

## 2 Integração de Dados

A integração de dados distribuídos e heterogêneos tornou-se um requisito essencial para a sociedade de informação desde os primeiros anos de desenvolvimento da ciência da computação. A necessidade de integração de dados deu origem a diferentes arquiteturas de bancos de dados. Com o advento da Web novas propostas de integração de dados estão sendo desenvolvidas.

Neste capítulo apresentamos os conceitos sobre a questão da integração de dados em sistemas de bancos de dados distribuídos e heterogêneos. O problema da interoperabilidade é abordado e são relacionados os tipos de heterogeneidades que podem aparecer na tentativa de promover esta interoperabilidade de acordo com a definição de (Sheth & Larson, 1990) - heterogeneidade entre SGBDs, estrutural, sintática e semântica.

É apresentada a visão de ontologias para representação de bases de conhecimento. A abordagem de federação de dados é apresentada como estratégia para promover a interoperabilidade entre as bases de dados distribuídas que estarão representadas por suas respectivas ontologias.

### 2.1 Interoperabilidade semântica

Para que se possa promover a integração entre várias fontes de informação, deve-se primeiro ter acesso aos dados que estão armazenados nas mesmas. Isso significa que estas fontes devem trabalhar juntas, fornecendo uma visão compartilhada de seus conteúdos. O problema de estabelecer essa comunicação entre as fontes de informação heterogêneas e distribuídas é conhecido como problema de interoperabilidade. A interoperabilidade deve ser atingida tanto no nível técnico quanto no nível da informação propriamente dita. De modo resumido, para se realizar o compartilhamento dos dados não é necessário que eles sejam apenas acessados, mas também que os mesmos possam ser integrados e processados por outros sistemas. Os problemas que podem surgir devido à heterogeneidade dos dados são bem conhecidos pela comunidade de banco de

dados distribuídos, são eles: heterogeneidade entre SGBDs, heterogeneidade estrutural, heterogeneidade sintática e heterogeneidade semântica.

## **2.2 Heterogeneidades entre Bancos de Dados**

Heterogeneidade em bancos de dados está ligada ao uso de diferentes modelos e tecnologias (Sheth & Larson, 1990). Essas diferenças podem ser de hardware, diferenças de comunicação ou ainda diferenças de software. Diferenças de software podem estar associadas ao uso de diferentes sistemas operacionais em um mesmo ambiente ou ainda diferentes SGBDs comerciais que estão disponíveis e são adotados em uma mesma empresa com o objetivo de facilitar o desenvolvimento dos sistemas assim como a adequação dos requisitos do negócio e que, podem estar distribuídos rodando em diferentes sistemas operacionais em diferentes versões. A comunidade de banco de dados tem se empenhado em resolver estes problemas durante anos classificando estas heterogeneidades e buscando soluções específicas para cada uma delas. A classificação que daremos para os diversos tipos de heterogeneidades segue a definição de (Sheth & Larson, 1990).

### **2.2.1 Heterogeneidade quanto a SGBDs**

Uma mesma empresa com diferentes requisitos a serem atendidos pode necessitar de diferentes SGBDs a fim de atender estas necessidades. Neste caso a heterogeneidade está relacionada às diferenças entre os diversos modelos de dados e sistema. Cada SGBD tem um modelo de dados próprio usado para definir estruturas de dados e regras. Devem ser considerados ainda aspectos de linguagens diferentes em que é desenvolvido cada SGBD. Desta forma podemos destacar diferenças que podem ser encontradas em SGBDs convivendo num mesmo ambiente e que podem trazer heterogeneidades.

### **2.2.2 Heterogeneidade Estrutural**

Heterogeneidade estrutural está relacionada ao fato de que diferentes fontes de informação armazenam seus dados em estruturas com esquemas distintos. Diferenças entre estruturas são provenientes de diferentes modelos de dados.

Alguns modelos de dados como os orientados a objetos, por exemplo, suportam generalização e propriedade de herança enquanto que outros não suportam.

Se duas representações têm o mesmo conteúdo da informação, é fácil trabalhar com diferenças entre estruturas. Podemos citar como exemplo, um cadastro de cliente onde o endereço pode ser representado como uma entidade em um esquema e como um atributo composto em outro esquema.

