

# Aplicando Agentes Móveis em Data Warehouses

Márcio Gonçalves, Olinto José Varela Furtado

Departamento de Informática e de Estatística – INE  
Programa de Pós-Graduação em Ciência da Computação – PPGCC  
Universidade Federal de Santa Catarina (UFSC)  
Florianópolis, SC – Brasil

marcio@datainfo.inf.br, olinto@inf.ufsc.br

**Abstract.** *The objective of this article is to show a solution which uses the mobile agent technology in order to improve the Extraction, Transformation and Load process in data warehouse. The theory described in this article, show us some aspects related to data warehouses and software agents. The main objective is to show the viability of the mobile agent technology during the development process and maintenance of the information in the data warehouses.*

**Resumo.** *Este artigo tem como objetivo apresentar uma solução que utiliza a tecnologia de agentes móveis para melhorar o processo de Extração, Transformação e Carga (ETL - Extraction, Transformation and Load) em data warehouses. A fundamentação descrita neste artigo apresenta aspectos relativos à data warehouses e agentes de software. O objetivo principal é demonstrar a viabilidade do uso da tecnologia de agentes móveis no processo de elaboração e manutenção da informação nos data warehouses.*

## 1. Introdução

Nos últimos anos, várias técnicas e conceitos foram elaborados com o objetivo de tentar facilitar a extração de informações nas grandes bases de dados existentes nas organizações. Dentre as técnicas e conceitos mais conhecidos estão os *data warehouses*, que têm por objetivo disponibilizar os dados em uma modelagem de fácil entendimento para os usuários.

Atualmente, tivemos um aumento significativo de empresas buscando desenvolver seus sistemas de apoio à decisão baseados em *data warehouse*. Contudo, as empresas que já investiram nestes tipos de sistemas ainda sofrem dificuldades em obter os resultados esperados simplesmente pela falta de ferramentas que possam integrar as diversas fontes de dados existentes dentro das organizações. Os fornecedores de *software* que atuam nesta área preocuparam-se em desenvolver as ferramentas finais para os usuários, mas esqueceram de tratar a questão da integração de dados, um requisito para os *data warehouses* e algo que somente as ferramentas ETL podem atender.

Este trabalho descreve a proposta e implementação de uma ferramenta de integração de dados para *data warehouse* baseada na tecnologia de agentes móveis, denominada DWE [Gonçalves, 2002]. Não é de nosso conhecimento a existência de ferramenta ETL similar no mercado que também utilize os conceitos e a tecnologia de agentes móveis na solução dos problemas de extração, transformação e carga de dados.

## 2. Data Warehouse

*Data warehouse* não é um novo conceito, pois foi originalmente apresentado como uma proposta de solução pela IBM, chamada de “*information warehouse*” e relançado várias vezes até hoje.

Em uma definição simples, um *data warehouse* pode ser considerado como a separação física entre os sistemas de dados transacionais e os sistemas de suporte à decisão em uma organização [Singn 2001].

### 2.1. Propriedades de um Data Warehouse

Segundo [Inmon 1992], um *data warehouse* deve apresentar as seguintes propriedades, ou seja, ele deve ser:

**Orientado a Assunto:** os *data warehouses* são projetados para ajudar uma organização a analisar os seus dados e a forma como estes dados são implementados categoriza-os como orientados a assunto.

**Integrado:** os *data warehouses* necessitam armazenar os dados coletados de diferentes fontes em um formato consistente. Contudo, para que os dados coletados tornem-se realmente integrados aos dados já existentes no *data warehouse*, estes precisam ser previamente tratados antes de serem armazenados de forma definitiva.

**Não Volátil:** o fato de um *data warehouse* não ser volátil significa que os dados uma vez carregados nunca mais são alterados pelos usuários, apenas consultados. As alterações que são comuns e ocorrem nos sistemas de dados transacionais não implicam necessariamente em alterações no *data warehouse*, mas sim, em novas cargas de dados.

**Variável com o Tempo:** nos sistemas convencionais os usuários geralmente possuem somente a última visão dos dados, não podendo visualizar qual era a situação de uma determinada informação antes da última atualização. Em um *data warehouse* isto é possível, pois cada alteração implica em uma nova entrada de dados, mapeando desta forma o histórico das alterações.

São estas as propriedades que diferem um *data warehouse* dos sistemas convencionais de apoio à decisão.

