

**Universidade Federal de Pernambuco**

**Centro de Informática**

**Pós-Graduação em Ciência da Computação**

## **Trabalho Individual 2**

**Roteamento de Consultas em PDMS**

Aluna: Crishane Azevedo Freire

Professora: Ana Carolina Salgado

Recife, Janeiro de 2011

## Conteúdo

1. Introdução .....	3
2. Estratégias de Busca e Roteamento em Sistemas P2P .....	4
2.1. Inundação .....	5
2.2. Controle Central .....	6
2.3. Brokering (agentes inteligentes).....	7
2.4. super-peer/peer .....	8
2.5. Tabela Hash Distribuída e Árvore de busca distribuída.....	8
2.6. Semantic Overlay Networks (SON).....	9
3. Trabalhos relacionados ao roteamento de consultas em PDMS.....	10
3.1. Árvore de Decisão e Hipergrafos Transversais [Ismail et al. 2010].....	10
3.2. Ontozilla[Joung, Chuang 2009] .....	12
3.3. SRI - Semantic Routing Index [Mandreolli et al. 2007b] .....	14
3.4. H-Link [Montanelli 2007].....	16
3.5. OntoSum [Li,Vuong 2007] .....	18
3.6. SEWASIE [Beneventano et al. 2007] .....	20
4. Quadro resumo dos artigos apresentados .....	22
5. Considerações .....	25
6. Referências.....	26

## 1. Introdução

Questões relativas à cooperação, integração e coordenação entre *peers* de informação em um ambiente de rede têm sido pesquisadas sobre diferentes contextos, incluindo Integração de Dados [Halevy 2000;Lenzerini 2002], Web Semântica [Heflin, Hendler 2001], Gerenciamento de Dados em arquiteturas *peer-to-peer* (PDMS) e *Grid* [Bernstein et al. 2001;Halevy et al. 2003] , serviços orientados a sistemas com agentes e computação distribuída. De uma forma geral, todos estes sistemas são caracterizados por uma arquitetura constituída por vários *peers* autônomos (chamados sítios, fontes, agentes ou pontos) que guardam informações e são interligados com outros *peers* através de mapeamentos. Dois problemas básicos são identificados nesta arquitetura: como descobrir, expressar e compor o mapeamento entre os *peers*, e como explorar estes mapeamentos com o objetivo de responder às consultas postas em cada *peer* de forma eficiente [Calvanese et al. 2004].

Requisitos para sistemas de informação amplamente distribuídos que apóiem as organizações virtuais têm dado origem a uma nova categoria de sistemas *peer-to-peer* (P2P) baseados em esquemas. Em tais sistemas cada ponto expõe seu próprio esquema e o principal objetivo é a busca eficiente através da rede P2P realizando o processamento de cada consulta de entrada sem o consumo excessivo de largura de banda da rede. A utilização destes sistemas depende de técnicas eficazes para encontrar e recuperar dados, porém, o roteamento de consultas com base no conteúdo de maneira eficiente é um problema desafiador em redes P2P [Ismail et al. 2010].

Sendo assim, um dos grandes desafios em *peer Data Management Systems* (PDMS) está relacionado ao processamento da consulta, em especial à técnica de roteamento utilizada. Considerando a não existência de um ponto de centralização de todo conhecimento distribuído e disponível nesses sistemas, torna-se importante observar as estratégias adotadas para o roteamento eficiente dessas consultas. Cada *peer* precisa decidir, independente dos demais *peers* que compõem a rede, para qual dos seus vizinhos a consulta deverá ser encaminhada. Essa decisão normalmente é feita tomando como base apenas o conhecimento local disponível no *peer*.

Nesse trabalho é feito um levantamento do estado da arte de soluções de roteamento utilizadas em PDMS. Sendo assim, faremos a apresentação de algumas tecnologias de busca e roteamento utilizadas em sistemas P2P, abordaremos alguns problemas relativos ao roteamento de consultas nesses ambientes e concluiremos apresentando alguns trabalhos que vêm sendo desenvolvidos com relação a estratégias de busca e roteamento aplicadas diretamente a PDMS.

## 2. Estratégias de Busca e Roteamento em Sistemas P2P

Os sistemas P2P adotam uma abordagem completamente descentralizada quanto ao gerenciamento de recursos em uma rede. Distribuindo o armazenamento e o processamento através de *peers* autônomos eles podem escalar sem a necessidade de poderosos servidores. Em se tratando da busca de informações, toda aplicação deve ter uma maneira de selecionar os *peers* relevantes que devem responder a uma determinada consulta submetida em um *peer* qualquer, mesmo que essa maneira não seja a melhor solução. Essa é uma forma de evitar que as consultas postas nesses ambientes acabem por congestionar toda a rede.

Em redes P2P não estruturadas, os dados estão distribuídos aleatoriamente, cada *peer* conhece seu *peer* vizinho, mas não sabe qual o recurso que ele dispõe. O roteamento de consulta é feito tipicamente a partir de mecanismos de inundação utilizados para distribuir a consulta aos demais *peers*. Sendo assim, um dos principais problemas em redes não estruturadas está relacionado ao seu crescimento e incompletude dos resultados alcançados pelas consultas em virtude do mecanismo de roteamento adotado. Alguns exemplos de sistemas P2P que utilizam redes não estruturadas incluem o Gnutella, KaZaA e FreeHaven [Akbarinia et al. 2007].

De forma contrária, as redes estruturadas surgiram como uma proposta de resolver os problemas de crescimento das redes não estruturadas. A localização dos dados assim como os seus mapeamentos é representada na forma de tabelas de roteamento distribuídas. Sendo assim, durante o processamento de uma consulta submetida em um determinado *peer*, um índice é utilizado para fazer o roteamento dessa consulta para os *peers* relevantes. Essa estrutura pode envolver vários graus de coordenação central ou conhecimento global dispostos, por exemplo, em *super-peers*. Exemplos de sistemas em redes P2P estruturadas incluem, Chord [Stoica et al. 2001], CAN [Ratnasamy et al. 2001], Tapestry [Zhao et al. 2004], Pastry [Rowstron, Druschel 2001], Freenet [Clarke et al. 2002] e P-Grid [Aberer et al. 2003] [Akbarinia et al. 2007].

Ao contrário das consultas em bancos de dados tradicionais, a maioria das consultas em sistemas P2P não é exaustiva. Por exemplo, quando um usuário inicia uma busca por uma música, ele não está interessado em cada instância daquela música. Semelhante a uma pesquisa na web, a maioria dos usuários fica satisfeita com um pequeno subconjunto de todas as correspondências encontradas [Crespo, Garcia-Molina 2003]. Muitos sistemas fazem uso de abordagens que combinam diversas técnicas e estratégias desenvolvidas a partir de diferentes áreas de pesquisa [Haase et al. 2008]. A seguir são apresentadas algumas delas:

## 2.1. Inundação

Normalmente utilizada em sistemas de compartilhamento de arquivos. As mensagens são enviadas a todos os *peers*, ou seja, um *peer* encaminha a consulta para seus vizinhos, que por sua vez encaminha para seus vizinhos e assim sucessivamente. Para evitar que as mensagens fiquem navegando indefinidamente é definido um tempo de vida – *time to live* (TTL) – medido em saltos, ou seja, em número de vezes que a consulta foi encaminhada [Costa 2009]. O objetivo é que *peers* possam encaminhar sua consulta aos demais *peers* de forma que um número suficiente de repostas seja alcançado ou até que um determinado número de encaminhamentos (saltos) seja realizado. Essa abordagem tem como aspecto negativo o fato de que cada consulta pode retornar uma grande quantidade de dados o que conseqüentemente ocasionaria uma sobrecarga na rede. Outro aspecto está relacionado à incompletude das respostas mesmo que a informação esteja disponível na rede. Ou seja, durante o processo de roteamento, é possível que a consulta atinja o número máximo de saltos (encaminhamentos) estimados antes mesmo de obter o dado desejado.

Com o objetivo de melhorar o desempenho da técnica de inundação, algumas pesquisas têm indicado o uso do algoritmo de colônia de formigas [Ciglaric, Vidmar 2006; Michlmayr 2006] na otimização dos caminhos que devem receber a consulta. Colônia de formigas, também conhecido como ACO – *Ant Colony Optimization System*, é um algoritmo de otimização baseado no comportamento das formigas e suas colônias na natureza. Sua utilização permite identificar quais caminhos têm maior probabilidade de retornar bons resultados. Outra característica importante deste algoritmo é a capacidade de absorver mudanças no grafo dinamicamente. Tal característica habilita esta técnica para o uso em sistema de roteamento em redes de computadores dinâmicas, como as redes *peer-to-peer* [Costa 2009].

