

8 Referências Bibliográficas

[Adl03] ADL - Advanced Distributed Learning, *About ADL*. Acessado em 2003 através do endereço: <http://www.adlnet.org/index.cfm?fuseaction=abtadl>

[Adl03a] ADL - Advanced Distributed Learning, *About SCORM*. Acessado em 2003 através do endereço: <http://www.adlnet.org/index.cfm?fuseaction=scormabt>

[Aic88] AICC – Aviation Industry CBT Committee. Acessado em 2003 através do endereço: <http://www.aicc.org/>

[Ala99] Maryam Alavi, *Knowledge Management Systems: Issues, Challenges, and Benefits*, Communications of the Association for Information Systems, Vol. 1, Art #7, Feb 1999. Acessado em 2003 através do endereço: <http://portal.acm.org>

[Alg02] ALGORA *Normes et démarches de qualité. Etat des lieux des études et projets*. Acessado em 2002 através do endereço: <http://ressources.algora.org/reperes/economie/tel>

[AM97] Agogino, A. M., & Muramatsu, B., *The Synthesis Coalition: The National Engineering Education Delivery System (NEEDS): A multimedia digital library of courseware*. International Journal of Engineering Education 13(5), pp. 333-340, 1997. Acessado em 2003 através do endereço: <http://www.smete.org>

[Ari97] ARIADNE – Foundation for the European Knowledge Pool. Acessado em 2003 através do endereço: <http://www.ariadne-eu.org/fr/about/general/history/>

[BBS01] Fouzia Benchikha, Mahmoud Bofaida e Lionel Seinturier. *Integration of the viewpoint mechanism in federated databases* – SAC 2001 – ACM. Acessado em 2002 através do endereço: <http://portal.acm.org>

[Bar01] Álvaro C. P. Barbosa, *Middleware para Integração de Dados Heterogêneos Baseado em Composição de Frameworks* – PUC-Rio – Tese de Doutorado, 2001

[Ber99] Henrike Berthold, *A Federated Multimedia Database System*, ACM Multimedia 99, 10/99 Orlando, FL, USA. Acessado através do endereço:
<http://portal.acm.org>

[BFA01] MICHAEL Bernnan, Susan Funke and Cushing Anderson, *The Learning Content Management System – A New eLearning Market Segment Emerges* – IDC White Paper, May 2001. Acessado em 2002 através do endereço:
<http://www.lcmscouncil.org>

[BH03] Brandon-Hall. Acessado em 2003 através do endereço:
http://www.brandonhall.com/public/resources/lms_lcms/

[BL00] Chuck Barritt and Deborah Lewis, *Reusable Learning Object Strategy, Definition, Creation Process, and Guidelines for Building*, Version 3.1, April 22, 2000, Cisco Systems, Inc. Acessado em 2002 através do endereço:
<http://www.cisco.com>

[BR01] Philip A. Bernstein e Erhard Rahm. *On Matching Schemas Automatically* – Microsoft Research – Technical Report – MSR-TR-2001-17, 2001. Acessado em 2002 através do endereço <http://www.cs.unibo.it>

[Bro02] Brockbank, Bray, *Demystifying e-Learning Portals, e-Learning Advisor Zone*. Acessado em 2003 através do endereço:
<http://www.advisor.com/Articles.nsf/aid/BROCB001>

[Can02] Canadian School Net. Acessado em 2002 através do endereço:
<http://www.schoolnet.ca/>

[Car02] CARLENTON, *e-Learning Abbreviations*. Acessado em 2003 através do endereço: <http://www.carleton.ca>

[CLD99] S. Chakravarty; R. Le; P. Dasari. *ECA Rule Processing in Distributed and Heterogeneous Environments*. In Proceedings of the 14th. International Conference on Data Engineering (ICDE), USA, 1999. Acessado em 2002 através do endereço: <http://www.acm.org>

[Ctl02] Computers in Teaching and Learning. Acessado em 2003 através do endereço: <http://www.staffs.ac.uk/cital/welcomeframe.html>

[DAg01] Andy Dong, Alice M Agogino, *Design Principles for the Information Architecture of a SMET Education Digital Library*, JCDL 01, ACM 2001, http://beta.smete.org/about_smete/publications/JCDL-0601

[DCM95] Dublin Core Metadata Initiative. Acessado em 2003 através do endereço: <http://dublincore.org/>

[DG00] Delphi Group, *Need to Know Integrating e-Learning with High Velocity Value Chains*, acessado em 2003 através endereço: <http://www.delphigroup.com>

[Don02] Jill Funderburg Donello, *Theory and Practice: Learning Content Management Systems*, LeadingWay Knowledge Systems Newsletter, July 19, 2002, acessado através do endereço: <http://www.leadingway.com>

[Fra98] Carl Frappaolo, *Ushering in The Knowledge-Based Economy*, The Delphi Group Symposium, 1998, acessado em 2003 através do endereço: <http://www.delphigroup.com>

[Geo01] GeoLearning, *The e-Learning Workbook*, acessado em 2003 através do endereço: <http://geolearning.com>

[Gil01] Kym Gilhooly, *Making e-learning effective*, Computerworld, 7/16/01

[GW96] Paul Grefen e Jennifer Widom. *Integrity Constraint Checking in Federated Databases* – 1996, acessado em 2002 através do endereço:

<http://dbpubs.stanford.edu:8090/pub/1996-48>

[Haw00] Hawryszkiewicz, Igor, *Creating and Supporting Learning Environment*, ACM 2000, acessado em 2002 através do endereço: <http://portal.acm.org>

[Hod02a] Wayne Hodgins – *Learning Technology Standards: The Promise of Personalization – on-line Learning 2002*, Sept. 2002 – Anaheim, CA USA. Acessado em 2003 através do endereço:

<http://www.onlinelearningconference.com/2002/>

[Hod02b] Wayne Hodgins – *The Future of Learning Objects* – A United Engineering Foundation Conference – 11-16 August 2002 – Davos Switzerland, acessado em 2003 através do endereço: <http://www.coe.gatech.edu/e-TEE>

[Ibm00a] IBM *DB2 Universal Database, XML Extender Administration and Programming*, Version 7. Acessado em 2002 através do endereço:

<ftp://ftp.software.ibm.com/ps/products/db2/info/db2sxe70.pdf>

[Ibm00b] IBM *Net.Data for OS/2, Windows NT and UNIX: Administration and Programming Guide*, Version 7. Acessado em 2003 através do endereço:

<http://www-3.ibm.com/software/data/net.data/docs/pdf/wrk/db2an.pdf>

[Ibm98a] IBM *DB2 Universal Database, Text Extender Administration and Programming*, Version 5 Release 2. Acessado em 2003 através do endereço:

<ftp://ftp.software.ibm.com/ps/products/db2/info/desu9mst.pdf>

[Ibm98b] IBM *DB2 Universal Database, Image, Audio and Video Extender Administration and Programming*, Version 5 Release 2. Acessado em 2003 através do endereço:

<ftp://ftp.software.ibm.com/ps/products/db2/info/dmba5mst.pdf>

[Ibm98c] IBM *DB2 Universal Database, Federated Systems Guide*, Version 7.

