

## Referências Bibliográficas

ALHADI, N., AHMAD, K. **Query Tuning in Oracle Database**. Journal of Computer Science, 8(11): 1889-1896, 2012.

BARRIENTOS, A. Y. M. **Uma arquitetura para auto-sintonia global de SGBDs usando agentes**. Dissertação de mestrado apresentada ao programa de pós-graduação em informática da PUC-Rio, 2004.

BELLATRECHE, L., KHOURI, S., BOUKHARI, I., BOUCHAKRI, R. **Using ontologies and requirements for constructing and optimizing data warehouses**. Proceedings of the 35th International Convention, pp. 1568 – 1573, 2012.

BELLATRECHE, L., SCHNEIDER, M., LORINQUER, H., MOHANIA, M. **Bringing together partitioning, materialized views and indexes to optimize performance of relational data warehouses**. Proceeding of the International Conference on Data Warehousing and Knowledge Discovery (DAWAK'2004), pp. 15-25, Springer Berlin Heidelberg, 2004.

BOROVICA, R., ALAGIANNIS, I., AILAMAKI, A. **Automated Physical Designers: What You See is (Not) What You Get**. Proceedings of the ACM SIGMOD, The Fifth International Workshop on Testing Database Systems (DBTest'12), pp. 9, ACM, 2012.

BOTTAZZI, E., FERRARIO, R. **Preliminaries to a DOLCE Ontology of Organizations**. International Journal of Business Process Integration and Management, v. 1, pp. 64-74, 2006.

BOUCHAKRI, R., BELLATRECHE, L. **On Simplifying Integrated Physical Database Design**. Proceedings of the 15th East European Conference on Advances in Databases and Information Systems (ADBIS), 2011.

BRUNO, N. **Automated Physical Database Design and Tuning**. CRC Press, 2011.

BRUNO, N., CHAUDHURI, S., KÖNIG, A. C., NARASAYYA, V., RAMAMURTHY, R., SYAMALA, M. **AutoAdmin Project at Microsoft Research: Lessons Learned**. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 34(4), pp. 12-19, 2011.

CAO, W., SHASHA, D. **AppSleuth: a Tool for Database Tuning at the Application Level**. Proceedings of the 16th International Conference on Extending Database Technology - EDBT/ICDT'13, pp. 589-600, 2013.

CHAUDHURI, S., NARASAYYA, V. **Self-tuning Database Systems: A Decade of Progress**. Proceedings of the 33<sup>rd</sup> International Conference on Very Large Databases (VLDB'07), pp. 3-14, 2007.

COOPER, L., WALLS, R. L., ELSER, J., GANDOLFO, M. A., STEVENSON, D. W., SMITH, B., PREECE, J., ATHREYA1, B., MUNGALL, C. J., RENSING, S., HISS, M., LANG, D., RESKI, R., BERARDINI, T. Z., LI, D., HUALA, E., SCHAEFFER, M., MENDA, N., ARNAUD, E., SHRESTHA, R., YAMAZAKI, Y., JAISWAL, P. **The Plant Ontology as a Tool for Comparative Plant Anatomy and Genomic Analyses**. *Plant and Cell Physiology*, 54(2): e1, 2013.

CORCHO, O., FERNANDEZ-LOPEZ, M., GOMEZ-PEREZ, A. **Methodologies, Tools and Languages for Building Ontologies: Where is their meeting point?**. *Journal Data & Knowledge Engineering*, 46(1): 41–64, 2003.

COSTA, R. L. C., LIFSCHITZ, S., NORONHA, M. AND SALLES, M. V. **Implementation of an Agent Architecture for Automated Index Tuning**. *Proceedings of the 21<sup>st</sup> International Conference on Data Engineering Workshops (ICDEW'05)*, pp. 1215, 2005.

DATE, C. J. **Introdução a sistemas de bancos de dados**. Tradução da 8ª edição americana, editora Campus, 2004.

DIAS, K., RAMACHER, M., SHAFT, U., VENKATARAMANI, V., WOOD, G. **Automatic Performance Diagnosis and Tuning in Oracle**. *Proceedings of the 2005 CIDR Conference*, pp. 84-94, 2005.

ELGHANDOUR, I. **Automatic Physical Design for XML Databases**. Tese de doutorado apresentada em University of Waterloo, Ontario, Canada, 2010.

ELMASRI, R., NAVATHE, S. B. **Sistemas de banco de dados**. 6ª edição, editora Pearson Education do Brasil, 2011.