O Sistema HyperCup (*P2P Hypercube*) [Schlosser et al. 2002] utiliza uma rede semântica global para organizar os *peers* em uma topologia de grafo hipercubo. Dessa forma, garante que qualquer mensagem enviada via inundação atingirá todos os *peers*. As informações disponíveis nos *peers* podem ser categorizadas em um conceito global. Conceitos podem ser organizados em uma ontologia global que define os relacionamentos entre conceitos existentes. Nesse caso, cada conceito será definido como um hipercubo (Figura 1)

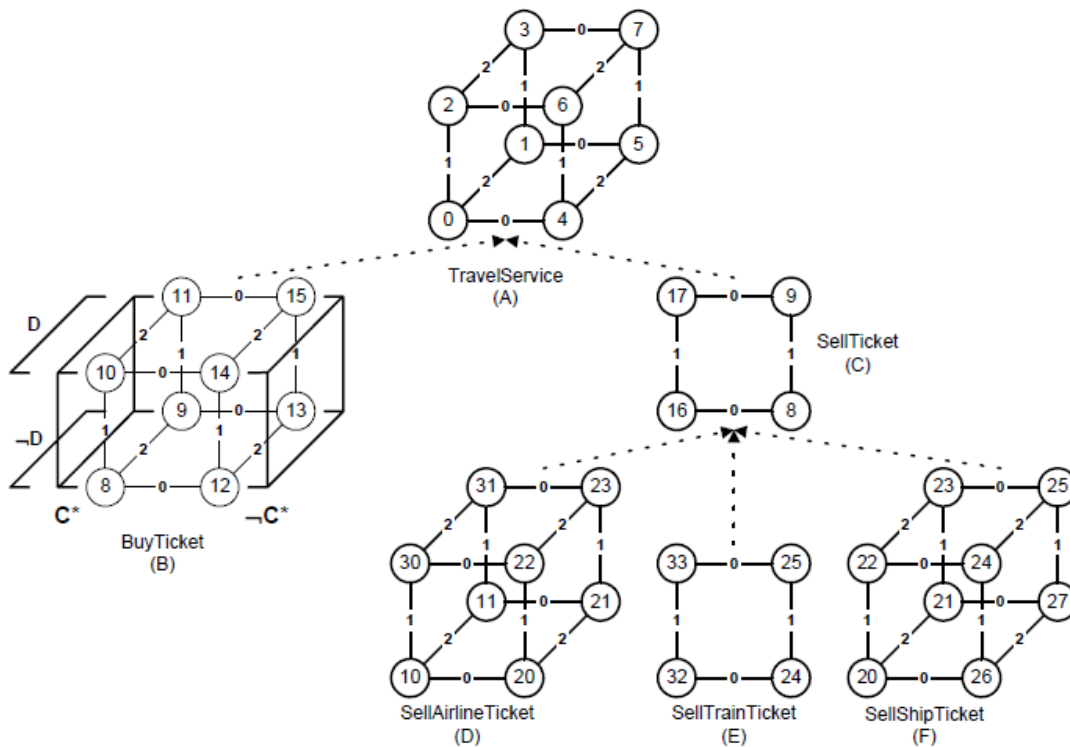


Figura 1: Topologia da rede estruturada por conceito (ontologia) [Schlosser et al. 2002]

## 2.2. Controle Central

O sistema mantém a centralização de todas as informações disponíveis nos *peers* que compõem a rede. Quando recebe uma solicitação, o diretório central identifica, a partir de um índice, o *peer* mais adequado a fornecer a resposta e a comunicação entre eles então é estabelecida. Em pequenas organizações, essa abordagem pode trabalhar muito bem porque a rede por ser pequena e estável apresenta pouco processamento de consulta e atualizações. Em grandes redes, problemas de escalabilidade devem surgir uma vez que o número de solicitações aumenta em virtude do aumento de usuários e *peers*. Conseqüentemente, o espaço para armazenamento e o gerenciamento central ficam comprometidos. Esse modelo foi popularizado pelo NAPSTER (2010) [Kamienski et al. 2005]. A Figura 2 ilustra a forma de resposta as requisições estabelecidas nesse modelo

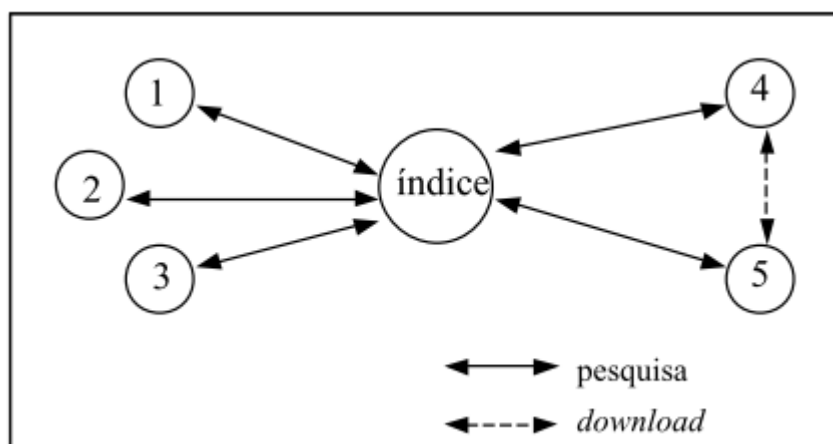


Figura 2: Modelo Centralizado [Kamienski et al. 2005]

### 2.3. Brokering (agentes inteligentes)

A comunidade de sistemas multi-agentes sugere o conceito de “*broker agents*” onde tecnologias relacionadas a agentes e ontologias são utilizadas como forma de desenvolver mecanismos de busca avançados para acesso inteligente a fontes de dados espalhadas na rede. O projeto InfoSleuth [Bayardo 1997] através do uso dos agentes *brokering* utiliza a representação da ontologia exportada pelo agente de recurso em LDL (*Logical Data Language*). O mecanismo dedutivo de LDL ajuda a determinar a consistência da restrição na consulta do usuário e daquela exportada pelo agente de recurso o qual determina a relevância da informação. Dessa forma, é possível fazer o roteamento de informações de forma inteligente baseada no conteúdo. Na Figura 3 são apresentados os componentes de um agente de recurso. O módulo de comunicação interage com os outros agentes. O processador de linguagem traduz a consulta expressa em termos da ontologia global em uma consulta expressa em termos do esquema do banco de dados. O resultado da consulta também é traduzido em uma forma que os outros agentes possam compreender. O mapeamento de informações necessárias para esse processo é criado durante a instalação do agente que necessita do conhecimento especializado do dado local e da ontologia global. A função do módulo de detecção de eventos é monitorar as fontes de dados para os eventos que tenham interesse e preparar as notificações a serem enviadas aos agentes interessados naqueles eventos.

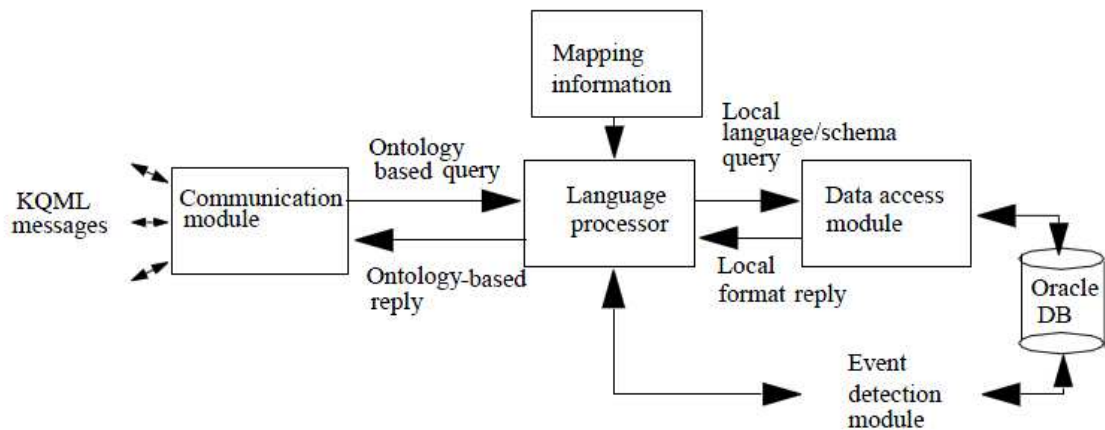


Figura 3: Componentes do agente de recurso [Bayardo 1997]

#### 2.4. *super-peer/peer*

Desenvolvida pela comunidade de pesquisa *peer-to-peer*, essa técnica oferece bons resultados para compartilhamento de arquivos. Como sistemas cliente-servidor, alguns *peers*, ou *super-peers*, agem como servidores dedicados a alguns outros *peers* e pode executar funções complexas tais como indexação, processamento de consulta, controle de acesso e gerenciamento de metadados. Um dos exemplos mais conhecidos desse modelo é o KaZaa cujos *peers* agem voluntariamente como *super-peers* que mantêm grandes tabelas de roteamento onde estão armazenadas informações sobre o conteúdo de outros *peers*. Essa abordagem não deixa de ser uma forma de centralização do sistema embora seja melhor que a solução de inundação. O trabalho de Nejdí e seu grupo [Nejdí et al. 2005], mostra como esta abordagem baseada em esquema pode ser usada para criar SOC (*Semantic Overlay Clusters*) em uma rede científica *peer-to-peer* com um pequeno conjunto de atributos de metadados que descrevem os documentos dispostos na rede. Para evitar inundação das consultas, foram introduzidos índices de roteamento entre os *super-peers* que permitem identificar os *peers* que contêm respostas relevantes a partir do armazenamento dos metadados usados em cada *peer*.

#### 2.5. Tabela Hash Distribuída e Árvore de busca distribuída

As *Distributed Hash Tables* (DHT) são baseadas na idéia de roteamento por conteúdo para aqueles pontos cujo identificador mais se aproxima do identificador do conteúdo desejado. Esta técnica parte do princípio de que todos os pontos têm a mesma função *hash* para associar identificadores únicos para qualquer tipo de conteúdo como, por exemplo, documentos, músicas, URL ou palavras. Alguns algoritmos como Chord (Stoica, 2001), Pastry (Druschel, 2001) e CAN (Ratnasamy, 2001) implementam esse modelo. Uma desvantagem desse modelo é o alto custo, devido às freqüentes mudanças na rede em função da dinâmica relacionada à entrada e saída de pontos. Em se tratando de árvores de busca distribuída o *P-Grid* é uma árvore de busca binária virtual que utiliza cópias dos pontos na rede e usa algoritmos randômicos para acesso e busca. Chaves de busca são representadas de forma binária e distribuídas através dos pontos.



Cada ponto armazena parte da árvore total e sua posição é representada na forma de uma seqüência de bits binários. Essa seqüência representa o subconjunto das informações contidas na árvore total pelo qual o ponto é responsável. Logo, para cada bit em sua seqüência, um ponto armazena o endereço de no mínimo outro ponto para garantir que embora algum ponto esteja fora da rede outros pontos poderão ser responsáveis pelo mesmo caminho.