Acessado em 2002 através do endereço:

<ftp://ftp.software.ibm.com/ps/products/db2/info/db2fp.pdf>

[Ibm03a] IBM *DB2 Universal Database, Federated Systems Guide*, Version 8.

Acessado em 2003 através do endereço:

<ftp://ftp.software.ibm.com/ps/products/db2/info/vr8/pdf/letter/db2fpe80.pdf>

[Ibm03b] IBM *DB2 Information Integrator, Data Source Configuration Guide*

Version 8. Acessado em 2002 através do endereço:

ftp://ftp.software.ibm.com/software/db2ii/info/v81/en_US/DataSrcConfigGde.pdf

[Ims01] IMS Global Learning Consortium, Inc. Acessado em 2002 através do endereço: <http://www.imsglobal.org/>

[Ims00] IMS *Meta-data FAQ*. Acessado em maio/2003 através do endereço:

http://www.imsproject.org/metadata/metadafaq.html#_Toc492376214

[Ims01a] IMS *Learning Resource Meta-Data Best Practice and Implementation Guide* – Version 1.2.1 Final Specification – 28 September 2001. Acessado em 2002 através do endereço: <http://www.imsglobal.org/specificationdownload.cfm>

[Ims01b] IMS *Learning Resource Meta-Data XML Binding* – Version 1.2.1 Final Specification – 28 September 2001. Acessado em 2002 através do endereço:

<http://www.imsglobal.org/specificationdownload.cfm>

[Ims01c] IMS *Learning Resource Meta-data Information Model* – Version 1.2 .1 Final Specification – 28 September 2001. Acessado em 2002 através do endereço:

<http://www.imsglobal.org/specificationdownload.cfm>

[Iee02] IEEE, *Draft Standard for Learning Object Metadata* – IEEE 1484.12.1-2002 – 15 July 2002. Acessado em 2003 através do endereço:

http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf

[Kha02] Khan, B, *A Framework for e-Learning*, acessado através do endereço: <http://www.bookstoread.com/framework> no arquivo: eLearningFramework.ppt

[Kos01] Donald Kossmann. The State of the Art in Distributed Query Processing – ACM 2000 acessado através do endereço: <http://portal.acm.org>

[Kru02] Kevin Kruse, *The State of e-Learning: Looking at History with the Technology Hype Cycle*. Acessado em 2003 através do endereço: www.e-learningguru.com/articles/hype1_1.htm

[Law03] Susan Lawson, *Using LOBs and Extenders in DB2*. Acessado em 2003 através do endereço: http://searchdatabase.techtarget.com/tip/1,289483,sid13_gci896492,00.html

[Lcm03] LCMS Council. Acessado em 2003 através do endereço: <http://www.lcmscouncil.org/index.html>

[Lea03] Learnativity.com – *Learning Standards Acronyms*. Acessado em 2003 através do endereço: <http://www.learnativity.com>

[Lom00] LOM WORKING DRAFT V4.1 [On-line]. Acessado em 2003 através do endereço: <http://ltsc.ieee.org>

[Lon00] Warren Longmire, *Content and Context: Designing and Developing Learning Objects, Learning without Limits – Vol. III – Emerging Strategies for Effective e-Learning Solutions*, Informania, Inc., 2000. Acessado em 2003 através do endereço: <http://www.learnativity.com>

[Lts96] Learning Technology Standards Committee. Acessado em 2003 através do endereço: <http://ltsc.ieee.org/>

[Mel02] R. N. Melo et al., *I PGL Database Research Workshop Report – PUC–Rio, 2002*

[MB03] R. N. Melo e L.B. Baruque, *A Database Approach to Partnership in Global Learning*, PGL Database Conference 2003 (PGLDB'2003) – PUC-Rio - 2003

[MHM01] Jeanette Muzio, Tanya Heins and Roger Mundell, *Experiences with Reusable e-Learning Objects: From Theory to Practice*, Center for Economic Development and Applied Research (CEDAR), Royal Roads University, Victoria, BC, Canada, June 6, 2001. Acessado em 2003 através do endereço:

<http://www.cedarlearning.com>

[MS02] Mangan, P. & Sadiq, S., *On Building Workflow Models for Flexible Processes*, 13th Australasian Database Conference (ADC2002), Melbourne, Australia, 2002. Acessado através do endereço: <http://adc.massey.ac.nz>

[Mur00] Brandon Muramatsu, *The Development of a National Science, Mathematics, Engineering and Technology Education Digital Library Lessons Learned from NEEDS*, Proceedings of the 2000 International Conference for Engineering Education, Aug 13-17, 2000, Taipei, Taiwan. Acessado em 2003 através do endereço: <http://www.smete.org>

[NH02] Norin, M. & Himes, F., *Building Effective Higher Education Partnerships With Information Technology Corporations*, MACROMEDIA, April 2002. Acessado em 2003 através do endereço: <http://www.macromedia.com>

[Nich01] Maish Nichani, *LCMS = LMS + CMS [RLOs]*, eLearning Post, May 02, 2001. Acessado em 2003 através do endereço: <http://www.elearningpost.com>

[Özsu96] M. Tamer Özsu, *Future of Database Systems: Changing Applications and Technological Developments*, ACM Dec 1996. Acessado em 2002 através do endereço: <http://portal.acm.org>

[ÖV99] M. Tamer Özsu e Patrick Valduriez, P., *Principles of Distributed Database Systems*, Prentice Hall, 1999, 2nd edition

[Pat98] N.W. Paton. *Active Rules for Databases*. Springer Verlag, 1998 acessado em 2002 pelo endereço: <http://portal.acm.org>

[PBE95] Evaggelia Pitoura, Omran Bukhres, and Ahmed Elmagarmid, *Object Orientation in Multidatabase Systems*, ACM Computing Surveys, Vol. 27, No 2, June 1995, acessado em 2002 pelo endereço: <http://portal.acm.org>

[Per+00] Sharon Perry et al, *The CO3 Project – Stretching the IMS Specifications to Achieve Interoperability*, Mach 2002, University of Wales Bangor, acessado em 2003 pelo endereço: <http://www.staffs.ac.uk/COSE>

[PMA00] Puzniak, J., McMartin, F. and Agogino A., *Building a digital learning community for faculty on the Internet*, Proceedings of the 2000 American Society for Engineering Education Annual Conference: June 18-21, 2000, St. Louis, MO. Acessado em 2003 através do endereço: <http://www.smete.org>

[PPM03] Luiz Antônio M. Pereira, Fábio André M. Porto, Rubens Nascimento Melo – *Objetos de Aprendizado Reutilizáveis (RLOs): Conceitos, Padronização, Uso e Armazenamento* – Monografias em Ciência da Computação – PUC-Rio, 2003

[Pur02] Purnell, Micks, *Fundamentals of IBM DB2 Federated Server and Relational Connect*, IBM Data Management Advanced Technical Support, IBM Silicon Valley Lab, June 2002, acessado em 2003 pelo endereço: <http://www7b.software.ibm.com>

[RHA03] Randal I House Associates, Inc, *SCORM – Sharable Courseware Object Reference Model - Overview*, acessado em maio 2003 no endereço: <http://www.rhassociates.com>