Se o conteúdo da informação não for o mesmo pode ser muito difícil trabalhar com estas diferenças. Um exemplo aqui pode ser um cadastro de funcionários. Funcionário pode estar representado em um departamento pelo número do seu CPF enquanto que em outro pelo número da sua matrícula na empresa. Neste caso informações com conteúdos diferentes que estando em estruturas diferentes provocarão uma heterogeneidade de difícil solução.

### **2.2.3 Heterogeneidade Sintática**

Podemos dar o seguinte exemplo para ilustrar a heterogeneidade sintática. Consideremos uma empresa com bases relacionais distribuídas usando o mesmo Sistema Gerenciador de Banco de Dados (SGBD) e, que precisam ser integradas para responder a um requisito do negócio. No momento da integração podemos encontrar cada uma destas bases de dados com o mesmo cadastro de funcionários. Em cada uma delas este cadastro será uma entidade representada por uma tabela que poderá receber nomes diferentes que estejam mais de acordo com o departamento ao qual cada uma destas bases representa. Em uma das bases o cadastro de funcionários poderá estar representado por uma tabela de nome FUNC, em outra base de dados o mesmo cadastro de funcionários poderá estar representado por uma tabela de nome FUNCIONÁRIO, ou seja, conteúdos iguais, mas com sintaxes diferentes. Esta heterogeneidade é de fácil solução e pode ser resolvida com o uso de um dicionário de termos léxicos.

### **2.2.4 Heterogeneidade Semântica**

A heterogeneidade semântica considera o significado dos dados. Ela se refere a diferenças ou similaridades no significado dos dados de cada base. Acontece quando existe um desacordo entre o significado, a interpretação ou a intenção do uso dos mesmos dados.

Segundo (Hakimpour & Geppert, 2004), semântica, é a interpretação que pessoas atribuem aos dados, ou seja, relacionam o dado ao que ele representa de acordo com o seu entendimento de mundo. Diferentes interpretações de dados causam heterogeneidade semântica.

Podemos citar como exemplo dois bancos de dados BD1 e BD2 com dados de telefonia. No primeiro banco de dados temos o atributo CUSTO\_TARIFA proveniente da relação TRÁFEGO\_CURSADO no banco de dados BD1 que descreve o custo de uma ligação local (na mesma área de cobertura) sem cobrança de taxas. Consideremos um atributo com o mesmo nome CUSTO\_TARIFA da relação TRÁFEGO\_FATURADO no banco de dados BD2 que descreve o custo de uma ligação local (ainda na mesma área de cobertura) incluindo a cobrança de taxas. Vemos que os atributos têm a mesma propriedade sintática. Se compararmos os atributos BD1.CUSTO\_TARIFA e BD2.CUSTO\_TARIFA estaremos cometendo um equívoco uma vez que estes atributos são heterogêneos semanticamente. Neste caso a heterogeneidade acontece devido às diferenças no significado. Desta forma, durante o processo de integração, precisamos considerar que estes dois elementos se referem a conceitos diferentes. Se este tipo de problema não for detectado e resolvido, o uso de dados integrados pode levar a um resultado inválido.

Nos casos em que os usuários não sabem sobre a heterogeneidade semântica dos dados em que eles acessam os erros podem ser ainda mais graves.

Recentemente, a comunidade de banco de dados tem voltado a sua atenção para o uso de ontologias. A maior motivação, segundo (Necib et al., 2005), é que ontologias aparecem como solução potencial para resolver o problema complexo de heterogeneidade semântica de dados em muitos casos. A razão reside no fato de que uma ontologia pode prover um entendimento comum compartilhado de um domínio de aplicação de maneira consensual com o significado dos termos e seus relacionamentos para o domínio modelado, ou seja, para prover a interoperabilidade entre sistemas de informação de tal forma que os usuários não precisam se preocupar com a origem dos dados e a forma de armazenamento.