EVERETT, J. O., BOBROW, D.G., STOLLE, R., CROUCH, R., PAIVA, V., CONDORAVDI, C., BERG, M. V. D., POLANYI, L. **Making Ontologies Work for Resolving Redundancies across Documents**. *Magazine Communications of the ACM – Ontology: different ways of representing the same concept*, 45(2): 55-60, 2002.

FACT++. **The Tableaux-based Reasoner for Expressive Description Logics (DL)**. Disponível em <<http://owl.cs.manchester.ac.uk/fact++/>>. Acesso em: agosto, 2013.

FERNÁNDEZ-LÓPEZ, M., GÓMEZ-PÉREZ, A. **Overview and Analysis of methodologies for building ontologies**. *Journal The Knowledge Engineering Review (KER)*, 17(2): 129-156, 2002.

FRIEDMAN-HILL, E. **Jess, the Rule Engine for the Java™ Platform**. Ferramenta desenvolvida em Sandia National Laboratories, Livermore, CA, disponibilizada em <<http://www.jessrules.com/jess/index.shtml>>. Acesso em: agosto, 2013.

GARCIA-MOLINA, H., ULLMAN, J. D., WIDOM, J. D. **Database Systems: The Complete Book**. Prentice Hall, 2001.

GO. **The Gene Ontology**. Disponível em <<http://www.geneontology.org/>>. Acesso em: agosto, 2013.

GOASDOUÉ, F., KARANASOS, K., LEBLAY, J., MANOLESCU, I. **View Selection in Semantic Web Databases**. Proceedings of the VLDB Endowment, 5(2): 97-108, 2011.

GRUBER, T. **Ontology**. Encyclopedia of Database Systems, Liu. L., Özsu. M. (eds.), Springer-Verlag, 2009.

GRUNINGER, M., ATEFI, K., FOX, M.S. **Ontologies to support process integration in enterprise engineering, Computational and Mathematical Organization Theory**. Journal Computational & Mathematical Organization Theory, 6(4): 381-394, 2000.

GUARINO, N. **Formal Ontology and Information Systems**. Proceedings of the First International Conference on Formal Ontology in Information Systems, pp. 3-15, 1998.

GUARINO, N. **Understanding, building and using ontologies**. International Journal of Human-Computer Studies, 46(2-3): 293-310, 1997.

GUIZZARDI, G., FALBO, R. A., GUIZZARDI, R. S. S. **A importância de Ontologias de Fundamentação para a Engenharia de Ontologias de Domínio: o caso do domínio de Processos de Software**. Revista IEEE América Latina, vol. 6, pp. 244-251, 2008.

HERRE, H., HELLER, B., BUREK, P., HOEHNDORF, R., LOEBE, F., MICHALEK, H. **General Formal Ontology (GFO) a foundational ontology integrating objects and processes, Part 1: Basic Principals, version 1.0.1**. Onto-Med Report 8, University of Leipzig, 2006.

IKEDA, M., SETA, K., KAKUSHO, O., MIZOGUCHI, R. **Task ontology: ontology for building conceptual problem solving models**. Proceedings of the ECAI98 Workshop on Applications of ontologies and problem-solving models, pp. 126-133, 1998.

JASPER, R. USHOLD, M. **A framework for understanding and classifying ontology applications**. Proceedings of the 12th International Workshop on Knowledge Acquisition Modeling and Management (KAW'99), pp. 16-21, 1999.

JIMENEZ, I., SANCHEZ, H., TRAN, Q. T., PLYZOTIS, N. **Kaizen: A Semi-AUtomatic Index Advisor**. Proceedings of the SIGMOD Conference, pp. 685-688, ACM, Maio, 2012.

KAON2. **KAON2 Infrastructure**. Disponível em <<http://kaon2.semanticweb.org>>. Acesso em: agosto, 2013.

KARIMI, V. **Semantic Web Rule Language (SWRL)**. Apresentação elaborada para disciplina lecionada em University of Waterloo, Ontario, Canada, 2008.

KHOURI, S., BELLATRECHE, L., BOUKHARI, I., BOUARAR, S. **More Investment in Conceptual Designers: Think about it!** Proceedings of the 15th IEEE International Conference on Computational Science and Engineering, pp. 88-93, 2012.

KNUTH, D. E. **The Art of Computer Programming: Sorting and Searching**. Vol. 3, Addison Wesley, 1973.

LENAT, D., GUHA, R.V. **Building Large Knowledge based Systems: Representation and Inference in the CYC Project.** Addison Wesley, 1990.

LIFSCHITZ, S., MILANES, A. Y. M., SALLES, M. V. **Estado da arte em auto-sintonia de SGBD relacionais.** Relatório técnico apresentado ao programa de pós-graduação em informática da PUC-Rio, 2004.