[Reg02] Regan, B., *Building Accessible Dynamic Websites with Macromedia ColdFusion MX*, MACROMEDIA White Paper, May 2002, acessado através do endereço: <http://macromedia.com>

[Saba01] SABA White Paper, *Next Generation e-Learning and the Road to Human Capital Development and Management: Successes, Challenges and Lessons Learned*, acessado em 2003 pelo endereço: <http://www.saba.com>

[SAL+96] Michael Stonebraker, Paul M. Aoki, Witold Litwin, Avi Pfeffer, Adam Sah, Jeff Sidell, Carl Staelin e Andrew Yu. *Mariposa: a wide-area distributed database system*. VLDB Journal – 1996, acessado em 2002 através do endereço: <http://db.cs.berkeley.edu/papers>

[Sar02] C.M. Saracco, *Coping with Disparate Data in WEB Applications, DBMS and Web Application Server Integration*, IBM Silicon Valley Lab, August 2002, acessado em 2003 pelo endereço: <http://www7b.software.ibm.com>

[SH01] Michael Stonebraker e Joseph M. Hellerstein. *Content Integration for E_Business* – ACM SIGMOD 2001 May 21-24, Santa Barbara, California USA. Acessado em 2002 através do endereço: <http://www.portal.acm.org>

[Siv96] Douglas Siviter, *Computer Assistance for Managing Educational Resources and Managing Collaborative Educational Processes*. In Proc. ACM SIGCSE/SIGCUE Conf., Barcelona, Spain, June 1996. Acessado em 2003 através do endereço: <http://portal.acm.org>

[SL90] Amit P. Sheth e James A. Larson, *Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases*, ACM Computing Surveys, Vol. 22, No 3, September 1990. Acessado em 2002 através do endereço: <http://portal.acm.org>

[Spi00] Israel Spiegler, *Knowledge Management: A New Idea or A Recycled Concept?* Communications of the Association for Information Systems, Vol. 3, Art 14, Jun 2000. Acessado em 2003 através do endereço: <http://portal.acm.org>

[Stee02] The Managed Learning Environments Steering Group Technical Report – A Report on the Interoperability Pilot program – March 2002. Acessado em 2003 através do endereço: <http://www.jisc.ac.uk>

[Tal02] John Talanca, *Let's Get Small*, Instructional Design. Acessado em 2003 através do endereço: <http://www.e-learningguru.com>

[Tho02] Denis Lamontagne, *214 plates-formes e-formation, plates-formes e-learning*, Thot/Cursus Jul. 18 2002. Acessado em 2003 através do endereço: <http://thot.cursus.edu>

[TS01] Thomas, D. & Steidley, C., *Toward a Comprehensive, Knowledge-Based Software Design Course*, JCSC 16, 3 (March 2001). Acessado em 2003 através do endereço: <http://portal.acm.org>

[Tür99] Can Türker. *Semantic Integrity Constraints in Federated Database Schemata - Volume 63 - Dissertations in Database and Information Systems-Infix* – 1999. Acessado em 2002 através do endereço: <http://portal.acm.org>

[ULM98] Elvira M. A. Uchôa, Sérgio Lifschitz, Rubens N. Melo, *HEROS: A Heterogeneous Object Oriented Database System*, DEXA, 1998 pp. 435-447

[VCR00] Genoveva Vargas-Solar, Christine Collet, Helena G. Ribeiro. *Active Services for Federated Databases* – ACM 2000. Acessado em 2002 através do endereço: <http://portal.acm.org>

[VS97] Sabrina De Capitani di Vimercati e Pierangela Samrati. *Access Control in Federated Systems* – ACM 1997. Acessado em 2002 através do endereço: <http://portal.acm.org>

[Wag00] Ellen D Wagner, *e-Learning: Where Cognitive Strategies, Knowledge Management, and Information Technology Converge, Learning without Limits* – Vol. III – Emerging Strategies for Effective e-Learning Solutions – Informania, Inc., 2000. Acessado em 2002 pelo endereço: <http://www.learnativity.com>

[Wag02] Ellen D. Wagner, *The New Frontier of Learning Object Design*, The e-Learning Developers' Journal, Jun 18 2002. Acessado em 2003 pelo endereço: www.eLearningGuild.com

[Wil01] David A. Wiley, II, *Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy*, Utah State University. Acessado em 2003 através do endereço: <http://www.reusability.org>

[WC02] Walczak, Krzysztof & Cellary, Wojciech, *Building Database Applications of Virtual Reality with X-VRML*, ACM 2002. Acessado em 2003 através do endereço: <http://portal.acm.org>

[WGR01] Wiley, Gibbons, Recker, *A reformulation of the issue of learning object granularity and its implications for the design of learning objects*. Acessado em 2003 através do endereço: <http://www.reusability.org>

[W3C01]] W3C, *XML Schema Part 1: Structures – W3C Recommendation* 2001. Acessado em 2003 através do endereço: <http://www.w3.org>

.

9 Glossário

9.1. Glossário de Termos

Neste tópico estão listados alguns termos do idioma inglês e a correspondente tradução empregada na dissertação.

Binding	Ligação, conexão
Botton-up	De baixo para cima
Browser	Navegador
e-Learning	Aprendizado eletrônico
Extender	Extensor
Foreign key	Chave estrangeira
Learning object	Objeto de aprendizado
Loosely coupled	Acoplamento fraco
Node	Nó, nodo
Peer to peer	Ponto a ponto
Primary key	Chave primária
Site	Local, endereço eletrônico, sítio
Reusable learning object	Objeto de aprendizado reutilizável
Tag	Rótulo
Tightly coupled	Acoplamento forte
Top-down	De cima para baixo
Wrapper	Invólucro

9.2. Glossário de Siglas

Neste tópico estão concentradas as siglas utilizadas no contexto do aprendizado eletrônico [Lea03, Lon00] e na modelagem dos objetos de

aprendizado. Algumas são explicadas devido ao papel relevante, e em determinados casos é indicado o endereço eletrônico para melhores referências.

- ADL** Advanced Distributed Learning - <http://www.adlnet.org/>
É uma iniciativa do Departamento de Defesa dos Estados Unidos. Desde 1997 o Departamento de Defesa, conjuntamente com o White House Office of Science and Technology Policy, vem promovendo colaborações entre o governo, as instituições de ensino e a indústria como uma forma de acelerar a efetiva utilização do aprendizado em linha.
- AICC** Aviation Industry CBT Committee - <http://www.aicc.org/>
É um fórum aberto aos profissionais de treinamento. Este fórum desenvolve diretrizes para a interoperabilidade aplicada à tecnologia de aprendizado. Nos últimos cinco anos a sua especificação, AICC/CMI - Computer Managed Instruction, tornou-se a mais utilizada para elaborar produtos e aplicações operacionais. Este documento define o acompanhamento a ser produzido por sistemas e lições interativas e o formato para intercâmbio de cursos entre sistemas produzidos por diferentes fabricantes
- ALIC** Advanced Learning Infrastructure Consortium -
<http://www.alic.gr.jp/>
- ALO** Atomic Learning Object
- ALS** Adaptive Learning Systems Project of the Advanced Technology Program
- ANSI/SPARC** American National Standards Institute/Standards Planning and Requirements Committee
- ARIADNE** Alliance of Remote Instructional Authoring and Distribution Networks for Europe - <http://www.ariadne-eu.org>
É um projeto pertencente ao programa de pesquisa e desenvolvimento da Comunidade Européia. Desde 1997 ARIADNE tem atuado junto com o IMS LTSC no sentido de atingir o quanto antes um conjunto de metadados educacionais.