### 2.2. Componentes de um Data Warehouse

A construção dos *data warehouses* pode diferir em estrutura e ou características de implementação. Contudo, independente das diferenças de implementação encontradas, alguns componentes são familiares em qualquer *data warehouse*. Segundo [Singn 2001], um *data warehouse* sempre apresenta os seguintes componentes:

**Dados Atuais:** os dados atuais são sem dúvida os que exigem mais atenção em um *data warehouse*, pois estes refletem os acontecimentos mais recentes, sempre de grande interesse para a organização.

**Dados Antigos:** os dados antigos, também conhecidos como históricos, são acessados com menor frequência e armazenados em nível de detalhe consistente com o detalhe dos dados atuais. Os dados históricos são geralmente armazenados em um meio alternativo devido ao grande volume de dados conjugado ao raro acesso.

**Dados Sumarizados:** os dados sumarizados são criados a partir dos dados detalhados e têm como objetivo melhorar o tempo de resposta das consultas dos usuários. Segundo [Singh 2001], a maior parte das consultas executadas pelos usuários no *data warehouse* são feitas utilizando dados sumarizados.

**Metadado:** o metadado é um tipo de dado muito importante, pois o mesmo ajuda a identificar e a localizar os demais dados no *data warehouse*. Os metadados além de manter informações sobre a localização dos dados também podem conter históricos de extração e transformação de dados, estatísticas de uso das informações, informações do espaço utilizado pelos objetos no *data warehouse* entre outras.

### 3. Agentes

Nos últimos anos, a proposta de desenvolver aplicativos utilizando a tecnologia de agentes conseguiu obter altos níveis de sucesso. Produtos de *software* em diversas áreas da informática estão disponíveis no mercado e seus fornecedores alegam possuir diferencial por estarem utilizando esta tecnologia.

O objetivo desta seção é prover informações inerentes ao conceito de sistemas baseados em agentes para que o leitor possa entender de forma mais clara a proposta de utilizar a tecnologia de agentes móveis como ferramenta para manter a arquitetura de um *data warehouse*.

#### 3.1. Características Encontradas nos Agentes

Os agentes possuem atributos específicos que modelam seu comportamento, estes atributos são as características que os diferenciam de um simples programa de computador. A seguir são descritas algumas características importantes que geralmente são encontradas nos agentes, conforme [Genesareth e Ketchpel 1994]:

**Autonomia:** agentes têm a capacidade de executar ações sem a intervenção direta de operadores. Um agente sempre possui algum tipo de controle sobre suas ações e seu estado interno. Eles podem decidir o que fazer quando a tarefa é executada com sucesso ou com falha.

**Habilidade Social:** para conseguir seus objetivos os agentes interagem com outros agentes. A habilidade social é a aptidão dos agentes em se comunicar a fim de conseguir recursos para concluir suas tarefas, ou para fornecer auxílio aos outros agentes.

**Reatividade:** os agentes detectam seu ambiente e respondem aos estímulos dele recebido. Em alguns casos os agentes podem ficar adormecidos até que certas mudanças ocorram no ambiente.

**Mobilidade:** é a habilidade de um agente mover-se em uma rede, ocupando diferentes nós e recursos ao longo de sua execução. De acordo com [Jennings e Wooldridge 1998], a característica de mobilidade permite a um agente ser transmitido de um lugar para outro, executar suas tarefas dentro de um ambiente virtual em uma locação remota qualquer, voltando em seguida ao seu endereço de origem.

**Racionalidade:** a habilidade de raciocinar durante a execução de uma tarefa é um dos aspectos chave da inteligência de um agente. Raciocinar implica em dizer que

um agente tem habilidade para deduzir e extrapolar, baseando-se para isto, em conhecimento atual e em experiências.

**Adaptabilidade:** um agente deve ter a capacidade de ajustar-se aos hábitos, métodos de trabalho e preferências de seus usuários. Segundo [Bond e Gasser 1988], um agente apresenta adaptabilidade quando possui a habilidade de modificar sua própria configuração, de modo a atender da melhor forma possível os requisitos do sistema.

### 3.2. Benefícios com a Utilização dos Agentes

Conforme [Bond e Gasser 1988], existem diversos benefícios em utilizar aplicações baseadas em agentes para implementar soluções distribuídas, dentre estes benefícios destacamos:

**Desenvolvimento e Gerenciamento:** a inerente modularidade dos sistemas multi-agentes permite o desenvolvimento das partes do mesmo de forma independente e paralela.

**Eficiência e Velocidade:** a concorrência e a distribuição das entidades computacionais em diferentes máquinas pode aumentar a velocidade de processamento pelo fato do paralelismo na execução de tarefas.