LOHMAN, G., VALENTIN, G., ZILIO, D., ZULIANI, M., SKELLEY, A. **Db2 Advisor: An optimizer smart enough to recommend its own indexes.** Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp. 101-110, 2000.

MARTINS, A. F. **Construção de ontologias de tarefa e sua reutilização na engenharia de requisitos.** Dissertação submetida ao Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo, 2009.

MONTEIRO, J. M. **Uma abordagem não-intrusiva para a manutenção automática do projeto físico de bancos de dados.** Tese submetida ao Programa de Pós-Graduação em Informática da PUC-Rio, 2008.

MONTEIRO, J. M., LIFSCHITZ, S., BRAYNER, A. **AIM-HypoPlans: Automatic Index Maintenance using Hypothetical Query Execution Plans.** Submetido para Information Systems para publicação, 2012.

MORELLI, E. M. T. **Recriação automática de índices em um SGBD Relacional.** Dissertação de mestrado apresentada ao programa de pós-graduação em informática da PUC-Rio, 2006.

MORELLI, E., ALMEIDA, A., LIFSCHITZ, S. MONTEIRO, J. M., MACHADO, J. **Autonomous re-indexing.** Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC '12), ACM Press, pp. 893-897, 2012.

NILES, I., PEASE, A. (2001). **Towards a Standard Upper Ontology.** Proceedings of Formal Ontology in Information Systems (FOIS 2001), Ogunquit, Maine, USA, pp 29, 2001.

NOY, N. F.; MCGUINNESS, D. L. **Ontology Development 101: A Guide to Creating Your First Ontology.** Knowledge Systems Laboratory, pp: 1-5, Stanford University, 2001.

O'CONNOR, M., KNUBLAUCH, H., TU, S., GROSOFF, B., DEAN, M., GROSSO, W., MUSEN, M. **Supporting rule system interoperability on the semantic web with SWRL.** Proceedings of the 4<sup>th</sup> International Conference on The Semantic Web (ISWC 2005), pp. 974-986, Springer Berlin Heidelberg, 2005.

OMG. **Acceleo - Transforming Models into Code.** Disponível em <<http://www.acceleo.org/>>. Acesso em: agosto, 2013.

ÖZSU, M. T., VALDURIEZ, P. **Princípios de sistemas de bancos de dados distribuídos.** Tradução da 2<sup>a</sup> edição americana, editora Campus, 2001.

PROTÉGÉ. **Protégé.** Ferramenta desenvolvida e suportada por National Institute of General Medical Sciences, Stanford Center for Biomedical

Informatics Research, disponibilizada em <<http://protege.stanford.edu/>>. Acesso em: agosto, 2013.

RAMAKRISHNAN, R., GEHRKE, J. **Database Management Systems**. Third edition, McGraw-Hill Higher Education, 2003.

RANGASWAMY, S., SHOBHA, G. **Online Indexing for Databases Using Query Workloads**. International Journal of Computer Science and Communication, 2 (2): 427-433, 2011.

RODD, S. F., KULKARNI, U. P., YARDI, A. R. **Adaptive neuro-fuzzy technique for performance tuning of database management systems**. Journal Evolving Systems, 4(2): 133-143, June, 2013.

SALLES, M. A. V. **Criação autônoma de índices em bancos de dados**. Dissertação de mestrado apresentada ao programa de pós-graduação em informática da PUC-Rio, 2004.

SCHNAITTER, K., POLYZOTIS, N. **Semi-Automatic Index Tuning: Keeping DBAs in the Loop**. Proceedings of the VLDB Endowment, 5(5): 478-489, 2012.

SHASHA, D., BONNET, P. **Database Tuning – Principles, Experiments, and Troubleshooting Techniques**. Morgan Kaufmann Publishers, Elsevier Science, 2003.

SIRIN, E., PARSIA, B., GRAU, B. C., KALYANPUR, A., KATZ, Y. **PELLET: A Practical OWL-DL Reasoner**. Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 5(2): 51-53, 2007.

SNOGGLE. **A Graphical, SWRL-based Ontology Mapper**. Disponível em <<http://snoggle.projects.semwebcentral.org/>>. Acesso em: agosto, 2013.

SOSNOVSKY, S., MITROVIC, A., LEE, D., BRUSILOVSKY, P., YUDELSON, M., BRUSILOVSKY, V., SHARMA, D. **Towards Integration of Adaptive Educational Systems: Mapping Domain Models to Ontologies**. Proceedings of the 6th International Workshop on Ontologies and Semantic Web for E-Learning (SWEL'2008) in Conjunction with ITS'2008, pp. 60-64, 2008.