## 2.6. *Semantic Overlay Networks (SON)*

Nessa estratégia, é definida uma rede virtual pelo agrupamento de nós com ligações semânticas. Durante a conexão dos pontos, o conteúdo é classificado de acordo com a semântica associada, podendo inclusive, ocorrer sobreposição de redes. Os pontos com conteúdo semanticamente equivalentes são agrupados em *clusters* (cada *cluster* forma uma SON) [Crespo, Garcia-Molina 2003]. A Figura 4 exemplifica esse tipo de rede onde uma SON é definida considerando a formação de *clusters* de acordo com o estilo musical que cada ponto possui.

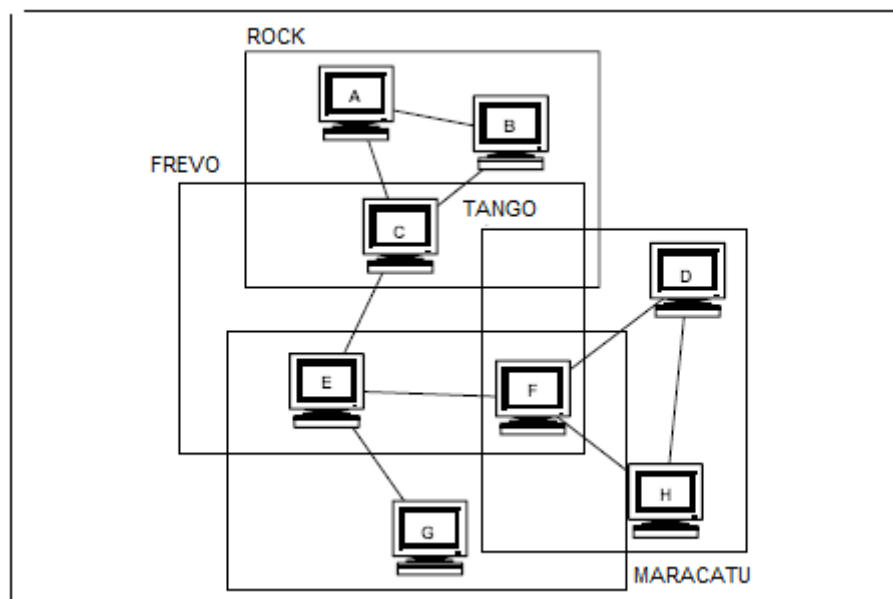


Figura 4: *Semant Overlay Networ*, adaptado de [Crespo 2003]

Quando um ponto possui um estilo musical que se encaixa também em outros estilos ocorre uma sobreposição das redes. As consultas executadas em um sistema que utiliza SON são enviadas apenas aos agrupamentos semânticos relacionados ao tema da consulta, ignorando os pontos que estejam fora do tema [COSTA, 2009].

### 3. Trabalhos relacionados ao roteamento de consultas em PDMS

Muitos foram os benefícios obtidos com o crescimento dos sistemas *peer-to-peer* na computação distribuída. Entretanto, a necessidade cada vez maior de incorporar valor semântico aos dados fez com que os *peer Data Management Systems* (PDMS) fossem introduzidos como uma solução para o problema de compartilhamento em larga escala de dados com maior riqueza semântica [Halevy et al. 2004].

Os PDMS foram introduzidos como uma extensão natural dos bancos de dados distribuídos em um contexto de sistemas P2P [Halevy et al. 2003]. Em um PDMS, cada *peer* mantém um banco de dados (documento XML, etc) associado a um esquema de dados que representa seu domínio de interesse. Os relacionamentos semânticos entre *peers* são providos localmente entre pares (ou um pequeno conjunto) de esquemas dos *peers*. Para consultar um *peer*, seu próprio esquema é usado para formular a consulta, e as respostas à consulta podem vir de qualquer outro ponto na rede que esteja conectado por meio de um caminho estabelecido por um mapeamento semântico entre os pontos [Tatarinov, Halevy 2004].

Para consultar um *peer* em um PDMS, seu próprio esquema é utilizado para formular a consulta e seus mapeamentos semânticos são também utilizados para reformular a consulta aos seus vizinhos sempre que se fizer necessário. Sendo assim, um dos grandes desafios nesses sistemas está relacionado ao roteamento da consulta, isto é, torna-se necessário o desenvolvimento de mecanismos que permitam o roteamento da consulta apenas aos *peers* mais relevantes ao contexto da consulta. A seguir apresentamos alguns trabalhos que buscam tornar eficiente a forma de recuperação de informações em um PDMS a partir da seleção do melhor caminho semântico durante consultas.

Um aspecto positivo dos PDMS está em aliar todos os benefícios advindos dos sistemas P2P às funcionalidades e recursos existentes em um SGBD (Sistema de Gerenciamento de Banco de Dados) a exemplo da transparência em operações de consultas e gerenciamento dos dados. A seguir serão apresentados alguns trabalhos relacionados ao roteamento de consultas tanto em PDMS quanto em sistemas que tratam com o gerenciamento de dados estruturados e semi-estruturados em redes P2P.

#### 3.1. Árvore de Decisão e Hipergrafos Transversais [Ismail et al. 2010]

Ismail e seu grupo [ISMAIL et al. 2010], propõem uma topologia *super-peer* para a criação de comunidades semânticas e utilizam um método baseado em árvore de decisão para selecionar os *peers* relevantes em uma dada consulta; algoritmos baseados em hipergrafos transversais também são utilizados para o roteamento da consulta. Segundo os autores, a vantagem desse modelo está relacionada à robustez do roteamento da consulta e às questões relacionadas ao crescimento da rede, privacidade dos dados e natureza dinâmica da rede *overlay* (*peers* podem sair e outros podem juntar-se à rede a qualquer tempo). O sistema proposto é um sistema híbrido P2P baseado na

organização de *peers* em torno de *super-peers* conforme similaridade de conteúdo. Cada *super-peer* (SP) também está conectado ao *Knowledge-super-peer* (KSP), um mecanismo que especifica os *super-peers* que tenham *peers* com dados relevantes que possam ser respondidos com o mínimo de tarefas de consulta e, por consequência, fornecem um melhor tempo de resposta. Os *peers* KSP têm um melhor poder computacional e maior largura de banda. Eles são responsáveis por encaminhar as consultas para *super-peers* relevantes, permitindo não somente reduzir o esforço de compilação das consultas mas também prevenir que as consultas sejam espalhadas por toda a rede. Cada *peer* KSP é representado por uma árvore de decisão, construída a partir das consultas processadas pelo próprio *peer*. O agrupamento de *super-peers* a partir do seu domínio semântico leva à construção explícita de comunidades onde cada uma é representada por um conjunto de *super-peers* com a restrição de que cada *super-peer* pode pertencer a mais de um agrupamento. Nesse caso, o conjunto de agrupamentos constitui um conjunto de hipergrafos, onde cada nó representa uma comunidade. O algoritmo utilizado produz um agrupamento entre *super-peers* que possuem histórico de sucesso em resposta a alguns itens correspondentes aos componentes de uma consulta. A Figura 5 exemplifica os conjuntos de rotas em um hipergrafo de *super-peers* { {SP1, SP2, SP6}, {SP1, SP6, SP8} e {SP3, SP7, SP8, SP10}}. Cada agrupamento tem no mínimo um *super-peer* em comum usado para encontrar a transversalidade mínima entre os agrupamentos. Os *super-peers* em comum são usados para encaminhar as consultas para outros agrupamentos.

Na Figura 5, assumimos que uma consulta Q1 é submetida em P1, e que o processo de roteamento da consulta se dará da seguinte forma:

1. O *super-peer* responsável por P1 é identificado, no caso, SP1
2. SP1 envia a consulta para cada *super-peer* de sua rota transversal {SP1, SP2, SP6}
3. Cada SP que recebe a consulta encaminha para seus *peers* aptos a responder a consulta
4. No final o retorno da consulta irá corresponder ao conjunto de *peers* relevantes e seus respectivos SP ((P2:SP1),(P11:SP8)...).