ASTD	American Society for Training & Development – Linking People, Learning and Performance – http://www.astd.org/
ATP	National Institute of Standards and Technology - NIST Advanced Technology Project - http://www.atp.nist.gov/
CAM	Content Aggregation Model
CANARIE	Canada’s advanced internet development organization - http://www.canarie.ca/
CanCore	Canadian Core Learning Resource Metadata Protocol - http://www.cancore.ca/
CAREO	Campus Alberta Repository of Educational Objects - http://www.careo.org/
CBT	Computer Based Training
CedMA	Computer Education Management Association – http://www.cedma.org/ É um fórum reunindo os gerentes de ensino das companhias que produzem equipamentos e produtos computacionais. Seus objetivos são acelerar a adoção dos padrões ligados à tecnologia de aprendizado pelos fabricantes.
CEN	Comité Européen de Normalisation - http://www.cenorm.be
CETIS	Center for Educational Technology Interoperability Standards - http://cetis.ac.uk
CIS	Corporate Information Systems
CLEO	Customized Learning Experiences Online - http://www.cleolab.org Reune empresas ligadas ao <i>e-learning</i> , pesquisadores e ADL. Foi constituído inicialmente por CISCO, Click2Learn, IBM, Microsoft e NETg e tem como foco aprimorar o SCORM. Entre pesquisadores acadêmicos reúne Carnegie Mellon University e Open University-UK e tem como gestora das atividades do dia a dia IEEE/LSTC
CMI	Computer Managed Instruction
D-Lib	Digital Library Project
DAD	Document Access Definition

DCMI	Dublin Core Metadata Initiative - http://dublincore.org É um fórum aberto para o desenvolvimento de padrões de metadados interoperáveis em linha e que atendam a uma variedade de propósitos e modelos de negócios. Desenvolve também vocabulários especializados para metadados para descrever recursos visando formar sistemas inteligentes de busca de informações.
DNER	Distributed National Electronic Resource
DoD	US Department of Defense
DTD	Document Type Definition
EdNa	Educational Network Australia - http://www.edna.edu.au/index.html
EML	Educational Modeling Language - http://eml.ou.nl
EOE	Educational Object Economy - http://www.apple.com/education/LTReview/winter98/book.html
EPSS	Electronic Performance Support System
ERP	Enterprise Resource Planning
GEM	Gateway to Educational Materials Project - http://www.thegateway.org
GSCO	Guidelines for Courseware Sharable Objects
GUI	Graphical User Interface
HTML	Hypertext Markup Language - http://www.w3c.org/MarkUp
IEC	International Engineering Consortium - http://www.iec.org
IEEE	Institute of Electrical and Electronics Engineers - http://www.ieee.org
ILT	Information Learning Technology

IMS	<p>Instructional Management System Project - http://www.imsproject.org</p> <p>Foi criado em 1997 pelo US Educom no âmbito da <i>National Learning Infrastructure Initiative (NLII)</i>.</p> <p>Atualmente faz parte do IMS Global Learning Consortium, Inc http://www.imsglobal.org</p> <p>É um consórcio formado por instituições de ensino superior e por produtores de tecnologia. Este consórcio trabalha no desenvolvimento de uma arquitetura que viabilize o aprendizado em linha. No foco desta arquitetura estão interfaces independentes de plataforma para metadados, conteúdo agregado, serviços de gerenciamento, perfis de usuário, e serviços externos como banco de dados. O consórcio IMS busca antecipar a ampla utilização da tecnologia emergente como, por exemplo, XML e prover assim uma excelente visão futura do aprendizado em linha.</p>
ISO	International Standards Organization - http://iso.org
ISSS	Information Society Standardization System - http://www.cenorm.be/iss/workshop/lt
IST	Information Society Technologies
JDOM	Java Data Object Model
LIP	Learner Information Package
LCMS	Learning Content Management System
LCO	Learning Content Object
LF	<p>Learning Federation - http://www.learningfederation.org</p> <p>Reúne pesquisadores públicos e privados buscando a pesquisa e o desenvolvimento de novos ambientes de aprendizado para incrementar o ensino e o treinamento.</p>
LML	Learning Markup Language
LMS	Learning Management System
LO	Learning Object
LOM	Learning Object Metadata - http://ltsc.ieee.org/wg12/index.html

LOM	Atributo utilizado na modelagem para descrição em XML do objeto de aprendizado segundo o modelo LOM
LRN	Learning Resource Interchange - http://www.microsoft.com/elearn
LTSC	Learning Technology Steering Committee of the IEEE - http://ltsc.ieee.org Tem a função de desenvolver padrões reais para tecnologia de aprendizado. Consórcios como IMS e AICC tem reconhecido o IEEE/LTSC como o único fórum para transformar especificações em padrões e encaminhado para estas suas iniciativas como, por exemplo, a especificação CMI e a especificação de metadados.
LTSN	Learning and Teaching Support Network
MERLOT	Multimedia Educational Resource for Learning and Online Teaching - http://www.merlot.org/Home.po
MLE	Managed Learning Environment
ODL	Open Distance Learning
ORD	Atributo representando o número de ordem na Sequência
QTI	Question and Test Interoperability
PDA	Personal Development Plan
POOL	Portal for Online Objects in Learning - http://www.canarie.ca/funding/learning/1999backgrounders/pool.html
PROMETEUS	Promoting Multimedia Access to Education and Training in European Society - http://prometeus.org
QA	Quality Assurance
RDF	Resource Description Format - http://www.w3c.org/RDF
RIO	Reusable Information Object
RLO	Reusable Learning Object
SAML	Security Assertion Mark-up Language
SCO	Sharable Content Object
SCORM	Sharable Courseware Object Reference Model - http://www.adlnet.org
SGBD	Sistema de gerência de banco de dados

SGBDF	Sistema de gerência de banco de dados federado
SGML	Structured Markup Language
SOAP	Simple Object Access Protocol
UDF	User Defined Function – função definida pelo usuário para utilização em um gerenciador de banco de dados
UDT	User defined type – tipo de dados definido pelo usuário para utilização em um gerenciador de banco de dados
ULF	Universal Learning Format
URI	Universal Resource Identifier - http://www.w3c.org/Addressing
URL	Universal Resource Locator - http://www.w3c.org/Addressing
VLE	Virtual Learning Environment
W3C	World Wide Web Consortium - http://www.w3c.org É um consórcio que traça os rumos gerais da Web. Apesar de não estar dedicado ao aprendizado eletrônico, este consórcio define a tecnologia básica que será utilizada pelas especificações voltadas para ensino.
WBT	Web Based Training
WBL	Work Based Learning
XML	EXtensible Markup Language - http://www.w3.org/XML

10 Apêndices

10.1. Listagem do Arquivo DTD

Como apresentado no corpo do documento, a impossibilidade de utilizar referência direta ao endereço eletrônico do IMS, para obter a versão atualizada da descrição de um objeto de aprendizado implica em manter uma cópia adaptada em arquivo local no servidor.