## 2.3 Ontologias

O termo ontologia tem a sua origem na filosofia, e é usado para representar uma visão de mundo em um sistema de categorias, ou seja, este sistema não depende de uma linguagem particular. No campo da Inteligência Artificial (IA) ontologias foram desenvolvidas, para facilitar o compartilhamento e o reuso do conhecimento. Desde os anos 90, ontologias têm se tornado um tópico de pesquisa popular, investigado por várias comunidades de pesquisa dentro da Inteligência Artificial, como engenharia do conhecimento, processamento de linguagem natural e representação do conhecimento. Em Inteligência Artificial ontologias fazem uso de um artefato da engenharia, constituído de vocabulário específico usado para descrever uma realidade, acrescida de um conjunto de suposições relacionadas ao resultado esperado que normalmente ficam na forma de lógica de primeira ordem, onde as expressões do vocabulário aparecem como predicados unários ou binários chamados respectivamente de conceitos e relações (Guarino, 1998).

Mais recentemente, a noção de ontologia está se expandindo em áreas como integração inteligente da informação, sistemas de informação cooperativos, recuperação da informação, comércio eletrônico e gestão do conhecimento. As ontologias tornaram-se muito populares devido à promessa de um entendimento comum e compartilhado de um domínio que pode ser comunicado entre pessoas e sistemas computacionais heterogêneos e distribuídos.

Segundo (Gruber, 1993), ontologias representam um conhecimento comum e compartilhado sobre um domínio específico, e têm como objetivo principal permitir a comunicação entre organizações, pessoas e aplicações. Mais especificamente, uma ontologia pode ser encarada como uma especificação explícita de uma conceitualização, que por sua vez pode ser entendida como a especificação dos conceitos, objetos e outras entidades existentes no domínio de interesse da ontologia, bem como a relação entre esses elementos.

A grande vantagem do uso de ontologias é a capacidade de estabelecer uma hierarquia de classes de objetos e seus relacionamentos. Computadores não são capazes de realmente entender uma informação, no entanto, são muito eficientes na manipulação de termos. A adição de regras de inferência torna as ontologias

ainda mais poderosas para serem utilizadas por computadores na manipulação de termos.

No caso de fontes de dados heterogêneas, o uso de ontologias pode tornar o processamento distribuído processável por computador através da descrição de entidades e relações entre elas, além de regras de integridade para o domínio. Neste contexto, uma ontologia pode ser usada para a definição de um sistema global, para servir como base fundamental no processo de integração dos dados.

Segundo (Hapimpour, 2004), existem duas aproximações que utilizam ontologias para resolver heterogeneidades semânticas em banco de dados. A primeira utiliza ontologias para tradução de consultas ou seus resultados. Esta aproximação é utilizada em sistemas de informação que sofrem a mudança freqüente de seus esquemas, quando muitas bases de dados são envolvidas ou ainda quando o número de bases envolvidas muda freqüentemente como em OBSERVER (Mena, 1996). A desvantagem desta aproximação está relacionada com o alto custo do processamento, pois à medida que consultas são realizadas a ontologia precisa ser processada a fim de derivar o mapeamento requerido.

A segunda aproximação usa ontologias para a geração de um esquema global. Esta aproximação se aplica em casos em que os esquemas não sofrem mudanças constantes. Nesta aproximação esquemas de banco de dados formam a ontologia de uma comunidade específica. Neste caso as bases de dados distintas convergem para uma ontologia por referência aos termos definidos nesta ontologia. Para se descobrir como os elementos dos diferentes esquemas das bases de dados estão relacionados podem ser utilizadas técnicas como relações de similaridades. Estas similaridades entre os termos podem ser encontradas por supervisores humanos ou podem ser descobertas por mecanismos automáticos apropriados como os algoritmos desenvolvidos para tal fim como em (Hapimpour, 2004).

No entanto, nenhuma destas aproximações será utilizada integralmente aqui neste trabalho, a fim de promover a heterogeneidade semântica. A solução para promover a heterogeneidade semântica partirá de ontologias representando bases de dados distintas que serão integradas em uma ontologia global numa referência ao esquema global gerado numa federação de dados resultante da integração de bases de dados distribuídas e heterogêneas.