STACKOVERFLOW. **SWRL rule using OR statements**. Disponível em <<http://stackoverflow.com/questions/6590406/swrl-rule-using-or-statements>>. Acesso em: agosto, 2013.

TPC. **Transaction Processing Performance Council**. Disponível em <<http://www.tpc.org/>>. Acesso em: agosto, 2013.

USCHOLD, M., GRUNINGER, M. **Ontologies: Principles, Methods and Applications**. The Knowledge Engineering Review, 11(2): p. 93-136, 1996.

VOBUGARI, S., SOMAYAJULU, D., SUBRAYA, B. **Index Tuning Through Query Evaluation Mechanism Based On Indirect Domain Knowledge**. Proceedings of the 14<sup>th</sup> International Conference, on Modelling and Simulation, pp. 630-635, 2012.

HORROCKS, I., PATEL-SCHNEIDER, P. F., BOLEY, H., TABET, S., GROSOFF, B., DEAN, M. **SWRL: A Semantic Web Rule Language - Combining OWL and RuleML**. National Research Council of Canada, Network Inference, and Stanford University, 2004.

WEIKUM, G., HASSE, C., MONKEBER, A., ZABBACK, P. **The COMFORT Automatic Tuning Project**. Invited project review, Information Systems, 16(5): 381-432, 1994.

ZHANG, G., CHEN, M., LIU, L. **A Model for Application-Oriented Database Performance Tuning**. Proceedings of the 6th International Conference on New Trends, Information Science and Service Science and Data Mining (ISSDM), pp. 389 – 394, 2012.

ZIVIANI, N. **Projeto de algoritmos com implementações em Java e C++**. São Paulo: Editora Thomson Learning, 2007.

## Anexo I

Nesse anexo são descritos e definidos os conceitos e propriedades levantados para a ontologia do domínio de sintonia fina e para o meta-modelo do *framework* (ontologia de tarefa de execução de heurísticas de sintonia fina).

### Termos e classes importantes da ontologia:

Classe	Definição
<b>Domínio de sintonia fina de banco de dados</b>	
SGBD	Sistema de gerenciamento de banco de dados. Responsável por tratar todas as requisições de acesso ao banco de dados. Essas requisições incluem a definição, armazenamento e recuperação de dados [Date, 2004]. As manipulações típicas incluem recuperação, inserção, exclusão e modificação dos dados [Elmasri et al, 2011].
ComandoDML	Comandos usados para transferir informações de e para o banco de dados, usando a linguagem de manipulação de dados (DML – <i>Data Manipulation Language</i> ) [Date, 2004].
Cláusula	Parte da instrução de um comando DML de acordo com os padrões SQL.
ExpressaoPredicado	Expressões lógicas contidas em cláusulas where que são usadas para restringir um subconjunto da relação.
Token	Representação abstrata de operadores que podem ser usados em cláusulas que fazem parte de um comando DML.
ObjetoSGBD	Componentes persistentes no banco de dados.
PlanoExecucao	Plano gerado pelo otimizador para processar um comando [Shasha et al, 2003]. Esse plano é gerado para executar grupos de operações com base nos caminhos de acesso disponíveis nos arquivos envolvidos na consulta [Elmasri et al, 2011]. É tipicamente uma árvore de operadores físicos que determinam precisamente como uma consulta deve ser executada. Os operadores físicos recebem sequencias de tuplas como entrada e produz como saída outra sequencia de tuplas (saída do operador raíz da árvore é o resultado da consulta) [Bruno, 2011].
TipoVarredura	Modos que o otimizador possui para percorrer uma relação.
EstruturaAcesso	Estruturas que permitem caminhos alternativos de acesso aos dados de uma relação.
EstruturaDados	Modo particular de armazenamento e organização dos dados [Ziviani, 2007].
ComandoDDL	Comandos usados para manipular os dados do catálogo/estrutura do banco, usando a linguagem DDL ( <i>Data Definition Language</i> ) [Shasha et al, 2003]

<b>Tarefa de execução de heurísticas de sintonia fina de banco de dados</b>	
Heurística	É um procedimento simplificador (embora não simplista) que, em face de questões difíceis envolve a substituição destas por outras de resolução mais fácil a fim de encontrar respostas viáveis, ainda que imperfeitas [Wikipedia, 2013].
Conceito	Qualquer item presente na ontologia de domínio de sintonia fina de banco de dados.
Fonte	Forma de obter as instâncias dos itens presentes na ontologia de domínio de sintonia fina de banco de dados.
Regra	Expressão lógica capaz de inferir algum conceito ou se for verdadeira, gerar uma ação no banco de dados.
Funcao	Função, ou seja, um conjunto de instruções capazes de gerar um conceito da ontologia de domínio ou executar ações no banco de dados.
Parametro	Dados de entrada que uma função precisa para ser executada ou dados que a função retorna como saída em resposta a sua execução. Para manter uma consistência e um vocabulário controlado, todo parâmetro precisa ser um item da ontologia de domínio.
Acao	Tarefas de sintonia fina a serem executadas no banco de dados.