A seguir é apresentada uma listagem deste arquivo.

```

<!-- imsmd_rootv1p2.dtd -->
<!-- extension: The extension element is used to create extensions at anyplace in the XML instantiation.
If the extension contains only elements from this DTD, maintaining those content models,
then additional elements do not need to be declared.
It is encouraged that extensions be created from the existing library of elements whenever possible. -->

<!ELEMENT extension ANY>

<!ELEMENT lom (general?, lifecycle?, metametadata?, technical?, educational?, rights?, relation*,
annotation*, classification*)>
  <!ELEMENT general (identifier?, title?, catalogentry*, language*, description*, keyword*, coverage*,
structure?,
aggregationlevel?, extension?)>
    <!ELEMENT identifier (#PCDATA)>
    <!ELEMENT title (langstring+)>
      <!ELEMENT langstring (#PCDATA)>
      <!ATTLIST langstring
        xml:lang NMTOKEN #IMPLIED>
    <!ELEMENT catalogentry (catalog, entry, extension?)>
      <!ELEMENT catalog (#PCDATA)>
      <!ELEMENT entry (langstring+)>
      <!ELEMENT language (#PCDATA)>
      <!ELEMENT description (langstring+)>
      <!ELEMENT keyword (langstring+)>
      <!ELEMENT coverage (langstring+)>
      <!ELEMENT structure (source, value)>
        <!ELEMENT source (langstring+)>
        <!ELEMENT value (langstring+)>
      <!ELEMENT aggregationlevel (source, value)>

```

```

<!ELEMENT lifecycle (version?, status?, contribute*, extension?)>
<!ELEMENT version (langstring+)>
<!ELEMENT status (source, value)>
<!ELEMENT contribute (role, centity*, date?, extension?)>
<!ELEMENT role (source, value)>
<!-- centity: The is the ENTITY element. The word ENTITY is reserved within XML,
so the name has been changed to centity to stand for "Contributing Entity". -->
<!ELEMENT centity (vcard)>
<!ELEMENT vcard (#PCDATA)>
<!ELEMENT date (datetime?, description?)>
<!ELEMENT datetime (#PCDATA)>

<!ELEMENT metametadata (identifier?, catalogentry*, contribute*, metadatascheme*, language?, extension?)>
<!ELEMENT metadatascheme (#PCDATA)>

<!ELEMENT technical (format*, size?, location*, requirement*, installationremarks?,
otherplatformrequirements?,
duration?, extension?)>
<!ELEMENT format (#PCDATA)>
<!ELEMENT size (#PCDATA)>
<!ELEMENT location (#PCDATA)>
<!ATTLIST location
type (URI | TEXT) #IMPLIED>

<!ELEMENT requirement (type?, name?, minimumversion?, maximumversion?, extension?)>
<!ELEMENT type (source, value)>
<!ELEMENT name (source, value)>
<!ELEMENT minimumversion (#PCDATA)>
<!ELEMENT maximumversion (#PCDATA)>
<!ELEMENT installationremarks (langstring+)>
<!ELEMENT otherplatformrequirements (langstring+)>
<!ELEMENT duration (datetime?, description?)>

<!ELEMENT educational (interactivitytype?, learningresourcetype*, interactivitylevel?, semanticdensity?,
intendedenduserrole*, context*, typicalagerange*, difficulty?, typicallearningtime?, description?, language*,
extension?)>
<!ELEMENT interactivitytype (source, value)>
<!ELEMENT learningresourcetype (source, value)>
<!ELEMENT interactivitylevel (source, value)>
<!ELEMENT semanticdensity (source, value)>
<!ELEMENT intendedenduserrole (source, value)>
<!ELEMENT context (source, value)>
<!ELEMENT typicalagerange (langstring+)>
<!ELEMENT difficulty (source, value)>
<!ELEMENT typicallearningtime (datetime?, description?)>

<!ELEMENT rights (cost?, copyrightandotherrestrictions?, description?, extension?)>
<!ELEMENT cost (source, value)>
<!ELEMENT copyrightandotherrestrictions (source, value)>

<!ELEMENT relation (kind?, resource?, extension?)>

```

```
<!ELEMENT kind (source, value)>  
<!ELEMENT resource (identifier?, description?, catalogentry*, extension?)>  
  
<!ELEMENT annotation (person?, date?, description, extension?)>  
<!ELEMENT person (vcard)>  
  
<!ELEMENT classification (purpose?, taxonpath*, description?, keyword*, extension?)>  
<!ELEMENT purpose (source, value)>  
<!ELEMENT taxonpath (source?, taxon?)>  
<!ELEMENT taxon (id?, entry?, taxon?)>  
<!ELEMENT id (#PCDATA)>
```

10.2. Listagem do Arquivo DAD

O arquivo DAD tem por objetivo atuar em conjunto com o arquivo DTD e definir tabelas paralelas que serão geradas automaticamente, quando da atualização dos dados XML. Estas tabelas paralelas deverão conter os conteúdos dos rótulos cuja expectativa de acesso seja maior, beneficiando deste modo o desempenho do conjunto.

Segue a seguir a listagem do arquivo DAD utilizado neste protótipo.

```
<?xml version="1.0"?>
<!DOCTYPE LO SYSTEM "E:\dxx\dtd\ims.dtd">
<DAD>
<dtid>ims.dtd</dtid>
<validation>YES</validation>
<Xcolumn>
  <table name="LO_side_tab">
    <column name="LO_title"
      type="varchar(1000)"
      path="/lom/general/title/langstring"
      multi_occurrence="NO"/>
    <column name="LO_description"
      type="varchar(2000)"
      path="/lom/general/description/langstring"
      multi_occurrence="NO"/>
  </table>
  <table name="LO_technical_format_tab">
    <column name="LO_technical_format"
      type="varchar(500)"
      path="/lom/technical/format"
      multi_occurrence="YES"/>
  </table>
  <table name="LO_technical_location_tab">
    <column name="LO_technical_location"
      type="varchar(1000)"
      path="/lom/technical/location"
      multi_occurrence="YES"/>
  </table>
</Xcolumn>
</DAD>
```


10.3. Documentação da Aplicação LO_DB

Neste item é apresentada a documentação da aplicação LO_DB, conforme produzido pela plataforma de desenvolvimento Together 6.0.

Esta documentação está escrita no idioma inglês, assim como, também estão neste idioma todos os comentários acrescentados ao código fonte dos programas.