## 2.4 Integração de Ontologias

Muitos trabalhos focam na heterogeneidade de estruturas e dados com a visão de banco de dados, estes trabalhos são frutos da vasta pesquisa em integração de bancos de dados heterogêneos. Novas aproximações de integração olham para a heterogeneidade ocorrida entre diferentes ontologias. Neste caso grande complicação é adicionada devido à necessidade de se ter que negociar com linguagens lógicas de diferentes níveis de expressividade no momento da integração. O reuso de ontologias requer certo esforço, ou seja, ontologias precisam ser combinadas de alguma forma para que o reuso seja possível. Isto pode ser feito pela *integração das ontologias*. Na integração de ontologias uma nova ontologia é criada a partir de duas ou mais ontologias existentes com partes se sobrepondo, sobreposição esta que pode ser virtual ou física gerando uma nova ontologia ou, as ontologias podem ser usadas em separado. Em ambos os casos as ontologias precisam ser alinhadas (trazidas para um acordo mútuo, tornando-as consistentes e coerentes).

Segundo (Mcguinnes et al, 2000), integrar ontologias consiste dos seguintes passos:

- 1) Encontrar os termos nas ontologias onde elas se sobrepõem;
- 2) Relacionar conceitos que são semanticamente fechados via relações de equivalência e classificação – alinhamento e,
- 3) Checar a consistência, coerência e não redundância de resultados. O alinhamento de conceitos entre ontologias é especialmente difícil porque ele requer compreensão do significado dos conceitos. Alinhar duas ontologias significa em mudanças em pelo menos uma delas, o que vai resultar em uma nova versão da ontologia que foi alterada.

Não é objetivo do nosso trabalho mostrar cada uma dessas etapas detalhadamente. A integração das ontologias será alcançada por meio de uma ferramenta desenvolvida para tal que será apresentada no capítulo 3 e que quando da sua concepção contempla cada uma destas etapas.

## 2.5 Diferenças entre Ontologias

Segundo (Klein et al., 2001), são dois os níveis em que podem acontecer diferenças na combinação entre ontologias. O primeiro é o nível da linguagem ou

nível do meta-modelo. Este é o nível das primitivas da linguagem que são usadas para especificar uma ontologia. Diferenças neste nível aparecem nos mecanismos usados para definir classes, relações e assim sucessivamente. O segundo nível é o nível da ontologia ou nível do modelo. Diferenças no nível da ontologia são as diferenças na forma como é modelado o domínio onde a ontologia está definida.

### 2.5.1 Diferenças no Nível da Linguagem

Diferenças no nível da linguagem ocorrem quando ontologias escritas em diferentes linguagens de ontologias são combinadas. Segundo (Klein, 2001), as diferenças neste nível podem ser classificadas entre:

- Diferenças de sintaxe da linguagem: Ocorre quando diferentes linguagens de ontologias usam sintaxes diferentes. Este tipo de diferença quase sempre vem acompanhado com outras diferenças no nível da linguagem. Um exemplo típico de uma diferença somente sintática é o de uma linguagem de ontologia que tem várias representações sintáticas. Neste caso um mecanismo para reescrever é suficiente para resolver esse problema.
- Diferenças nas representações lógicas: Esta diferença é um pouco mais complexa que a anterior. Podemos citar como exemplo o caso em que duas classes A e B, disjuntas, sejam representadas em uma linguagem por *disjoint A B* enquanto que em outra seja necessário usar negação das subclasses para representar esta disjunção, ou seja, *A subclass-of (Not B)* ou ainda, *B subclass-of (Not A)*. O problema aqui não é se alguma coisa pode ser expressa, pois os argumentos podem ser logicamente equivalentes, mas sim quais construtores da linguagem podem ser usados para expressá-la.
- Diferenças nas primitivas semânticas: A diferença mais sutil no nível do meta-modelo é a semântica dos construtores da linguagem. Apóia-se no fato em que o mesmo nome é usado como construtor em duas linguagens, mas com semântica diferentes, por exemplo, há várias representações para *A igual a B*. Notamos que duas ontologias podem usar a mesma sintaxe, mas a semântica pode divergir.

- Diferenças quanto à expressividade da linguagem: A diferença no nível do meta-modelo que mais causa impacto é a diferença em expressividade entre duas linguagens. Ela está ligada ao fato de que algumas linguagens podem expressar coisas que outras não podem. Por exemplo, algumas linguagens têm construtores para representar negação, enquanto que outras não têm.