### Hierarquias entre as classes:

Classe	Subclasse	Definição
<b>Domínio de sintonia fina de banco de dados</b>		
ComandoDML	Consulta	Tipo de comando para realizar a recuperação de informações de um banco de dados [Elmasri et al, 2011].
	Atualizacao	Usada para alterar os valores de um ou mais atributos em uma tupla(s) de alguma relação [Elmasri et al, 2011].
	Remocao	Proporciona a exclusão de dados de uma relação. Pode violar a integridade referencial se a tupla que estiver sendo excluída for referenciada por chaves estrangeiras de outras tuplas no banco de dados. Para especificar a exclusão, uma condição sobre os atributos da relação seleciona a(s) tupla(s) a ser(em) excluída(s) [Elmasri et al, 2011].
	Insercao	Oferece uma lista de valores de atributo para que uma nova tupla t possa ser inserida em uma relação R [Elmasri et al, 2011].
Cláusula	Select	Cláusula composta por uma lista de nomes de atributos, chamados de atributos de projeção, cujos valores devem ser recuperados pela consulta [Elmasri et al, 2011].
	From	Cláusula composta por um lista de nomes de relação exigidos para processar a consulta [Elmasri et al, 2011].

Cláusula	Where	Cláusula composta por expressão condicional (booleana) que deve ser verdadeira para identificar as tuplas a serem recuperadas [Elmasri et al, 2011].
	GroupBy	Cláusula que especifica os atributos de agrupamento. Possui a finalidade de particionar a relação em subconjuntos de tuplas (ou grupos) não sobrepostos. Cada grupo (partição) consistirá nas tuplas que possuem o mesmo valor de algum(ns) atributo(s) de agrupamento [Elmasri et al, 2011].
	OrderBy	Cláusula que especifica uma ordenação ou classificação das tuplas em alguma sequência definida de acordo com os atributos indicados [Date, 2004].
	Having	Oferece uma condição sobre a informação de resumo referente ao grupo de tuplas associado a cada valor dos atributos de agrupamento [Elmasri et al, 2011].
	Delete	Remove tuplas de uma relação. Inclui uma cláusula where para selecionar as tuplas a serem excluídas [Elmasri et al, 2011].
	Update	Usado para modificar valores de atributo de uma ou mais tuplas selecionadas [Elmasri et al, 2011].
	Set	Cláusula adicional no comando de atualização (update) que especifica os atributos a serem modificados e seus novos valores [Elmasri et al, 2011].
	Insert	Em sua forma mais simples, é usado para acrescentar uma única tupla a uma relação, tendo que especificar o nome da relação e uma lista de valores para a tupla [Elmasri et al, 2011].
Token	ConectorLogico	Símbolo ou palavra usado para conectar duas ou mais sentenças (tanto na linguagem formal quanto na linguagem natural) de uma maneira gramaticalmente válida, de modo que o sentido da sentença composta produzida dependa apenas das sentenças originais [Wikipedia, 2013].
	OperadorComparacao	Operadores básicos de comparação lógicos para comparar valores de atributo entre si e com constantes literais [Elmasri et al, 2011].
ConectorLogico	AND	Palavra usada para combinar duas expressões lógicas, validando que as duas sejam verdadeiras.

ConectorLogico	OR	Palavra usada para combinar duas expressões lógicas, validando que apenas uma delas seja verdadeira.
OperadorComparacao	Between	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor dentro do intervalo de valores definidos no objeto do seu lado direito na expressão lógica.
	Igual	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor igual ao valor do objeto do seu lado direito na expressão lógica.
	In	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor contido no conjunto de valores especificados no objeto do seu lado direito na expressão lógica.
	Like	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor similar que o valor do objeto do seu lado direito na expressão lógica.
	MaiorIgualQue	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor maior ou igual que o valor do objeto do seu lado direito na expressão lógica.
	MaiorQue	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor maior que o valor do objeto do seu lado direito na expressão lógica.
	MenorIgualQue	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor menor ou igual que o valor do objeto do seu lado direito na expressão lógica.
	MenorQue	Operador lógico básico que compara se o objeto do seu lado esquerdo em uma expressão lógica possui o valor menor que o valor do objeto do seu lado direito na expressão lógica.
	Outro	Qualquer outro operador não definido e que não precise ser explícito para uso em alguma heurística de sintonia fina.
ObjetoSGBD	Coluna	É o nome de um papel desempenhado por algum domínio no esquema de relação [Elmasri et al, 2011].
	Tabela	Coleção de dados relacionados que não possui ordem e permite elementos em duplicata [Date, 2004].