10.3.1. Root Package

Class Diagrams

- diagram** <default>
- diagram** Architecture View
- diagram** Object View
- diagram** OpenFirst
- diagram** System Overview

Deployment Diagrams

- diagram** LODB System

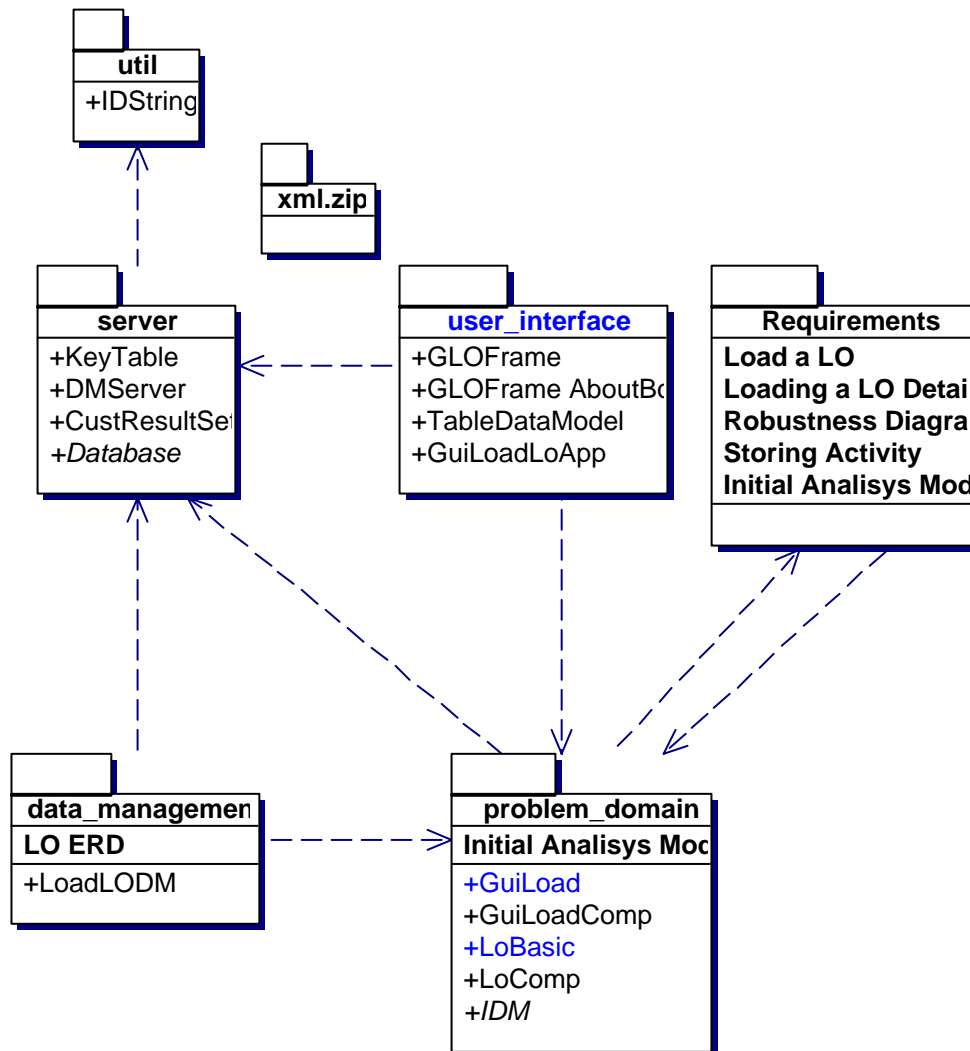
Subpackages

- package** data_management
- package** problem_domain
- package** Requirements
- package** server
- package** user_interface
- package** util
- package** xml*zip

10.3.1.1. Class Diagrams

10.3.1.1.1.
Class Diagram <default>

package: <default>

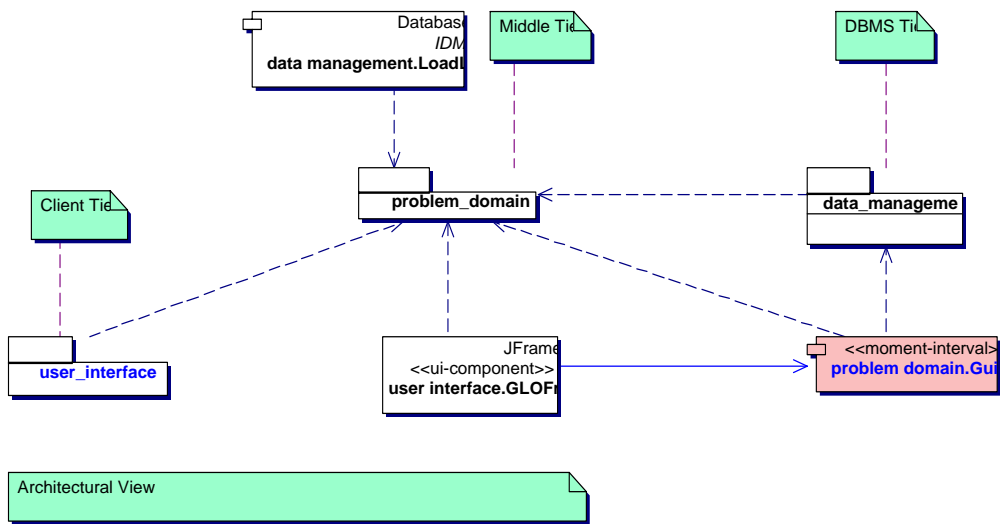


Package Nodes

- data_management
- problem_domain
- Requirements
- server
- user_interface
- util
- xml.zip

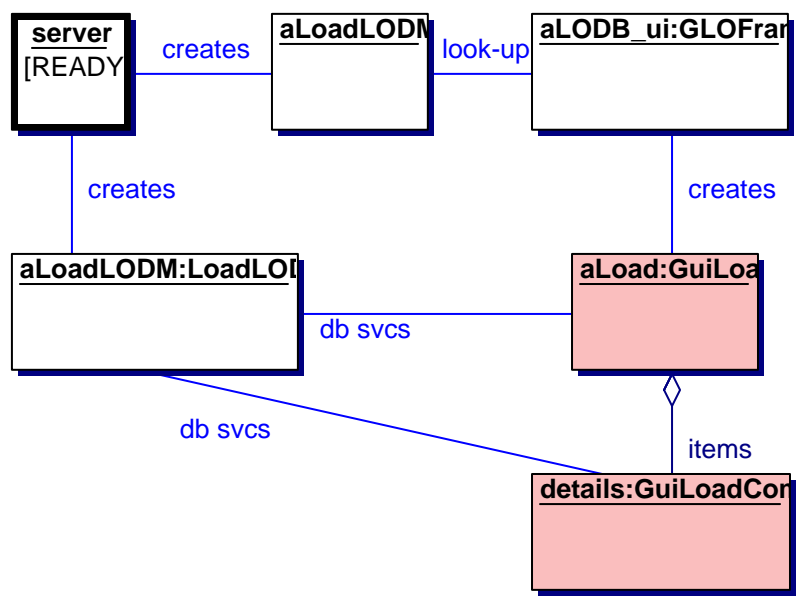
10.3.1.1.2.
Class Diagram Architecture View