**Integração:** os agentes permitem a integração de recursos distribuídos e até mesmo heterogêneos tais como *hardware* e *software* de diferentes plataformas.

**Segurança:** o controle dos processos locais pode ser encarado como uma maneira de proteção ou de aumento de segurança do sistema.

**Naturalidade:** alguns problemas são mais naturalmente resolvidos com a ajuda de uma configuração distribuída.

**Confiabilidade:** os sistemas distribuídos podem exibir um grau maior de confiabilidade e de segurança quando comparados a um sistema tradicional de processamento centralizado, pois podem prever redundância de dados e prover múltiplas verificações.

**Bom aproveitamento de recursos:** os agentes computacionais individuais, mesmo que ligados a recursos escassos, podem superar e resolver problemas complexos.

## 4. Data Warehouse Extractor

A ferramenta Data Warehouse Extractor apresentada nesta seção é o resultado de um grande esforço executado com o objetivo de implementar uma ferramenta ETL que possa ser utilizada para construir *data warehouses* com maior facilidade.

O Data Warehouse Extractor (DWE) [Gonçalves, 2002] permite extrair informações de múltiplas fontes de dados e disponibiliza recursos para executar o transporte e a carga destas informações para os *data warehouses*.

Para atender aos requisitos de uma ferramenta de coleta de dados (ETL), o DWE é composto por um Grupo de Agentes Cooperantes e por um Módulo de Gerenciamento de Agentes. Neste artigo, estaremos discutindo somente o módulo do Grupo de Agentes Cooperantes, dado que a aplicação de agentes móveis é destinada a este módulo.

O Grupo de Agentes Cooperantes (GAC) contém diversos agentes responsáveis pela execução das várias atividades existentes no processo de coleta de dados. Atividades como a extração, criptografia, compactação, transporte e recepção dos dados são executadas por agentes contidos neste grupo.

A figura 1 ilustra de uma forma resumida o fluxo do processo de coleta de dados padrão, para que o leitor possa visualizar de uma maneira geral a funcionalidade do sistema (as atividades dos agentes e a arquitetura do GAC). A figura ilustra o fluxo de comunicação entre os agentes e o estado dos dados coletados em cada etapa.

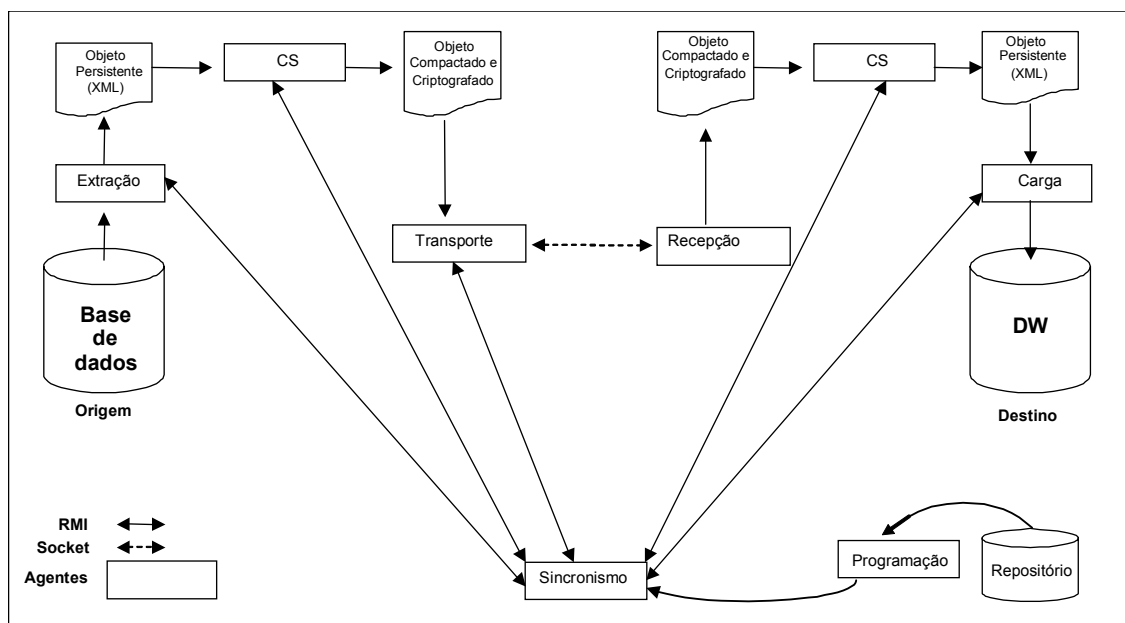


Figura 1 – Fluxo do Processo de Coleta dos Dados

#### 4.1. Agentes de Extração

Os agentes de extração têm a responsabilidade de executar o processo primário da coleta de dados. São estes os agentes que extraem informações das diversas fontes de dados e convertem estes dados para um formato conhecido por todos os agentes.