PlanoExecucao	PlanoExecucaoHipotetico	Plano de execução gerado pelo otimizador considerando estruturas hipotéticas, ou seja, que não existam fisicamente na base de dados e que foi gerado a partir da manipulação de um plano de execução real [Monteiro, 2008].
	PlanoExecucaoReal	Plano de execução gerado pelo otimizador através do processamento de uma consulta Q submetida ao SGBD [Monteiro, 2008].
TipoVarredura	FullTableScan	Análise de cada registro de cada página de uma tabela [Shasha et al, 2003].
	FullIndexScan	Caminho de acesso em que as chaves de um índice são lidas antes ou ao invés das páginas de dados da tabela correspondente [Shasha et al, 2003].
EstruturaAcesso	Indice	Organização de dados para agilizar a execução de consultas em tabelas ou coleções orientadas a objetos. Ele consiste de uma estrutura de dados e uma organização da tabela [Shasha et al, 2003].
	VisaoMaterializada	Armazena os dados associados à visão, ou seja, ao resultado da consulta descrita na visão. Considerada informação redundante, mas que pode agilizar consultas substancialmente [Shasha et al, 2003].
	Partição	Divisão de conjunto de dados ou atributos que são acessados unicamente, ou em sua maioria, por conjuntos de aplicativos distintos [özsü et al, 2001].
EstruturaDados	ArvoreB+	Estrutura de árvores balanceada que permite acesso rápido aos dados necessários para uma consulta. Todos os pares chave-ponteiro estão no nível folha [Shasha et al, 2003].
	Hash	Estrutura em árvore cuja raíz é uma função, chamada de hash. Dada uma chave, a função hash retorna uma página que contém ponteiros para os registros que inclui a chave [Shasha et al, 2003].
	Bitmap	Estrutura de índice usada por cláusulas where que possuam muitas condições não seletivas. Os bitmaps trabalham bem nessas situações porque realizar o cruzamento de vetores de bits é uma operação extremamente rápida [Shasha et al, 2003].

Índice	ÍndiceHipotetico	Índice que existe apenas na metabase do SGBD, ou seja, não existe fisicamente. Sendo assim, não ocupa espaço em disco e nem pode ser utilizada no acesso aos dados [Monteiro, 2008].
	ÍndiceReal	Índice que existe fisicamente no SGBD, ou seja, ocupa espaço em disco e pode, efetivamente, ser usado no acesso aos dados [Monteiro, 2008].
ComandoDDL	Create	Usado para especificar uma nova relação, dando-lhe um nome e especificando seus atributos e restrições iniciais [Elmasri et al, 2011].
	Drop	Usado para remover elementos nomeados do esquema, como tabelas, domínios ou restrições. Também pode ser usado para remover um esquema [Elmasri et al, 2011].
	Reindex	Usado para reconstruir estruturas de índices fisicamente no banco de dados.
	CreateHypothetical	Usado para especificar um novo índice hipotético na base de dados, dando-lhe um nome e especificando seus atributos [Costa et al, 2005].
	DropHypothetical	Usado para remover índices hipotéticos da base de dados [Costa et al, 2005].
<b>Tarefa de execução de heurísticas de sintonia fina de banco de dados</b>		
Fonte	Regra	Qualquer expressão lógica definida em alguma linguagem padrão que seja processável por uma máquina de regras.
	Funcao	Qualquer função contendo um conjunto de instruções descritas em alguma linguagem de programação padronizada e que seja processável por uma máquina.

### Propriedades das classes:

Classe	Propriedade	Definição
<b>Domínio de sintonia fina de banco de dados</b>		
SGBD	nome	Nome do SGBD.
	Custo Entrada Saída	O custo de entrada e saída (entrada do disco para a memória principal e saída da memória para o disco). Custo que domina o custo de operações típicas de banco de dados [Ramakrishnan et al, 2003].
	Custo_CPU	Custo para processar uma tupla que esteja em memória [Shasha et al, 2003].
	Coeficiente Custo_ I_O_ CPU	Coeficiente que relaciona, percentualmente, o custo de uma operação de E/S com o custo de CPU para processar uma tupla que esteja em memória [Salles, 2004].