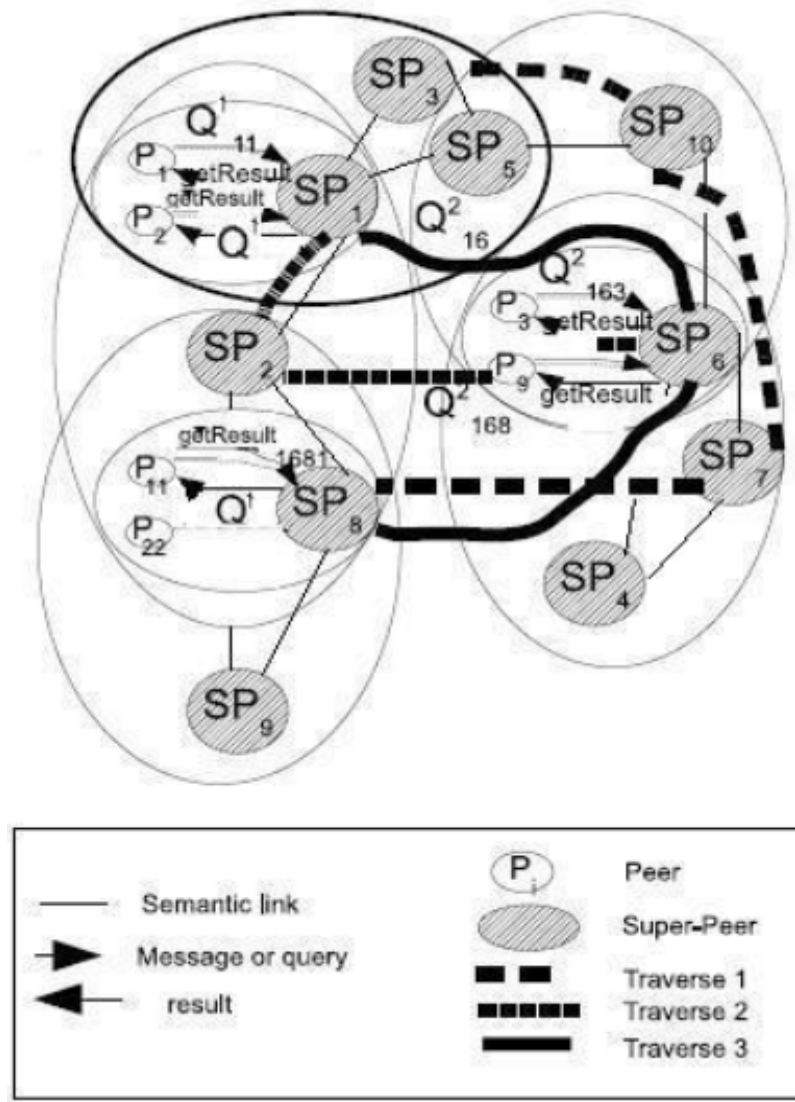


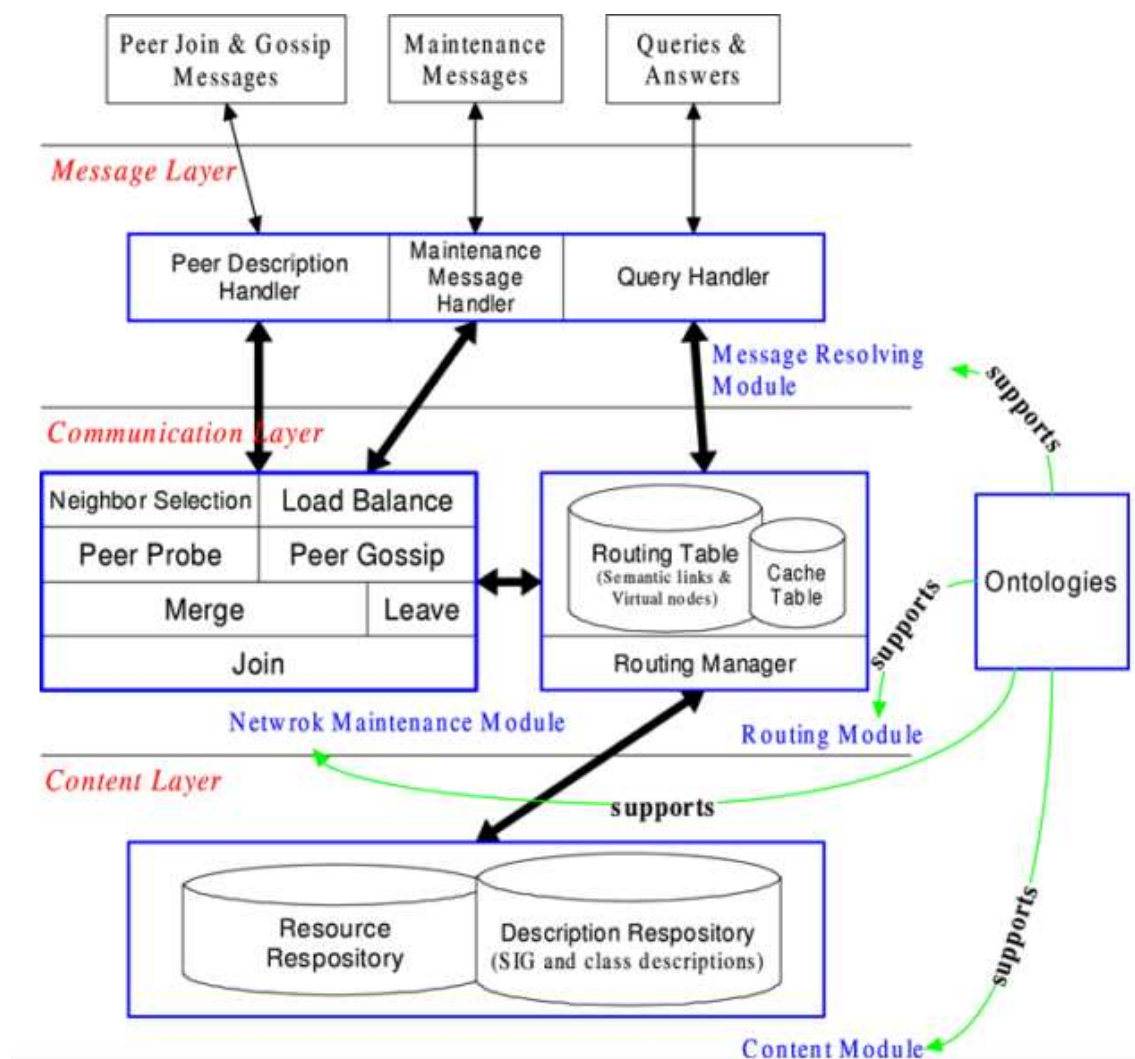
Figura 5: Roteamento da consulta (abordagem KSP) e configuração da rede [Ismail et al. 2010]

### 3.2. Ontozilla[Joung, Chuang 2009]

O Ontozilla utiliza ontologias em sistemas P2P com o objetivo de melhorar o desempenho do processo de busca de informações e facilitar o processo de integração e interoperabilidade. Os *peers* que compõem a rede com mesmo domínio de conhecimento são agrupados em *clusters*, e os relacionamentos entre eles são modelados de acordo com o conhecimento que eles representam. Conforme a nomenclatura adotada pelo Ontozilla, *peers* com interesses idênticos ou similares são agrupados em SIG (*Special Interest Groups*). Em cada SIG, os *peers* empregam algum sistema de classificação para classificar seus interesses em classes, onde cada classe representa um *cluster* de *peers* que suporta o mesmo conceito. Cada *link* que representa uma ligação entre um *peer* pai e outro filho é representado por relacionamento de subclasse “*is subclass of*” (*is-a*). SIG são descrições conhecidas por todos os *peers* na rede. Cada classe tem uma descrição contendo o nome da classe, o nome SIG, a

hierarquia de classificação e algumas anotações da classe. As descrições de SIG e classes são representadas com linguagens de ontologia. Logo, os relacionamentos entre SIG e/ou classes podem ser inferidos com uma ontologia pré-definida, ou usando algumas técnicas de correspondência semântica onde se possa medir a similaridade entre os SIG ou classes de forma a determinar se dois *peers* abrigam o mesmo assunto.

O roteamento no Ontozilla é feito conforme o esquema de roteamento intra-SIG e inter-SIG. O roteamento intra-SIG é feito de forma direta. Se um *peer* recebe uma consulta que solicita um recurso pertencente a sua super-classe, ele encaminha a consulta para o *cluster* mais superior. Se a consulta é para recursos pertencentes a sua subclasse, ele encaminha a consulta para o *cluster* mais abaixo. Finalmente, se a consulta é para os recursos pertencentes a sua própria classe, ele transmite a consulta ao seu *cluster* através de ligações individuais. Se ainda, o *peer* que fez a consulta não estiver satisfeito com o número de respostas, ele pode fazer uma nova consulta agora envolvendo as subclasses da classe destino. O roteamento inter-SIG está relacionado ao pedido de recurso fora do SIG do próprio *peer*. Quando um *peer* necessita um recurso ele procura em sua tabela de roteamento e encaminha a consulta para o *peer* correspondente via ligações SIG. Se a tabela de roteamento não contém qualquer *peer* correspondente, ele transmite um *SIG inquire messages*, que são mensagens utilizadas no processo de junção de SIG, dentro de um número de saltos estimado, para que seja feita uma busca do novo SIG. Caso o SIG seja encontrado, a consulta é enviada para ele e, então, é roteada conforme o processo intra-SIG. Na Figura 6, é possível visualizar na arquitetura do Ontozilla os módulos e tabelas responsáveis pelo processo de roteamento da consulta.



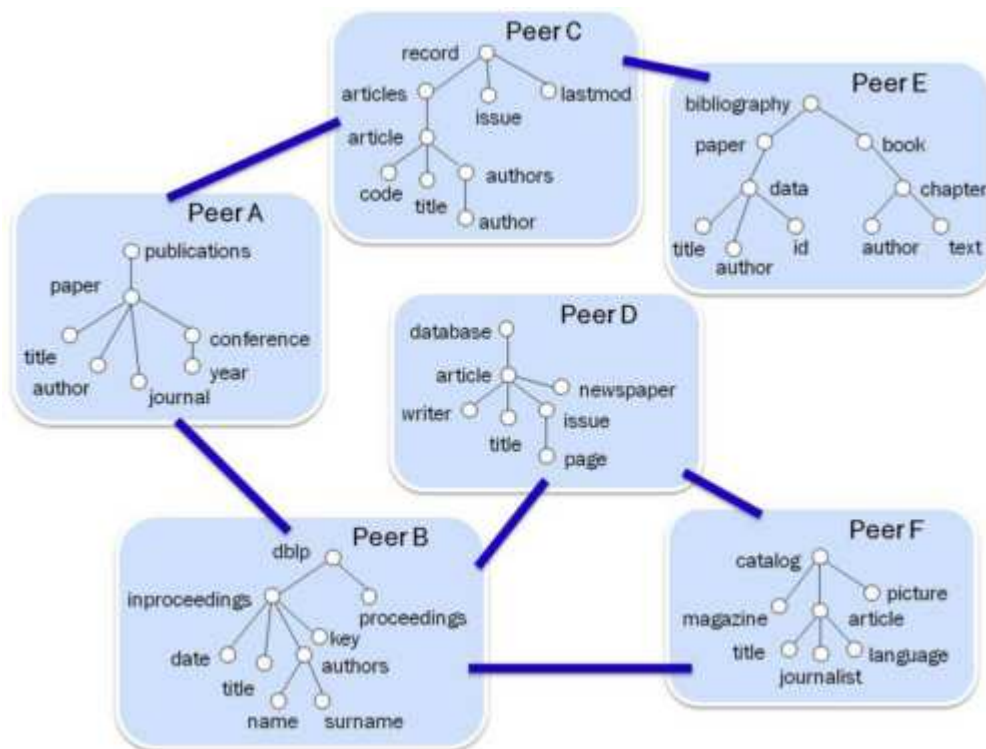
**Figura 6: Arquitetura do Ontozilla**

A eficiência e correteza do roteamento da consulta estão baseadas na confiança das ligações semânticas entre os pontos. Entretanto, os *peers* registrados na tabela de roteamento podem estar indisponíveis, considerando que os *peers* podem entrar ou sair da rede de forma imprevisível. Além disso, devido às falhas e ao atraso na propagação das mudanças, as visões dos *peers* na rede podem estar inconsistentes. É importante então destacar a necessidade de mecanismos que melhorem essa instabilidade ocasionada na rede.

