
- [FG96] S. Franklin, A. Graesser. Is it an Agent, or Just a Program?: A Taxonomy for Autonomous Agents. *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*. Springer-Verlag. 1996.
- [FH87] R. Firby, S. Hanks. A Simulator for Mobile Robot Planning. *Proceedings of the DARPA Knowledge-based Planning Workshop*. 1987.
- [Fir97] Firefly Inc. 1997.
<http://www.firefly.com>
- [Gil+95] D. Gilbert, et al., IBM. *The Role of Intelligent Agents in the Information Infrastructure*. 1995.
<http://activist.gpl.ibm.com:81/WhitePaper/ptc2.htm>
- [GP94] M. Genesereth, S. Ketchpel. Software Agents. *Communications of the ACM*, 37(7):48-53, July 1994.
- [Gra95] Robert Gray. Agent Tcl: A transportable agent system. *Proceedings of the CIKM Workshop on Intelligent Information Agents*, (CIKM'95), 1995.
- [HCK95] C. Harrison, D. Chess, A. Kershenbaum. *Mobile Agents: Are they a good idea?*. IBM, 1995.
- [Hew77] C. Hewitt. Viewing Control Structures as Patterns os Passing Messages. *Artificial Intelligence*. 8(3):323-364, 1977.
- [IBM96] IBM Tokyo Research Laboratory. *Aglets workbench: Programming mobile agents in Java*. <http://www.trl.ibm.co.jp/aglets>. 1996.
- [Inf96] The Informant. 1996.
<http://informant.dartmouth.edu/>
- [Jen92] N. Jennings. On being Responsible. *Proceedings of the 3rd European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds*. Eds. E. Werner, Y. Demazeau. 1992.
- [KM93] R. Kozierok, P. Maes. A Learning Interface Agent for Scheduling Meetings. *Proceedings of the ACM-SIGCHI Intelligent Workshop on Intelligent User Interfaces*. 81-93.1993.
- [Len95] B. Lenat. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM*, 38(11):33-38, Nov. 1995.
- [LF94] Y. Labrou, T. Finn. A semantic approach for KQML — a general purpose communication language for software agents. *Proc. of Third International Conference on Information and Knowledge Management (CIKM'94)*, 1994.

<http://www.cs.umbc.edu/kqml/papers/>

- [Mae94] P. Maes. Agents that Reduce Work and Information Overload. *Communications of the ACM*, 37(7):31-40, July 1994.
- [Mic96] Microsoft. *ActiveX Technology for Interactive Software Agents*. 1996.
<http://www.microsoft.com/intdev/agent/agent.htm>
- [Nor94] D. Norman. How Might People Interact with Agents. *Communications of the ACM*, 37(7):68-71, July 1994.
- [NT95] K. Nagoa, A. Takeuchi. Social Interaction: Multimodal Conversation with Social Agents. 1995
- [Nwa96] H. Nwana. Software Agents: An Overview. *Knowledge Engineering Review*. 11(3), 1-40. Cambridge University Press, 1996.
- [OW96] P. O'Brien, M. Wiegand. Agents of Change in Business Process Management. *BT Technology Journal*, 14(4). 1996.
- [RG95] A. Rao, P. Georgeff. BDI Agents: from Theory to Practice. *Proceedings of the First International Conference on Multi-Agent Systems*. 1995.
- [RZ94] G. Rosenschein, J. Zlotkin. Designing Conventions for Automated Negotiation. *AI Magazine*. 15(3):29-46, 1994.
- [SBH96] M. Strasser, J. Baumann and F. Hohl. Mole: A Java-based mobile object system. In [BTV96].
- [SDCC94] J. Sichman, Y. Demazeau, R. Conte, C. Castelfranchi. A Social Reasoning Mechanism Based on Dependence Networks. *Proceedings of 11th European Conference of Artificial Intelligence*. A. Cohen (ed.), John Willey & Sons, 1994.
- [Sea69] J. Searle. *Speech Acts*. Cambridge University Press. 1969.
- [Sho93] Y. Shoham. Agent Oriented Programming. *Artificial Intelligence*. 60(1):51-92, 1993.
- [VB90] S. Vere, T. Bickmore. A Basic Agent. *Computational Intelligence*, 6:41-60.1990.
- [Wei93] M. Wellman. A Market-Oriented Programming Environment and its Application to Distributed Multicommodity Flow Problems. *Journal of AI Research*. (1):51-92, 1993.
- [Whi94] James White. Telescript technology: The foundation for the electronic marketplace. *General Magic White Paper*, General Magic, 1994.
- [WJ95] M. Wooldridge, N. Jennings. Intelligent Agents: Theory and Practice. *Knowledge Engineering Review*. 10(2), 115-152. Cambridge University Press, 1995.