SGBD	tamanhoDiscoSF	Tamanho em disco, expresso em kylobytes, destinado a sintonia fina. Esse tamanho deve ser usado para criar estruturas de acesso.
ComandoDML Cláusula	Descricao	Descrição do comando DML, exatamente como ele é submetido ao banco de dados, ou descrição da cláusula, onde o comando DML é particionado em substrings que iniciam com uma cláusula do padrão SQL.
ComandoDML	Probabilidade Execucao	Probabilidade de execução de um comando DML. Definição, em porcentagem, da chance que o comando tem de ser executado no banco de dados, caso o DBA já possua tal conhecimento.
Expressao Predicado	Objeto Expressao	Objeto presente do lado esquerdo do operador (token) de uma expressão lógica, podendo ser uma coluna ou uma expressão lógica (expressaoPredicado).
	Token	Conector lógico ou operador de comparação presente em uma expressão de predicado.
	Valor Predicado	Objeto presente do lado direito do operador (token) de uma expressão lógica, podendo ser uma coluna ou uma expressão lógica (expressaoPredicado).
	Valor PredicadoLiteral	Valor de predicado que não seja uma coluna e nem uma expressão lógica, ou seja, valor do tipo literal (ex.: integer, string).
<i>Token</i>	Símbolo	Símbolo que corresponde ao conector lógico ou operador de comparação presente em uma expressão de predicado, segundo o padrão SQL.
Objeto SGBD	Nome	Nome atribuído à instância do objeto presente no banco de dados.
Coluna	Tipo	Tipo de dados que a coluna suporta. Ex.: integer, string.
	Ordem	Ordem da posição da coluna na tabela na base de dados.
Tabela	Numero Tuplas	Número de tuplas da tabela.
	Numero Paginas	Número de páginas de dados ocupadas pela tabela.
Plano Execucao	Numero Tuplas Atualizadas	Número de tuplas atualizadas por um comando de atualização [Salles, 2004].
	Valor Custo Beneficio	Diferença entre o custo da consulta com a configuração de estruturas de acesso reais e o seu custo usando uma configuração de estruturas de acesso reais e hipotéticas [Salles, 2004].
Plano Execucao Hipotetico	Custo Total PlanoExecucao Hipotetico	Custo total estimado para a execução de um comando DML, considerando estruturas reais e hipotéticas.
Plano Execucao Real	Custo Total PlanoExecucao Real	Custo total estimado para a execução de um comando DML, considerando apenas estruturas reais.

Índice	Nome	Nome do índice presente no SGBD.
Índice	Valor Custo Atualizacao	Custo estimado de atualização de um índice durante o processamento de um comando de atualização [Salles, 2004].
	Valor Custo Estimado Criacao	Custo estimado de criação de um índice. [Salles, 2004]
	numeroBlocos	Quantidade de blocos em disco que o índice ocupa [Morelli, 2006].
	DataHora Atualizacao	Data e hora de atualização dos valores de custos do índice para acompanhamento do mesmo.
	OrdemColunas	Ordem da posição das colunas indexadas no índice.
	Índice Hipotetico	Valor Custo Beneficio Acumulado
Quantidade Utilizacao		Quantidade de vezes que um índice hipotético foi útil, ou seja, foi escolhido pelo otimizador na geração de plano de execução hipotético para um comando DML submetido ao banco de dados [Morelli, 2006].
Índice Real	Valor Bonus	Valor resultante da divisão do benefício acumulado, ou seja, igual ou um pouco superior ao custo de criação, pela quantidade de vezes em que o índice, enquanto hipotético, foi usado [Morelli, 2006].
	valorBonus Acumulado	Acúmulo do valor de bônus do índice real a cada vez que o mesmo é utilizado em algum plano de execução gerado para um comando DML submetido ao banco de dados [Morelli, 2006].
	Valor RazaoInicial	Razão entre o tamanho de tuplas de uma tabela e o tamanho de blocos de um índice no momento em que um índice acabou de ser criado [Morelli, 2006].
	Valor GrauFragmentacao	Valor estimado do grau de fragmentação de um índice real, ou seja, a porcentagem do quanto um índice se encontra com suas páginas de dados fragmentadas. Valor derivado da fórmula proposta por [Morelli, 2006] $(100 - [(ValorRazaoInicial/ValorRazaoAtual) * 100])$ . O valorRazaoAtual pode ser considerada a razão entre o tamanho de tuplas de uma tabela e o tamanho de blocos de um índice no momento em que um índice é usado ou impactado por comandos de atualizações .
	Numero VarredurasUtil	Quantidade de vezes que um índice real foi útil para consultas que executaram varreduras [Morelli, 2006].
	valorFator Preenchimento	Percentual de preenchimento das páginas do nível folha que podem ser ocupadas por um índice real [Morelli, 2006].