A extração de dados é um processo complexo devido às diversas tecnologias de armazenamento de dados existentes na atualidade. Nas organizações, as informações podem ser encontradas nas mais variadas fontes, como por exemplo, em bancos de dados relacionais, em arquivos textos, arquivos binários, planilhas eletrônicas, banco de dados orientado a objetos, etc.

Dentro deste contexto, é possível dizer que a extração de dados é um ponto chave no processo de coleta de dados e que compromete o potencial de muitas ferramentas ETL existentes no mercado. A vantagem do DWE é que o processo de extração de dados é baseado em agentes, o que possibilita que novos agentes possam ser construídos gradativamente (inclusive por outros fornecedores) e adicionados ao Grupo de Agentes Cooperantes.

## 4.2. Agentes de Compactação e Segurança

Os agentes de compactação e segurança (CS) têm como responsabilidade executar as funções relativas à compactação e segurança dos dados, após o processo de extração e antes do processo de carga (o agente de carga será visto na seção 4.5).

Estes agentes, quando utilizados após os agentes de extração, são responsáveis por executar a criptografia e a compactação dos dados extraídos, deixando, desta forma, os dados prontos para serem transportados pela rede até o *data warehouse*. De outra forma, quando os agentes CS são executados antes dos agentes de carga, estes têm a função inversa que é desfazer a compactação e desfazer a criptografia dos dados que chegaram ao *data warehouse*, de modo que o agente de carga possa utilizá-los.

O processo de compactar e descompactar os dados é essencial para o transporte de grandes quantidades de informação, fato comum quando se trata de *data warehouse*.

## 4.3. Agentes de Transporte

Os agentes de transporte são uma classe de agentes contida dentro do GAC que tem como responsabilidade transportar para o *data warehouse* os dados extraídos das bases transacionais. Estes agentes instanciam os objetos persistentes que foram anteriormente tratados pelos agentes CS e os transferem para o agente de recepção que fica localizado no computador onde se encontra o *data warehouse* (o agente de recepção está descrito na próxima seção).

Uma das grandes vantagens que destaca os agentes de transporte é a habilidade que estes agentes possuem na retransmissão de dados (em conjunto com os agentes de recepção). Ou seja, quando um agente de transporte não consegue finalizar a transmissão de dados para o *data warehouse* por algum problema de comunicação na rede, este aborta a transmissão e armazena a posição do último registro de dados que foi enviado com sucesso. Desta forma, o agente aguarda até a comunicação da rede se normalizar e logo após reinicializa a transmissão dos dados a partir do ponto em que foi interrompido.

## 4.4. Agentes de Recepção

Como citado na seção anterior, o agente de recepção tem como responsabilidade receber os dados enviados pelos agentes de transporte e descarregá-los novamente como objetos persistentes agora no lado do *data warehouse*. Os agentes de recepção precisam trabalhar o tempo todo em sincronia com os agentes de transporte. Caso contrário, não é possível transportar para o *data warehouse* os dados extraídos das bases transacionais. Um agente de recepção pode receber dados de vários agentes de transporte ao mesmo tempo (conceito de *thread*).

Um bom modelo para implementar a comunicação entre os agentes de transporte e de recepção, é a tecnologia *server socket*. Esta tecnologia permite que estes agentes consigam fazer a transferência de grandes quantidades de informações (exemplo, os dados extraídos das bases de dados) de forma simples e transparente, permitindo inclusive uma maior independência entre os agentes.

#### 4.5. Agentes de Carga

Os agentes de carga têm como responsabilidade executar o processo final de uma coleta de dados (no que se refere às etapas de coleta de dados da ferramenta DWE). São estes os agentes que tem a finalidade de disponibilizar as informações coletadas no *data warehouse*.

Os agentes de carga possuem restrições de implementação muito parecidas com as dos agentes de extração, pois este tipo de agente também necessita trabalhar com diversas bases de dados de formatos variados. Ou seja (um caso prático para um melhor entendimento), em uma organização o *data warehouse* pode ser instalado em um banco de dados relacional, enquanto em outra organização este pode ser instalado em um banco de dados orientado a objetos.