### 3.3. SRI - Semantic Routing Index [Mandreolli et al. 2007b]

No trabalho de Mandreolli e seu grupo (Mandreolli et al. 2007b) o PDMS SUNRISE usa índices de roteamento semânticos (SRI) [Mandreolli et al. 2007a, Mandreolli et al. 2006] como forma de tornar eficiente o processamento da consulta em seu sistema. Cada *peer* que compõe a rede mantém um resumo das informações a respeito do grau de similaridade semântica entre os conceitos armazenados entre um

*peer* e seus respectivos vizinhos. Essa informação é mantida em uma estrutura de dado local denominada SRI. Sendo assim, um *peer*  $p$  que tenha  $n$  vizinhos e  $m$  conceitos em seu esquema, armazena um SRI estruturado como uma matriz com  $m$  colunas e  $n+1$  linhas, onde a primeira linha refere-se ao conhecimento sobre o esquema local do *peer*  $p$ , como mostra a Figura 7. É possível assim, que cada *peer* sintetize, para cada conceito de seu esquema, a aproximação semântica das sub-redes acessíveis a partir de seus vizinhos e, assim, fornecer uma informação sobre a relevância dos dados que podem ser alcançados em cada trajeto a ser escolhido.



(a) Cenário de demonstração de um PDMS

PeerA SRI	paper	title	author	...
PeerB	0.51	0.49	0.37	...
PeerC	0.81	0.86	0.66	...

(b) Parte do Índice de Roteamento Semântico (SRI) do *peer* A

Figura 7: Exemplo de referencia do trabalho [Mandreolli et al 2007b]

O SUNRISE permite ao usuário, por meio de uma interface gráfica, explorar os caminhos mais promissores durante uma busca. Nessa interface, o usuário pode indicar o *peer* e o conceito a ser explorado no início do processo, a condição de parada e a estratégia a ser explorada. São condições de parada: (a) máximo de saltos (b) objetivo satisfeito (uma medida de qualidade dos caminhos a serem explorados). Com relação às estratégias de busca o usuário poderá escolher entre randômico, baseado no mapeamento semântico (explorando apenas os vizinhos do mapeamento) e baseado em

SRI (utilizando os índices semânticos). Na Figura 8, o *peer A* é o ponto inicial de busca, o conceito solicitado é *paper* e a estratégia baseada em SRI. O SRI do *peer A* indica que a direção mais promissora é o *peer C*. Logo, o *peer C* é escolhido e o processo é iniciado, *peer C* torna-se o *peer* corrente e o conceito é atualizado conforme o mapeamento dos esquemas. O conceito *paper* do *peerA* torna-se o conceito *article* para o *peerC*. Em seguida, o usuário pode continuar a busca ou finalizar.

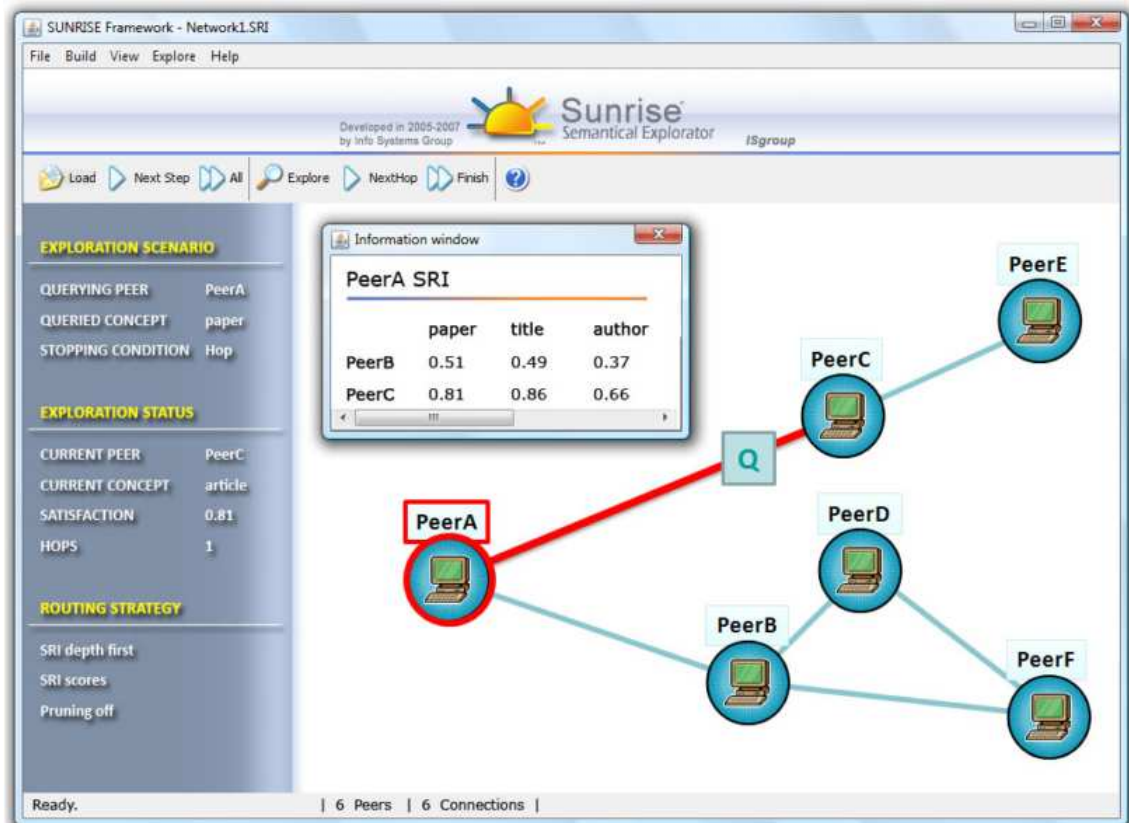


Figura 8: Interface gráfica do SUNRISE [Mandreolli et al 2007b]

### 3.4. H-Link [Montanelli 2007]

O H-link é um mecanismo de roteamento semântico para sistemas P2P cuja principal característica está no uso de ontologias para representação do conhecimento dos *peers*, de técnicas de correspondência de ontologias para seleção de *peers* semanticamente similares e gerenciamento independente pelos *peers* de sua própria ontologia. A idéia chave do H-Link é explorar os resultados das interações durante a descoberta de conhecimento para treinar o comportamento do mecanismo de roteamento. Para alcançar esse objetivo, os *peers* são conectados por meio de medidas de confiança baseadas em técnicas de correspondência que acompanham a afinidade semântica entre os conteúdos dos diferentes *peers*. Logo, os *peers* são organizados em uma SON (*Semantic Overlay Network*) cujos *peers* têm conhecimento similar e estão interligados como vizinhos semânticos.



A ontologia do *peer* no H-Link provê uma descrição formal do contexto do *peer* em termos de (1) o conhecimento local dos recursos do *peer* que serão compartilhados com os outros *peers*, e (2) o conhecimento do *peer* sobre a rede que é o conhecimento do *peer* sobre seus vizinhos semânticos progressivamente descobertos ao longo das interações que ocorrem na rede.

Para ilustrar esse mecanismo, o autor propõe um exemplo, Figura 9, onde cada *peer* é independente e se uniu ao sistema por meio de sua própria ontologia. Suponha que o *peer* A está interessado em localizar outros *peers* que possuam recursos semanticamente relacionados ao domínio de publicação. Para isso, o *peer* A formula uma consulta Q1 e submete ao sistema contendo os conceitos de interesse *Publication* e *Book* com suas respectivas propriedades *year* e *author*. Ao receber a consulta Q1 os *peers* B, C e D usam o gerador de correspondência semântica (H-Match) para comparar a descrição dos conceitos na consulta e comparar com a ontologia do *peer*. Conforme o resultado desse processo, os *peers* B e D enviam ao *peer* A uma lista com os conceitos encontrados ordenados, e para cada entrada, o cálculo do valor de afinidade semântica (SA). Como pode-se observar na Figura 9, apenas os *peers* B e D retornam valores enquanto que C não responde ao *peer* A por não possuir conceitos correspondentes.