<b>Tarefa de execução de heurísticas de sintonia fina de banco de dados</b>		
Heurística	Nome Heurística	Nome da heurística de sintonia fina.
	Classificacao	Classificação atribuída a heurística de acordo com as ações que ela pode tomar. O objetivo dessa propriedade é agrupar as heurísticas de acordo com as atividades de sintonia fina realizadas. Por exemplo: manutenção de índices, manutenção de visões materializadas.
Conceito Fonte	Nome	Nome do conceito presente na ontologia de domínio, que é pré-condição da heurística de sintonia fina ou nome da função ou regra que deve ser chamada para que o conceito possa ser instanciado ou a ação realizada.
Conceito	Ordem Instanciacao	Número que corresponde a ordenação de instanciação dos conceitos, visto que existem conceitos dependentes de outros.
Funcao	Classe	Nome da classe responsável por implementar a função que precisa ser executada para obter a instância de um conceito. Esse termo é aplicado para a implementação do framework em uma linguagem de programação orientada a objetos, tal como java.
Regra	Descrição Regra	Descrição exata da expressão da regra que deve ser submetida à máquina de regras.
Parametro	Conceito	Nome da classe que possui a propriedade que vai servir como entrada para uma função ou que vai receber a instância retornada como saída de uma função.
	Datatype	Nome da propriedade que vai servir como entrada para uma função ou que vai receber a instância retornada como saída de uma função.
	Tipo	Tipo de dados da propriedade que vai servir como entrada para uma função ou que vai receber a instância retornada como saída de uma função. Ex.: integer, string.
Acao	Descricao	Descrição do comando DDL, exatamente como ele é submetido ao banco de dados.
	DecisaoDBA	Indicação de aceitação ou recusa do DBA em relação a ação sugerida pela heurística de sintonia fina.
	Justificativa	Texto explicativo do DBA para a recusa de uma ação sugerida por uma heurística de sintonia fina.

**Facetas das propriedades:**

Propriedade	Tipo de valor	Cardinalidade
<b>Domínio de sintonia fina de banco de dados</b>		
CoeficienteCusto_I_O_CPU	Float	1
Custo_CPU	Float	1
CustoEntradaSaída	Float	1
CustoTotalPlanoExecucaoHipotetico	Float	1
CustoTotalPlanoExecucaoReal	Float	1
DataHoraAtualizacao	Date	1
Descricao	String	1
Nome	String	1
numeroBlocos	Int	1
NumeroPaginas	Int	1
NumeroTuplas	Int	1
NumeroTuplasAtualizadas	Int	1
NumeroVarredurasUtil	Int	1
ObjetoExpressao	ExpressaoPredicado Coluna	N
ProbabilidadeExecucao	Float	1
QuantidadeUtilizacao	Int	1
Tipo	String	1
Token	String	N
ValorBonus	Float	1
valorBonusAcumulado	Float	1
ValorCustoAtualizacao	Float	1
ValorCustoBeneficio	Float	1
ValorCustoBeneficioAcumulado	Float	1
ValorCustoEstimadoCriacao	Float	1
valorFatorPreenchimento	Float	1
ValorGrauFragmentacao	Float	1
ValorPredicado	ExpressaoPredicado Coluna	N
ValorPredicadoLiteral	String	N
Símbolo	String	1
ValorRazaoInicial	Float	1
<b>Tarefa de execução de heurísticas de sintonia fina de banco de dados</b>		
Classe	String	1
Conceito	String	1
Datatype	String	1
DecisaoDBA	Boolean (Aceito ou não)	1
Descricao	String	1
DescriçãoRegra	String	1
Justificativa	String	1
Nome	String	1
NomeHeuristica	String	1
OrdemInstanciacao	Int	1
Tipo	String	1

**Instâncias pré-definidas:**

Classe	Subclasse	Subclasse	Propriedade	Valor da instância
<b>Domínio de sintonia fina de banco de dados</b>				
Token	Conector Logico	AND	Simbolo	AND
		OR	Simbolo	OR
	Operador Comparacao	Between	Simbolo	between
		Igual	Simbolo	=
		In	Simbolo	in
		Like	Simbolo	like
		MaiorIgualQue	Simbolo	>=
		MaiorQue	Simbolo	>
		MenorIgualQue	Simbolo	<=
MenorQue	Simbolo	<		

Anexo II