### 2.5.2 Diferenças no nível da Ontologia

Diferenças ao nível da ontologia ou nível do modelo acontecem quando duas ou mais ontologias que descrevem parcialmente domínios sobrepostos são combinadas. Estas diferenças podem acontecer quando duas ontologias são descritas na mesma linguagem ou ainda em linguagens diferentes. Segundo (Klein, 2001), as principais diferenças neste nível são diferenças de conceitualização do domínio, diferenças de especificação, diferenças de terminologia e diferenças de codificação.

- Diferenças de conceitualização estão ligadas à maneira como o domínio é interpretado o que resulta em diferentes conceitos ontológicos ou diferentes relações entre estes conceitos. Podem estar associadas ao escopo do modelo como, por exemplo, duas classes podem representar o mesmo conceito sem possuir as mesmas instâncias, embora elas possam se interceptar. Diferenças deste tipo podem não ser resolvidas automaticamente, mas podem requerer a presença de um especialista de domínio a ser modelado.

Diferenças de conceitualização podem ainda estar relacionadas à cobertura e granularidade do domínio ao qual a ontologia está associada. (Chalupsk, 2000) cita como exemplo três ontologias sendo que a primeira precisa modelar carros, mas não caminhões, uma segunda que precisa modelar caminhões, mas, modelá-los em algumas poucas categorias e, uma terceira que precisa fazer uma distinção detalhada em tipos de caminhões tendo por base as suas estruturas físicas gerais como peso, propósito etc. Neste caso a diferença não é um problema: pelo contrário, é um motivo para

usarem-se diferentes ontologias juntas. O problema resultante é alinhar as partes sobrepostas da ontologia.

- Diferenças de explicação estão relacionadas com a forma como a conceitualização é especificada. Podem estar associadas ao estilo ou paradigma de modelagem (diferentes representações para tempo, ações, planos etc), ou ainda na descrição dos conceitos (convenções de modelagem). Várias escolhas podem ser feitas para a modelagem de conceitos em uma ontologia. Por exemplo, a distinção entre duas classes pode ser modelada usando um atributo qualificador ou introduzindo uma classe separada. Estas escolhas são influenciadas pelo sistema de inferência escolhido. Outra escolha na distinção de conceitos é a forma como a hierarquia *é-um* é escolhida na distinção de conceitos
- As diferenças de terminologias acontecem quando mesmos conceitos são representados por diferentes nomes e são ditas diferenças entre termos sinônimos e termos homônimos. (Klein, 2001) cita como exemplo de termo sinônimo o uso do termo carro em uma ontologia e o termo automóvel em outra. Uma solução para isto seria usar *Thesaurus*. *Thesaurus* são termos escolhidos a partir de uma estrutura conceitual previamente estabelecida, destinados à indexação e à recuperação de documentos e informações num determinado campo do saber. No *Thesaurus* cada termo corresponde a um conceito. Uma vez aceito, esse termo torna-se um “descriptor” ou um “indexador”. Caso o termo não seja aceito como “descriptor”, ele pode ser aceito como “remissivo”, isto é, remete para um termo autorizado (descriptor); e todos os termos estão relacionados entre si. Nenhum termo pode figurar no *Thesaurus* sem estar relacionado a algum outro, sendo essa relação determinada pelo seu significado. No caso de termos homônimos o significado de um termo pode mudar de um contexto para outro. (Klein, 2001), cita neste caso o termo condutor que em um domínio de música pode ter um significado bem diferente do significado do mesmo termo em um domínio de engenharia elétrica. Essa inconsistência é muito difícil

de ser encontrada e requer conhecimento humano para ser descoberta.

- Diferenças de codificação é um tipo trivial de diferenças. Valores em uma ontologia podem estar codificados em diferentes formatos. Um exemplo seria a codificação de data que em uma ontologia poderia ser *dd/mm/yyyy* e em outra ter o formato *mm-dd-yy* ou ainda a distância que pode estar representada em milhas em uma ontologia e quilômetros em outra.

## 2.6 Mapeamento, alinhamento e tradução.

Mapeamento, alinhamento e tradução de ontologias tem sido uma área de pesquisa ativa com o objetivo de conseguir a interoperabilidade e a integração semântica de informação.