#### 4.6. Agentes de Sincronismo

Os agentes de sincronismo são agentes que têm como responsabilidade determinar a ordem de execução dos demais agentes (coordenar). Em cada processo de carga de dados que ocorre no *data warehouse* são os agentes de sincronismo que invocam cada um dos agentes configurados na carga. Ou seja, são os agentes de sincronismo que solicitam a transferência dos agentes (mobilidade) para o computador onde o trabalho deve ser executado.

O RMI (*Remote Method Invocation*, da linguagem Java) é uma boa tecnologia para os agentes de sincronismo utilizarem para poder conversar com os demais agentes. O RMI permite que os agentes de sincronismo possam coordenar e distribuir atividades para os demais agentes de forma simples.

#### 4.7. Agentes de Programação

Os agentes de programação, também conhecidos como agentes *schedule*, são agentes que tem como responsabilidade disparar, nos horários programados, as cargas de dados configuradas. São os agentes de programação que identificam o horário para iniciar o processo de qualquer carga (através de uma monitoração contínua no repositório de dados) e os responsáveis por instanciar os agentes de sincronismo.

**Tabela 1 – Tipos de Programação Disponíveis**

<b>Tipo de Programação</b>	<b>Aplicação</b>
1 – uma vez	Disparar uma carga de dados para que execute somente uma vez em um horário agendado.
2 - a cada n minutos	Disparar uma carga de dados a cada <b>n</b> minutos.
3 - a cada n horas	Disparar uma carga de dados a cada <b>n</b> horas.
4 - a cada n dias	Disparar uma carga de dados a cada <b>n</b> dias.
5 – semanalmente	Disparar uma carga de dados em alguns dias da semana (exemplo: segunda, terça e quinta).
6 - a cada n meses	Disparar uma carga de dados a cada <b>n</b> meses.
7 - a cada n anos	Disparar uma carga de dados a cada <b>n</b> anos.

Os agentes de programação disponibilizam na ferramenta DWE uma grande variedade de tipos de programação de cargas. A tabela 1 mostra possíveis maneiras de se programar uma carga de dados.

## 5. Considerações Finais

Pesquisar sobre um assunto interessante e cogitado pelas grandes organizações, torna o estudo de *data warehouse* altamente estimulante. As organizações já reconhecem a necessidade de investir em *data warehouse* para que possam agregar mais informação aos seus sistemas de apoio à decisão.

Integrar todos os dados de uma organização não é uma tarefa fácil. Contudo, através do estudo realizado e dos experimentos implementados, constatamos que o uso da tecnologia de agentes móveis neste tipo de atividade aumenta as chances de sucesso, visto que tal tecnologia responde adequadamente às características deste tipo de aplicação e contribui significativamente na solução de muitos problemas tipicamente encontrados em ferramentas similares.

A ferramenta Data Warehouse Extractor (DWE) apresentada neste artigo foi fundamental para mostrar a viabilidade do uso de agentes na carga de data warehouses. Ótimos resultados foram obtidos tanto em facilidades de implementação quanto em facilidades de incorporação de novos componentes. Os experimentos realizados envolveram diferentes fontes de dados (arquivos de texto, planilhas e bases de dados distintas) e diferentes ambientes operacionais [Gonçalves, 2002].

Embora tenhamos empregado um grande esforço neste trabalho, o assunto não foi esgotado, muitas pesquisas e muitos trabalhos estão por ser feitos nesta área. Estamos atualmente em fase de definição de um produto derivado do protótipo implementado.

## Referências

- Bond, Alan H. e Gasser, Les (1988). “*Readings in Distributed Artificial Intelligence*”, Morgan Kaufmann Publishers, San Mateo-California.
- Genesareth, M. e Ketchpel, S. (1994). “*Software Agents. Communications of ACM*”, n.7, v.4, p.48-53.
- Gonçalves, M. (2002). “*Uma Ferramenta para Extração de Dados para Data Warehouse baseada em Agendas Distribuídos*”, Dissertação de Mestrado, CPGCC, UFSC, Florianópolis, SC.
- Inmon, Willian H. (1992). “*Como Construir o Data Warehouse*”, Tradução: Ana Maria Neto Guz, Rio de Janeiro, Campus.
- Jennings, Nicholas R. e Wooldridge, Michael J. (1998). “*Agent Technology – Foundations, Applications and Markets*”, Springer, New York.
- Singn, Harry S. (2001). “*Data Warehouse: Conceitos, Tecnologias, Implementação e Gerenciamento*”, Tradução: Mônica Rosemberg, Revisão Técnica: José Davi Furlan, São Paulo, Makron Books.