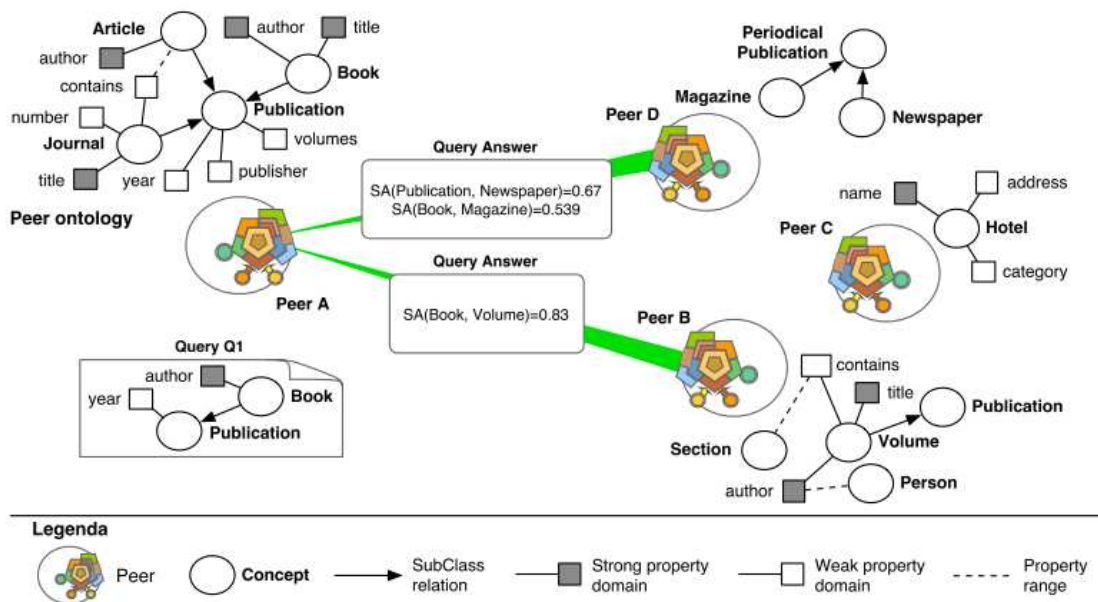


Figura 9: Mecanismo de roteamento H-Link

É importante destacar que a resposta fornecida ao *peer* A poderá ser utilizada em interações futuras. Quando consultas similares forem feitas, esses resultados contribuirão no processo de identificação dos melhores *peers* disponíveis na rede, aptos a responder a consulta. [Castano, Nontanelli 2008]

### 3.5. OntoSum [Li,Vuong 2007]

É um mecanismo de roteamento de consultas em sistemas P2P baseado na utilização de um índice da sumarização da ontologia dos *peers* que compõe a rede. Esse mecanismo assume que cada *peer* pode usar sua própria ontologia para descrever o conhecimento relativo aos seus recursos e que a topologia da rede será ajustada de acordo com as propriedades ontológicas de cada *peer*. Uma estratégia de indexação permite encaminhar as consultas apenas aos *peers* semanticamente relacionados. A organização dos *peers* na rede se apresenta da seguinte forma: *peers* com conteúdo similar formam um grande domínio e dentro desse domínio, os *peers* podem agrupar-se em *clusters* se eles compartilharem da mesma ontologia conforme pode-se observar na Figura 10.

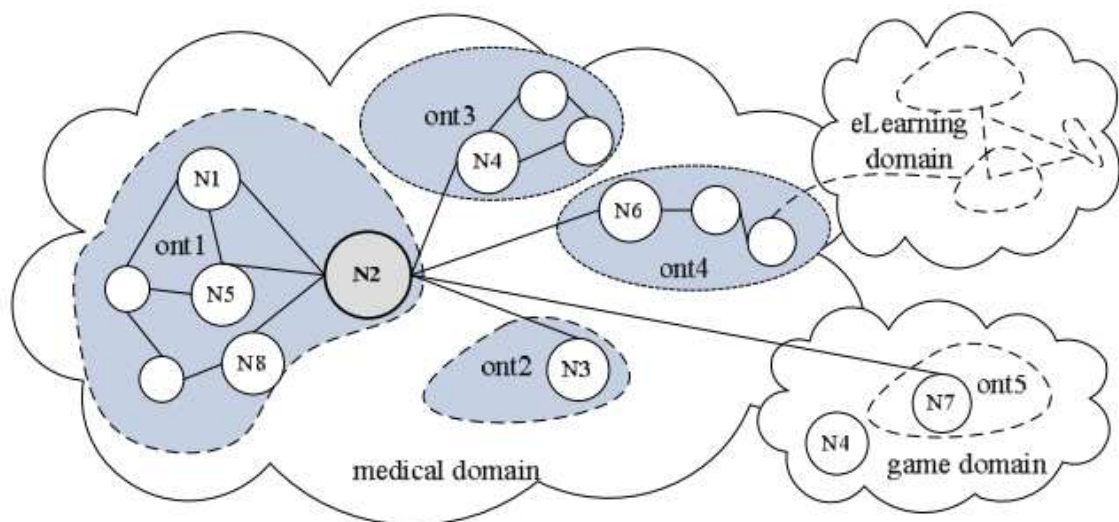


Figura 10: Topologia da Rede [Li,Vuong 2007]

Na Figura 10, todos os *peers* do domínio *medical* estão interessados nos diferentes aspectos dos recursos médicos, e cada um dos *peers* pode estar usando diferentes ontologias para descrever seus recursos. Entretanto, se eles compartilham o mesmo interesse, eles estarão conectados uns aos outros. No mesmo domínio *medical*, observamos que os *peers* N1, N2, N5 e N8 estão agrupados porque utilizam a mesma ontologia ont1. Nessa rede estruturada multi-nível, por exemplo, um *peer* B vizinho de A pode distinguir três tipos de vizinhos baseado na sua similaridade semântica: (1) vizinho com distância zero (ou vizinho de mesma ontologia, vizinho *intra-cluster*), se  $sim(A,B)=1$  (similaridade semântica entre A e B é igual a 1); (2) vizinho com curta distancia (ou semanticamente relacionado), se  $sim(A,B) \geq t$  ( $0 < t < 1$  é o limite semântico de A); (3) vizinho com longa distância (ou vizinhos não relacionados semanticamente),

se  $sim(A,B) < t$ . O princípio básico é permitir que um *peer* sempre consiga achar os vizinhos mais próximos, mesmo aqueles que estão em distâncias maiores alcançáveis apenas por outros *clusters*.

Para o roteamento eficiente das consultas, é proposto nesse trabalho, dois esquemas de roteamento: roteamento *inter-cluster*, para rápida localização dos *clusters* relacionados semanticamente; e roteamento *intra-cluster*, para localização dos recursos que satisfazem às restrições da consulta. Para construir tais esquemas, duas tabelas de roteamento são mantidas em cada *peer*: tabela de roteamento *inter-cluster* (*inter-table*) e tabela de roteamento *intra-cluster* (*intra-table*). Essas tabelas de roteamento mantêm um conhecimento mais refinado sobre os vizinhos. Essa é uma das características dessa estratégia de roteamento: a consulta inicialmente, caminha sobre a rede, e após atingir o domínio de destino amplia esse domínio e investiga as propriedades que possam indicar seus vizinhos semânticos.

A Tabela 1, mostra a representação de uma tabela de roteamento *inter-cluster* para o *peer* N2. Os *peers* N3, N4, e N6 são vizinhos de curta distancia de N2. O *peer* N7 é um vizinho de longa distancia que tem um domínio semântico não relacionado a N2. Os vetores com assinatura de vizinhos semânticos são comprimidos em um *Bloom filter* como seqüências de 0s e 1s. A última coluna da tabela armazena os mapeamentos inter ontologias entre N2 e os outros vizinhos semanticamente relacionados.

neighbor	semantic similarity	compressed signature vector	Inter-ontology mappings
N3	0.8	ont2 [1001010...]	$Ca=Ca', P1=P1' \dots$
N4	0.7	ont3 [0111010...]	$Cm \supset Cm', P2 \supset P2'$
N6	0.6	ont4 [1100010...]	$Ct \subset Ct' \dots$
N7	0	ont5 [0001010...]	none

Tabela 1: Tabela de roteamento *inter-cluster* para o *peer* N2

Após o *cluster* ter sido identificado por meio da tabela de roteamento *inter-cluster*, a tabela *intra-cluster* será usada para encaminhar a consulta dentro do *cluster*. Essa tabela inclui um sumário dos recursos sobre os vizinhos dentro do mesmo *cluster*, que podem ser localizados a partir de determinado *peer*.

O roteamento da consulta dentro do *cluster* é baseado em um algoritmo de roteamento sobre um vetor de distancia de recurso (RDV). Cada *peer* mantém uma tabela de índices de recursos com informações sobre a distância de cada recurso (em

número de saltos). Quando um *peer* recebe uma consulta, o algoritmo escolhe a rota mais próxima e encaminha a consulta. Um exemplo pode ser observado na Figura 11. O *peer* A recebe uma consulta para o recurso que está mapeado para as posições 3 e 6. Ele checa sua tabela de roteamento e acha 2 casamentos: o *peer* C com 2 saltos; e o *peer* D com 3 saltos. De acordo com o algoritmo o caminho mais curto será o escolhido, no caso, pelo *peer* C. A consulta é encaminhada para C que de forma similar repetirá o processo e encaminhará para E onde a consulta é satisfeita.

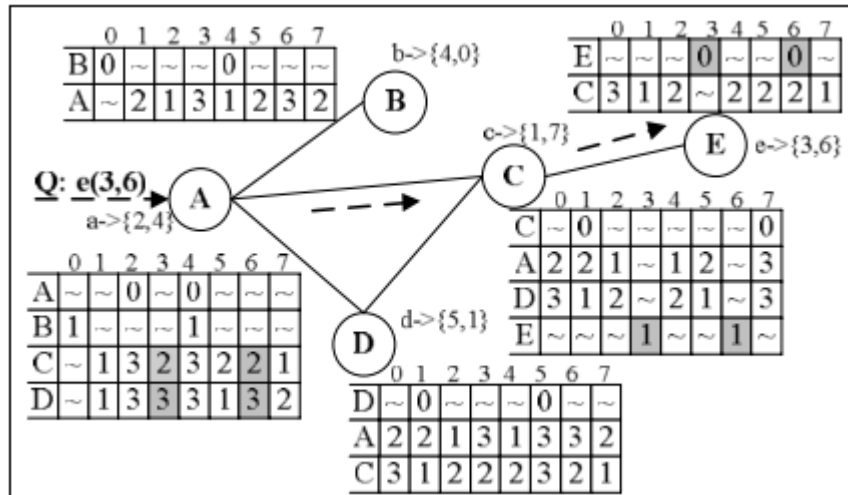


Figura 11: exemplo de roteamento da consulta utilizando o vetor de distância de recurso (RDV)

### 3.6. SEWASIE [Beneventano et al. 2007]

O projeto SEWASIE (*SEmantic Web and AgentS in Integrated Economies*) é baseado na tecnologia de sistema multi-agentes. Sua arquitetura prevê duas camadas: uma de nível local, onde os *peers* (desenvolvidos como agentes mediadores) mantêm uma visão integrada de suas fontes locais; outra, no nível de rede, onde agentes (*brokering agents*) mantêm os mapeamentos entre os diferentes *peers*. A Figura 12 oferece uma visão macro da arquitetura do SEWASIE definidos da seguinte forma [SEWASIE 2011]:

- SEWASIE *Information Nodes* (*SINodes*) – são sistemas baseados em mediadores que fornecem uma visão virtual das fontes de informação gerenciadas pelo *SINode*. Os *SINodes* utilizam tradutores (*wrapper*) para extrair dados e metadados das fontes. Um construtor de ontologias (*Ontology Builder*) é utilizado para criar uma ontologia integrada de todos os esquemas das fontes, denominado *Global Virtual View* (GVV).