### 2.6.1 Mapeamento/Alinhamento

**Mapeamento:** relaciona conceitos similares ou relações a partir de diferentes bases entre si por uma relação de equivalência.

**Alinhamento:** traz duas ou mais ontologias para um entendimento mútuo fazendo-as consistentes e coerentes

### 2.6.2 Merging/Integração

**Merging:** funde diferentes ontologias sobre o mesmo assunto em uma ontologia simples que unifica todas elas.

**Integração:** constrói uma nova ontologia reusando outras ontologias disponíveis (agrupa, estende, especializa).

## 2.7 Bancos de Dados Federado

Um Sistema de Bancos de Dados Federado, segundo (Sheth & Larson, 1990), é formado por uma coleção de sistemas de banco de dados autônomos e possivelmente heterogêneos que cooperam entre si para produzir respostas consolidadas para as consultas realizadas sobre a federação.

Os sistemas de bancos de dados componentes são integrados de várias formas. O software que provê a manipulação controlada e coordenada dos sistemas de bancos de dados componentes é chamado de Sistema Gerenciador de Bancos de Dados Federado (FDBMS).

Um dos aspectos importantes de uma Federação de Bancos de Dados é que um sistema de banco de dados componente pode realizar suas operações locais ao mesmo tempo em que participa da Federação. A integração dos sistemas de bancos de dados componentes pode ser mantida pelo usuário da Federação ou pelo administrador da Federação junto com os administradores dos sistemas de banco de dados componentes. A integração depende da necessidade dos usuários da Federação e da permissão dos administradores dos sistemas de banco de dados componentes que participam da Federação em permitir o compartilhamento dos seus bancos de dados.

As Federações de Bancos de Dados podem ser classificadas de acordo com quem as gerencia e como os dados estão integrados, podendo ser classificadas em fracamente acopladas ou fortemente acopladas. Uma federação é fracamente acoplada se a responsabilidade em manter e gerenciar os dados da federação é do usuário da federação e não existe controle centralizado por parte do sistema federado e de seu administrador. Federação fortemente acoplada é aquela cujo sistema federado ou seu administrador controla o acesso aos bancos de dados componentes.

### 2.7.1 Arquitetura de cinco níveis

(Sheth & Larson, 1990) definem uma arquitetura de cinco camadas para um Sistema Federado como mostrado na . Esta arquitetura é composta dos seguintes itens (Figura 1):

- **Fontes de Dados:** representa as diversas fontes de dados que compartilham informações possivelmente heterogêneas.
- **Esquema Local:** representa o esquema conceitual de um Sistema de Banco de Dados Componente. Um esquema local é expresso no modelo de dados nativo do Sistema de Banco de Dados Componente. Desta forma, diferentes esquemas locais podem ser expressos em diferentes modelos de dados.

- **Esquema Componente:** é o esquema gerado a partir da tradução do esquema local para o modelo de dados canônico adotado pela federação.
- **Esquema de Exportação:** nem todo dado de um Sistema de Banco de Dados Componente pode estar disponível para todos os usuários da Federação. Assim, o esquema de exportação representa o subconjunto do esquema Componente que será compartilhado na federação. Este esquema possui ainda informações para controle de acesso com relação ao uso das informações disponíveis na federação por usuários específicos.
- **Esquema Federado:** é a integração dos múltiplos esquemas de exportação, também chamado de Esquema Conceitual Global.
- **Esquema Externo:** define a visão que um usuário ou uma aplicação terão dos dados da federação. O uso de esquemas externos pode viabilizar a personalização das informações disponibilizadas aos usuários. Como o esquema global de uma federação pode ser grande, complexo e de difícil atualização, o uso de um esquema externo pode ser personalizado. Um esquema externo pode conter somente um subconjunto das informações da federação contendo apenas as informações relevantes para os usuários da federação, podendo ainda ser representado em um modelo de dados mais adequado para o usuário da federação.