package: <default>



10.3.1.1.3.
Class Diagram Object View

package: <default>

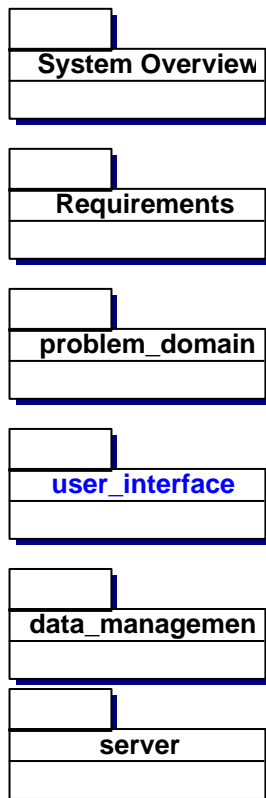


Objects

aLoad
aLoadLODM
aLoadLODM
aLODB_ui
details
server

**10.3.1.1.4.
Class Diagram OpenFirst**

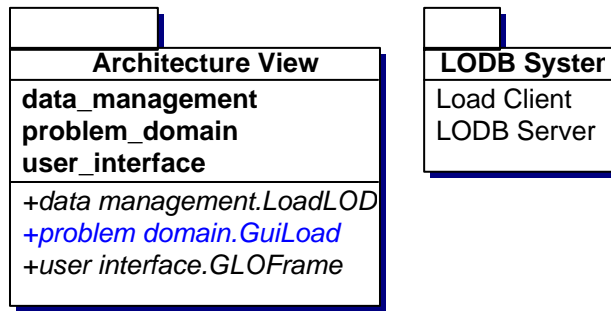
package: <default>



This diagram is used to show how to navigate through the LO_DB system.

**10.3.1.1.5.
Class Diagram System Overview**

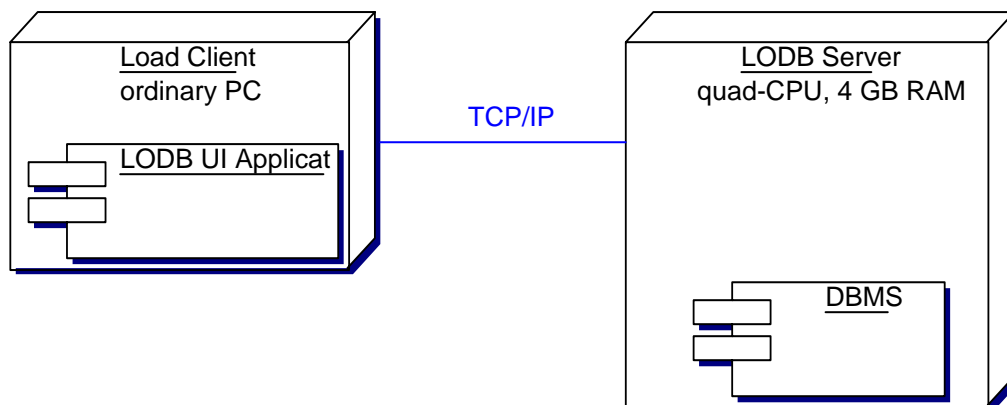
package: <default>



This logical diagram shows the architectural views.

10.3.1.2. Deployment Diagram LODB System

package: <default>



This presents a view of what the LO_DB system deployment might look like.

description:

Shows the configuration of run-time processing elements and the software components, processes, and objects that live on them.

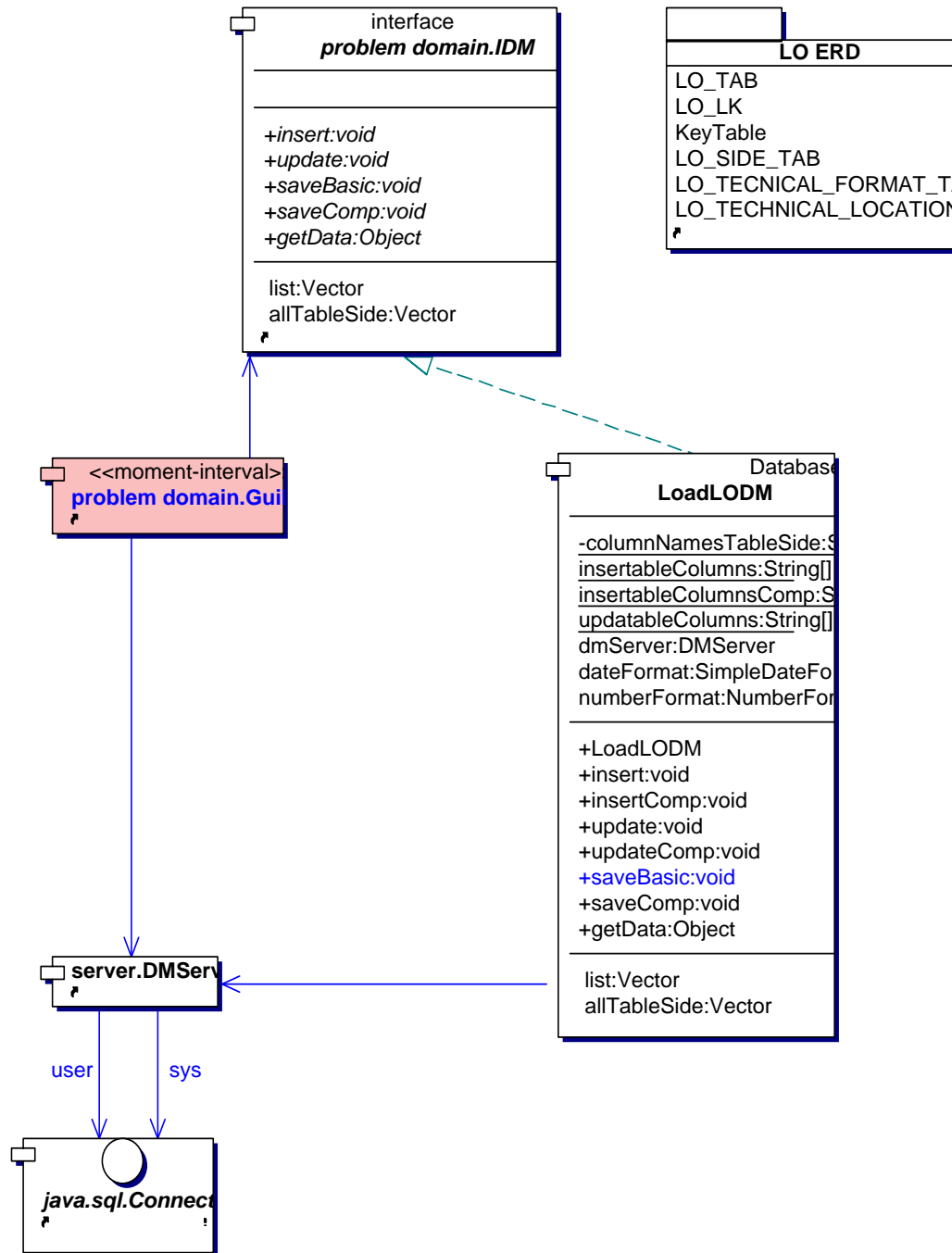
10.3.2. Package data_management

Class Diagrams

diagram data_management

*Interaction Diagrams***diagram** Saving Basic Info*ER-Diagrams***diagram** LO ERD*Classes***class** LoadLODM**10.3.2.1.****Class Diagram data management**

package: data_management



10.3.2.2. Class Detail

10.3.2.2.1. Class LoadLODM

package: **data_management**

```
server.Database
|
+--data_management.LoadLODM
```

```
public class LoadLODM
```

Extends:

server.Database

Implements:

problem_domain.IDM

Class LoadDM

Description: It is a class that extends the Database class and it implements the IDM class

<i>Field Summary</i>	
private final static String	columnNameTableSide Used to extract data from parallel tables generated by DB2
SimpleDateFormat	dateFormat
DMServer	dmServer
final static String	insertableColumns Used to insert basic information
final static String	insertableColumnsComp Used to insert component information
NumberFormat	numberFormat
final static String	updatableColumns Used to update basic info

<i>Constructor Summary</i>	
public	LoadLODM (DMServer dmServer) Constructor must be passed the Server instance.