Essas ontologias serão futuramente integradas ao *Brokering Agent* para estabelecer uma ligação entre os *SINodes* e a interface do usuário.

- *Brokering Agents* (BA) – responsável por integrar as GVV de diferentes *SINodes* em uma *Brokering Agent Ontology* (BA *Ontology*) e fazer o roteamento das consultas na rede. A ontologia do BA é usada para guiar os *Query Agents* para os *SINodes* que contem dados relevantes a consulta. É possível que no sistema existam vários BA, cada um representando um domínio específico. Mapeamentos entre diferentes BA podem ser criados e dessa forma fazer com que novos encaminhamentos de consultas possam ser realizados. Logo, o SEWASIE se apresenta com uma arquitetura *super-peer*, onde os BA agem como *super-peers*, e os *SINodes* como *peers* de dados.
- *Query Agent* – após receber a consulta (expressa como uma ontologia do BA), reescreve em função do GVV do *SINode* (identificado pelo BA) e faz o envio.
- *User Agent* (UA) – o usuário interage com uma interface web gerenciada pelo UA que disponibiliza uma lista das ontologias dos BA disponíveis para que a consulta possa ser formulada. O UA instancia um QA que traduz, por meio dos BA, a consulta do usuário em um conjunto de consultas que deverão ser realizadas no nível dos *SINodes*. No retorno da consulta o QA integra as respostas e retorna os dados no formato XML para o UA. [Beneventano et al. 2007]

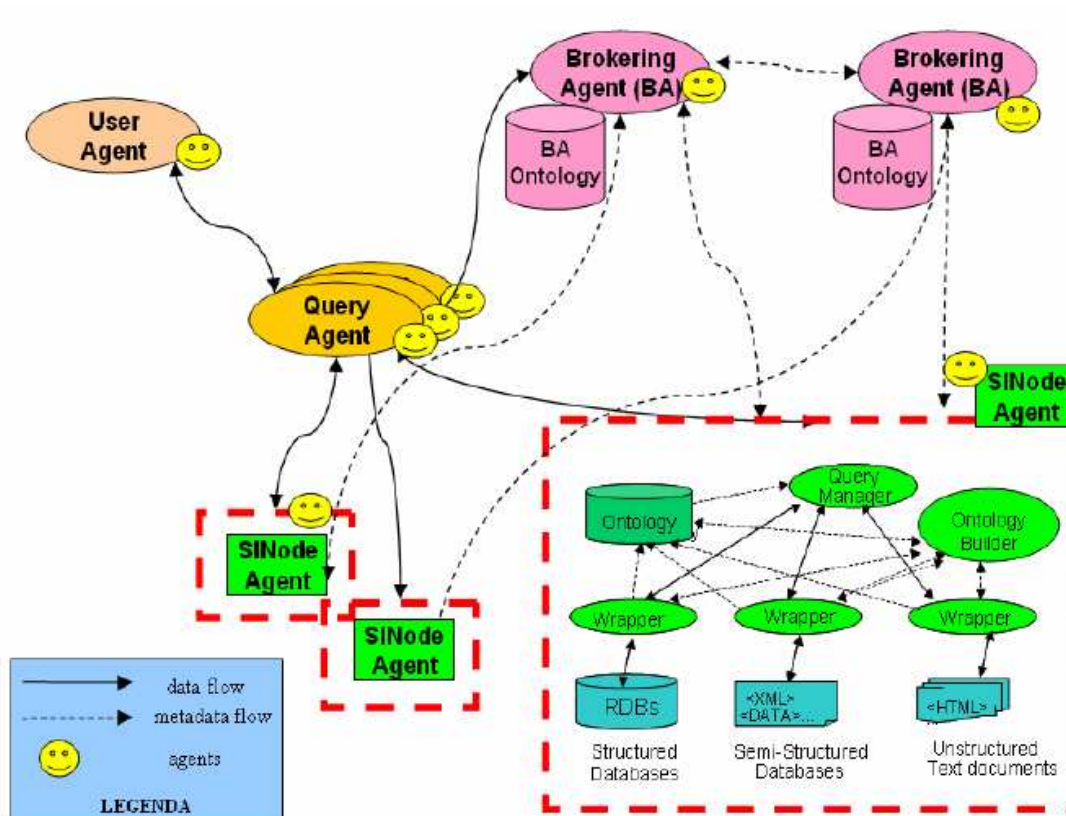


Figura 12 - Arquitetura SEWAISE (SEWAISE, 2011)

#### 4. Quadro resumo dos artigos apresentados

O quadro a seguir fornece um resumo dos trabalhos apresentados conforme algumas características: Organização da rede (organização dos *peers* no sistema, forma de agrupamento semântico); Localização de conteúdo (forma utilizada para identificação e organização dos vizinhos semânticos); Estratégia de Roteamento (estratégia utilizada para o encaminhamento das consultas entre os *peers*); Outras informações (anotações complementares de dados relevantes do trabalho).

Artigos	Organização da rede	Localização de vizinhos	Estratégia de Roteamento	Outras informações
Árvore de Decisão e Hipergrafos Transversais [Ismail et al. 2010]	<i>super-peer</i> e agrupamentos por similaridade de conteúdo em KSP ( <i>knowledge super peer</i> )	Árvore de decisão	Algoritmos baseados em Hipergrafos transversais	Agrupamento de <i>super-peers</i> forma um conjunto de hipergrafos com no mínimo um <i>super-peer</i> em comum usado para encontrar a transversalidade entre os agrupamentos
Ontozilla [Joung, Chuang 2009]	Agrupamentos em SIG ( <i>Special Interest Groups</i> )	Tabela de roteamento (links semânticos e <i>peers</i> virtuais)	<i>Intra-SIG</i> <i>Inter-SIG</i>	SIG representado por ontologias; utiliza o conceito de classe e super-classe
SRI - <i>Semantic Routing Index</i> [Mandreolli et al. 2007b]	<i>Semantic Overlay Network</i> similaridade de conteúdo	Índices de Roteamento Semântico (SRI)	Randômico (obedecendo os mapeamentos semânticos) Índices de roteamento	Cada <i>peer</i> mantém um resumo das informações a respeito do grau de similaridade semântica entre os conceitos armazenados entre ele e seus respectivos vizinhos
H-Link [Montanelli 2007]	<i>Semantic Overlay Network</i> com conhecimento similar e <i>peers</i> interligados como	<i>Match</i> de ontologias e <i>links</i> estabelecidos por meio de medidas de afinidade	Explora os resultados contidos na camada de conhecimento da rede ( <i>network knowledge layer</i> ) e	Ontologia do peer organizada em duas camadas: <i>network knowledge layer</i> e <i>content knowledge layer</i> .

	vizinho semânticos		o uso do sistema de correspondência <i>H-Match</i>	
OntoSum [Li, Vuong 2007]	Agrupamentos por domínio de conhecimento e clusters com mesma representação ontológica	Índice de sumarização	<i>Inter-cluster</i> <i>Intra-cluster</i>	Ontologia para descrever o domínio do <i>peer</i> ; o roteamento <i>intra-cluster</i> faz uso de um RDV (Vetor de distância de recurso)
SEWASIE [Beneventano et al. 2007]	<i>super-peer (Brokering Agent-GVV e SINodes-GVV)</i>	<i>SINodes</i> (informação integrada das fontes locais); <i>Broker Agent</i> (informações dos <i>SINodes</i> e de outros <i>Broker Agent</i> )	Comunicação entre agentes ( <i>Brokering Agent, Query Agent e SINodes</i> )	Tecnologia de sistemas multi-agente



## 5. Considerações

Outros sistemas não abordados nesse trabalho de forma mais detalhada trazem propostas também interessantes para o roteamento semântico em sistemas P2P, como o REMINDIN' [Staab et al. 2004] que provê um mecanismo de roteamento baseado em ontologias e seleção de *peers* a partir da memorização dos resultados de sucesso obtidos nas respostas fornecidas por outros *peers*. O NeuroGrid [Joseph 2002], onde cada peer mantém uma base de conhecimento que contém associações entre *peers* e palavras-chave, tal base é utilizada durante o processo de busca de recursos na rede. O QSummary [Hose et al 2006], que utiliza índice de roteamento combinando estruturas em árvore e histogramas para tornar mais eficiente o processo de busca, reduzindo o custo de execução, e propondo uma estratégia de manutenção eficiente desses índices.

É fato que, um dos grandes desafios em soluções para sistemas P2P está em tornar eficiente o processamento da consulta. Rotear uma consulta nesses ambientes trata diretamente com questões do tipo: (1) localização de conteúdo, decidir para quais outros pontos a consulta deve ser encaminhada de forma a responder com eficiência e eficácia; (2) sistemas que espalham todas as consultas (algoritmos de inundação) para todos os *peers* sofrem com questões de eficiência e escalabilidade; (3) consultas podem ter qualquer forma e conforme o ponto onde tenha sido submetida é preciso conhecer e confiar nos vizinhos semânticos para os quais serão encaminhadas; (4) tempo de resposta; (5) crescimento e dinamismo da rede (entrada e saída de *peers* e *super-peers*). E mais, em soluções de roteamento que utilizam índices é preciso garantir a manutenção das tabelas, e considerar que até mesmo os processos de balanceamento e formação da rede podem ter impacto direto no mecanismo de roteamento utilizado.