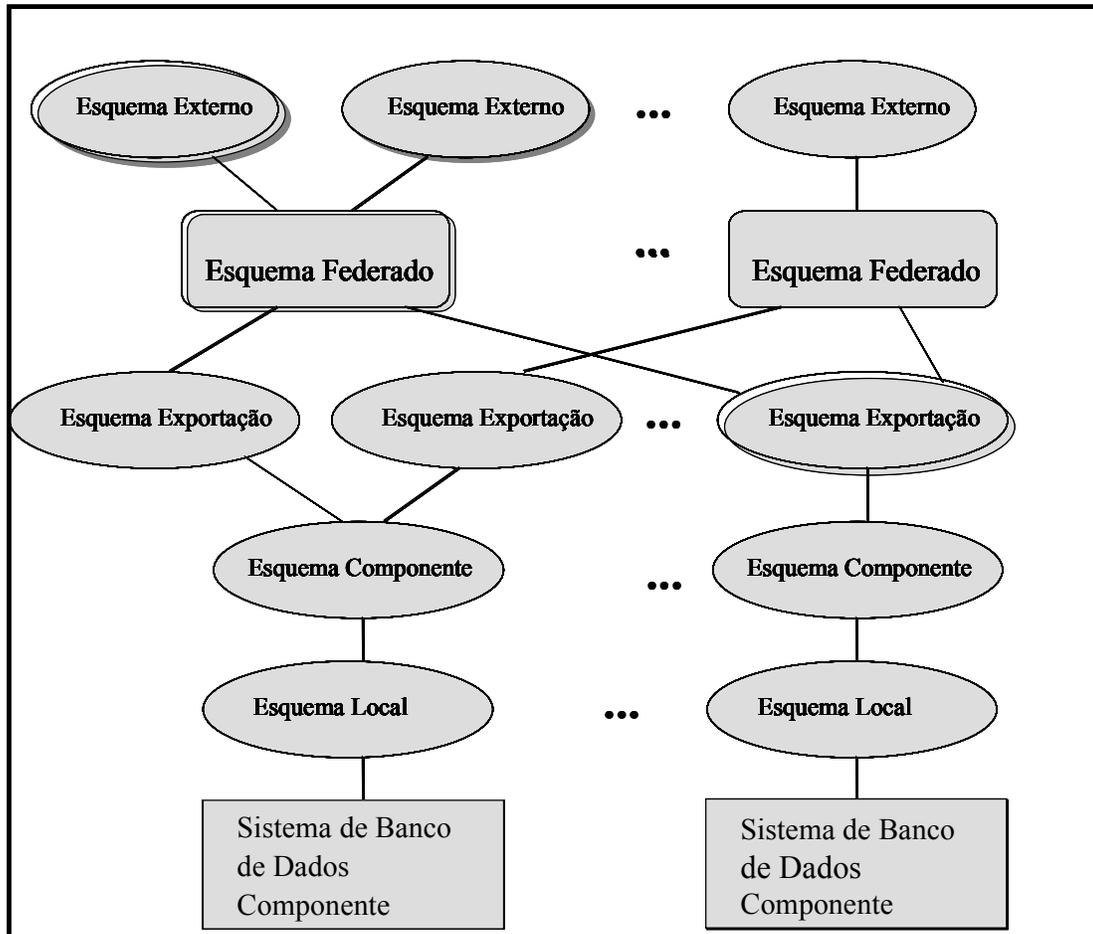


Figura 1 : Arquitetura de 5 níveis para uma federação de bancos de dados.

## 2.8 Abordagem do trabalho

Após apresentarmos a arquitetura de cinco camadas definida por (Sheth & Larson, 1990) para uma Federação de Bancos de Dados definiremos uma arquitetura similar para uma federação de ontologias. A idéia básica é partir do conhecimento da tecnologia de bancos de dados para a integração de fontes de dados componentes que já tem seus esquemas locais heterogêneos, que são transformados em esquemas de exportação homogêneos representando um modelo de dados comum. Estes esquemas de exportação são integrados em um modelo global representando o esquema da federação de bancos de dados. De maneira análoga, vamos realizar a integração de fontes de dados heterogêneas que tem ontologias descrevendo-as, numa situação equivalente aos esquemas locais que descrevem as fontes de dados. Vamos realizar a integração destas ontologias usando uma ferramenta desenvolvida para tal e que nos permitirá gerar uma

ontologia global. Adotaremos então a aproximação de federação de bancos de dados e vamos construir de maneira similar a federação de ontologias.