Dessa forma, as pesquisas relacionadas a mecanismos de roteamento vêm se tornando um grande desafio. É preciso investigar melhor as soluções de roteamento semântico aplicadas aos sistemas P2P. Além disso, pode-se verificar uma grande necessidade em pesquisar métricas que possam ser utilizadas para medir a relevância e qualidade de um ponto em determinada consulta.

## 6. Referências

ABERER K., CUDRÉ-MAUROUX P., DATTA A., DESPOTOVIC Z., HAUSWIRTH M., PUNCEVA M., SCHMIDT R. (2003). "P-GRID: A SELF-ORGANIZING STRUCTURED P2P SYSTEM". ACM SIGMOD RECORD, 32(3), 29-33

AKBARINIA R., PACITTI, E., VALDURIEZ P.(2007) "Query processing in P2P systems" Relatório Técnico de Pesquisa. no. 6112, INRIA – Institute National de Recherche en Informatique et en Automatique,. – ISSN 0249-6399

BAYARDO R. J., BOHRER W., Brice R. S., CICHOCKI A., Fowler J., HELAL A., KASHYAP V., KSIEZYK T., MARTIN G., NODINE M. H., RASHID M., RUSINKIEWICZ M., SHEA R., UNNIKRIISHNAN C., UNRUH A., WOELK D. (1997) "Infosleuth: semantic integration of information in open and dynamic environments" (experience *paper*). In: Peckham J (ed) SIGMOD 1997, Proceedings ACM SIGMOD international conference on management of data, USA, ACM Press, pp 195– 206

BENEVENTANO D., BERGAMASCHI S., GUERRA F., VINCINI M. (2007). "The SEWASIE Network of Mediator Agents for Semantic Search". Journal of Universal Computer Science, vol. 13, no. 12, pp.1936-1969.

BERNSTEIN, P. A.; GIUNCHIGLIA, F.; KEMENTSIETSIDIS, A.; MY- LOPOULOS, J.; SERAFINI, L.; AND ZAIHRAYEU, I. (2002). "Data management for peer-to-peer computing: A vision." In Proc. of the 5th Int. Workshop on the Web and Databases (WEBDB 2002)

CASTANO, S, MONTANELLI S., (2005) "Semantic Self-Formation of Communities of *peers*". In proc of the 2nd IEuropean Semantic Web Conference (ESWC), Greece

CASTANO, S, MONTANELLI S., (2006) "Enforcing a Semantic Routing Mechanism based on *peer* Context Matching". In proc of the 2nd International Workshop on Contexts and Ontologies: Theory, Practice and Applications, Italy

CASTANO, S, MONTANELLI S., (2008). "Semantically routing queries in *peer*-based systems: the H-Link approach". In Journal The *Knowledge Engineering Review*, vol 23, PP 51-72

CIGLARIC, M.; VIDMAR, T. (2006) "Ant-inspired query routing performance in dynamic peer-to-peer networks. Parallel and Distributed Processing Symposium, 20th International, IEEE Computer Society, pp. 287, 2006.

CLARKE I., MILLER S., HONG T. W., SANDBERG O., WILEY B. (2002). "Protecting free expression online with Freenet." IEEE Internet Computing, 6(1), 40-49.

COSTA, L. R. (2009). Roteamento de consultas em bancos de dados *peer-to-peer* utilizando colônias de formigas e ontologias. Dissertação de Mestrado. Universidade Estadual Paulista. São José do Rio Preto, Brasil.

CRESPO A., GARCIA-MOLINA H. (2003) "Semantic Overlay Networks for P2P Systems". In Proceedings of the 29<sup>th</sup> VLDB Conference, Germany.

HAASE P., SIEBES R., HARMELEN F. (2008). "Expertise-Based Peer Selection". In Journal of Knowledge and Information Systems (1):75-107

HAASE P., SIEBES R., HARMELEN F. v. (2008). " Expertise-based *peer* selection in *peer-to-peer* networks". Institute AIFB, University of Karlsruhe, 76128, Germany

HALEVY A., IVES Z., MADHAVAN J., MORK P., SUCIU D., TATARINOV I. (2004) The Piazza *peer* Data Management System. IEEE TKDE, 16(7):787–798, 2004.

HALEVY, A. Y. (2000). "Theory of answering queries using views". SIGMOD RECORD 29(4):40–47.

HALEVY, A.; IVES, Z.; SUCIU, D.; AND TATARINOV, I. (2003). "Schema mediation in peer data management systems". In Proc. of the 19th IEEE Int. Conf. on Data Engineering (ICDE 2003). pp 505–516.

HEFLIN, J., HENDLER, J. (2001). "A portrait of the semantic web in action". IEEE INTELLIGENT Systems 16(2):54–59.

HOSE K., KLAN D., SATTLER K. (2006) "Distributed Data Summaries for Approximate Query Processing in PDMS". In Proc of 10th International Database Engineering and Applications Symposium (IDEAS'06), Delhi, India

ISMAIL A., QUAFAROU M., DURAND N., NACHOUKI G. HAJJAR M. (2010). "Queries Mining for Efficient Routing in P2P Communities". In Proc. of Journal of Database Management Systems (IJDMS), Vol.2, No.1, February 2010

JOSEPH S. (2002), 'NEUROGRID: Semantically Routing Queries in peer-to-peer Networks', in

Proc. of the Int. Workshop on peer-to-peer Computing , PISA, ITALY..

JOUNG Y, CHUANG F. (2009). “OntoZilla: An ontology-based, semi-structured, and evolutionary *peer-to-peer* network for information systems and services” in Journal Future Generation Computer Systems, vol 25, no. 1, pp 53-63

KAMIENSKI C., SOUTO E., ROCHA J., DOMINGUES M., CALLADO A., SADOK D. (2005). “Colaboração na Internet e a Tecnologia Peer-to-Peer”. In: XXV Congresso da Sociedade Brasileira de Computação – SBC2005. 25 a 29 Jul. São Leopoldo – RS.

LENZERINI, M. (2002). “Data integration: A theoretical perspective”. In Proc. of the 21st ACM SIGACT SIG- MOD SIGART Symp. on Principles of Database Systems (PODS 2002), 233–246.

LI J., VUONG S. (2007). “OntSum : A Semantic Query Routing Scheme in P2P Networks Based on Concise Ontology Indexing” in Proc of 21st International Conference on Advanced Networking and Applications(AINA'07), pp 94-101, Ontario, Canada.

MANDREOLI F., MARTOGLIA R., PENZO W., SASSATELLI S, VILLANI G. (2007b) “SUNRISE : Exploring PDMS Networks with Semantic Routing Indexes” in Proc of eswc07

MANDREOLI, F., MARTOGLIA, R., PENZO, W., SASSATELLI, S. (2007a) “SRI @ work : Efficient and Effective Routing Strategies in a PDMS” in Proc. of Web Information Systems Engineering – WISE 2007 8th International Conference on Web Information Systems Engineering, pag 285-297, Nancy, France.

MICHLMAYR, E. (2006) “Ant algorithms for search in unstructured peer-to-peer” networks. In: ICDEW '06: Proceedings of the 22nd International Conference on Data Engineering Workshops. USA: IEEE Computer Society, pp. 142–146.

MONTANELLI S., CASTANO S. (2008) “Semantically routing queries in *peer*-based systems: the H-Link approach”. The *Knowledge Engineering Review*, vol. 23:1, 51-72. Cambridge University Press

NAPSTER. (2010). Acesso em 20 denovembro de 2010, disponível em <http://www.napster.com>

NEJDL W, WOLPERS M, SIBERSKI W, SCHMITZ C, SCHLOSSER M, BRUNKHORST I, LOSER A (2003) *super-peer*-based routing and *clustering* strategies for rdf-based *peer-to-peer* networks. In: Proceedings of the 12th international world wide web conference, Budapest,

Hungary, cite- seer.nj.nec.com/nejdl02superpeerbased.html

PIRES, C. (2009). Ontology-based *Clustering* in a *peer* Data Management System. Tese de Doutorado. Universidade Federal de Pernambuco. Recife, Brasil.

RATNASAMY S., FRANCIS P., HANDLEY M., KARP R., SHENKER S. (2001) "A scalable content-addressable network". ACM SIGCOMM Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications, 161-172, INRIA

ROWSTRON A., DRUSCHEL p. (2001). "Pastry: scalable, distributed object location and routing for large-scale peer-to-peer systems". IFIP/ACM Int. Conf. on Distributed Systems Platforms (Middleware), 329-350..

SCHLOSSER M. T., SINTEK M., DECKER S, NEJDL W.(2002) Hypercup- hypercubes, ontologies, and efficient search on *peer-to-peer* networks. In: Moro G, Koubarakis M (eds) AP2PC, vol 2530 of lecture notes in computer science, Springer, pp 112–124

SEWAISE. (2011). "SEmantic Webs and AgentS in Integrated Economies". Acesso em 3 de março de 2011, disponível em <http://www.sewaise.org>

STAAB S., TEMPICH C., WRANIK A. (2004). "REMINDIN': Semantic Query Routing in *peer to-peer* Networks based on Social Metaphors.) In Proc. of the 13th Int. conference on World Wide Web (WWW 2004), New York, USA

STOICA I., MORRIS R., KARGER D.R., KAASHOEK M.F., BALAKRISHNAN H. (2001). "Chord: a scalable peer-to-peer lookup service for internet applications". ACM Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM), pp. 149-160.

ZHAO B. Y., HUANG L., STRIBLING J., RHEA S.C., JOSEPH A. D., KUBIA-TOWICZ J.D. (2004) "Tapestry: a resilient global-scale overlay for service deployment". IEEE Journal on Selected Areas in Communications (JSAC), 22(1), 41-53, 2004.