<i>Method Summary</i>	
public <code>Vector</code>	getAllTableSide() This retrieves all of the loaded titles in the database
public <code>Object</code>	getData(CustResultSet rs) Return a set of values (id, title)
public <code>Vector</code>	getList() Retrieval of the ID and name of an object.
public <code>void</code>	insert(Object obj) Prepare SQL command for inserting basic info related to a LO, such as, XML file, audio, video and image data files, as well, as others.
public <code>void</code>	insertComp(Object obj) Prepare SQL command for inserting components info
public <code>void</code>	saveBasic(Object obj) If the object exists do an update otherwise insert
public <code>void</code>	saveComp(Object obj) Execute insertComp, forecast for future development
public <code>void</code>	update(Object obj) Prepare SQL command for update basic info
public <code>void</code>	updateComp(Object obj) Prepare SQL command for update component info

Field Detail**columnNamesTableSide**

```
private final static String columnNamesTableSide = new String[]
{
    "ID"
    , "LO_TITLE"
}
```

Used to extract data from parallel tables generated by DB2

dateFormat

```
SimpleDateFormat dateFormat = null
```

dmServer

```
DMServer dmServer = null
```

insertableColumns

```
final static String insertableColumns = new String[]
{
    "ID"
    , "LO_XML"
    , "LO_AUDIO"
    , "LO_IMAGE"
    , "LO_VIDEO"
    // , "LO_OTHER" // not implemented yet
    // , "LO_EXER" // not implemented yet
    // , "LO_ASSE" // not implemented yet
}
```

Used to insert basic information

insertableColumnsComp

```
final static String insertableColumnsComp = new String[]
{
    "ID"
    , "ID_COMP"
    , "LK_ORD"
}
```

Used to insert component information

numberFormat

`NumberFormat` numberFormat

updatableColumns

```
final static String updatableColumns = new String[]
{
    "LO_XML"
    , "LO_AUDIO"
    , "LO_IMAGE"
    , "LO_VIDEO"
}
```

Used to update basic info

Constructor Detail**LoadLODM**

```
public LoadLODM(DMServer dmServer)
```

Constructor must be passed the Server instance.

Method Detail**getAllTableSide**

```
public Vector getAllTableSide()
```

This retrieves all of the loaded titles in the database

Throws:

SQLException

getData

```
public Object getData(CustResultSet rs)
```

Return a set of values (id, title)

Throws:

SQLException

getList

```
public Vector getList()
```

Retrieval of the ID and name of an object.

Throws:

SQLException

insert

```
public void insert(Object obj)
```

Prepare SQL command for inserting basic info related to a LO, such as, XML file, audio, video and image data files, as well, as others.

Throws:

SQLException

insertComp

```
public void insertComp(Object obj)
```

Prepare SQL command for inserting components info

Throws:

SQLException

saveBasic

```
public void saveBasic(Object obj)
```

If the object exists do an update otherwise insert

Throws:

SQLException

saveComp

```
public void saveComp(Object obj)
```

Execute insertComp, forecast for future development

Throws:

SQLException

update

```
public void update(Object obj)
```

Prepare SQL command for update basic info

Throws:

SQLException

updateComp

```
public void updateComp(Object obj)
```

Prepare SQL command for update component info

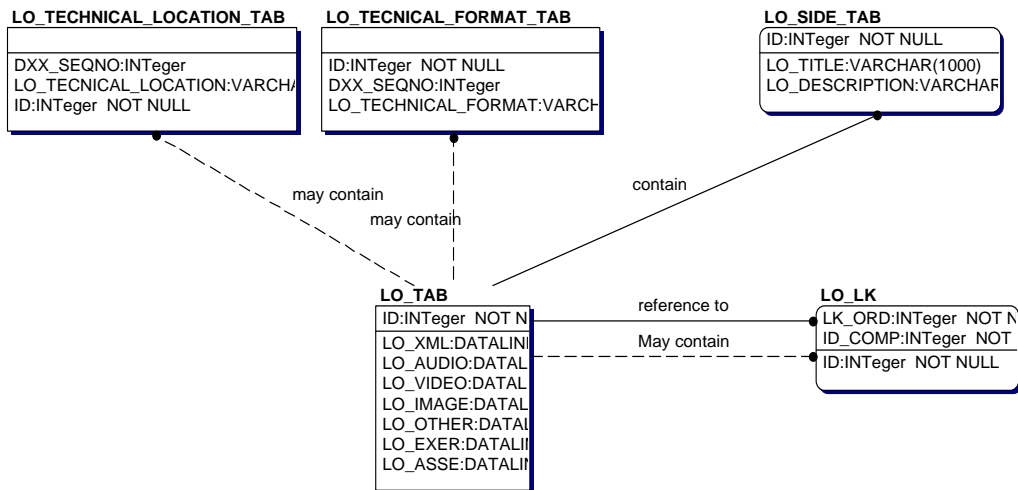
Throws:

SQLException

10.3.2.3. Entity Relationship Diagrams

ER-Diagram *LO ERD*

package: data_management



KeyTable

TableName:VARCHAR(25) NO
ColumnName:VARCHAR(25)
NextAvailableKey:INTEger

ERD schema for LO_DB.

<i>ER-Entity Summary</i>	
KeyTable	Used to store next key for each.

<i>ER-Entity Summary</i>	
LO_LK	Stores the components tuples with proper execution order.
LO_SIDE_TAB	Entity automatically generated by DB2, after specification and enabling LO_TAB for DB2 XML extender.
LO_TAB	Stores lo's basic information, as: XML description, audio, video, image and/or other technical files, and the practice and assessment files.
LO_TECHNICAL_LOCATION_TAB	Entity automatically generated by DB2, after specification and enabling LO_TAB for DB2 XML extender.
LO_TECNICAL_FORMAT_TAB	Entity automatically generated by DB2, after specification and enabling LO_TAB for DB2 XML extender.

10.3.3. Package ***problem_domain***

This diagram shows business-level classes.

Class Diagrams

diagram problem_domain

Interaction Diagrams

diagram GuiLoad.GuiLoad(1)

diagram GuiLoad.saveBasic(1)

diagram List All Titles

Classes

class GuiLoad

class GuiLoadComp

class LoBasic

class LoComp

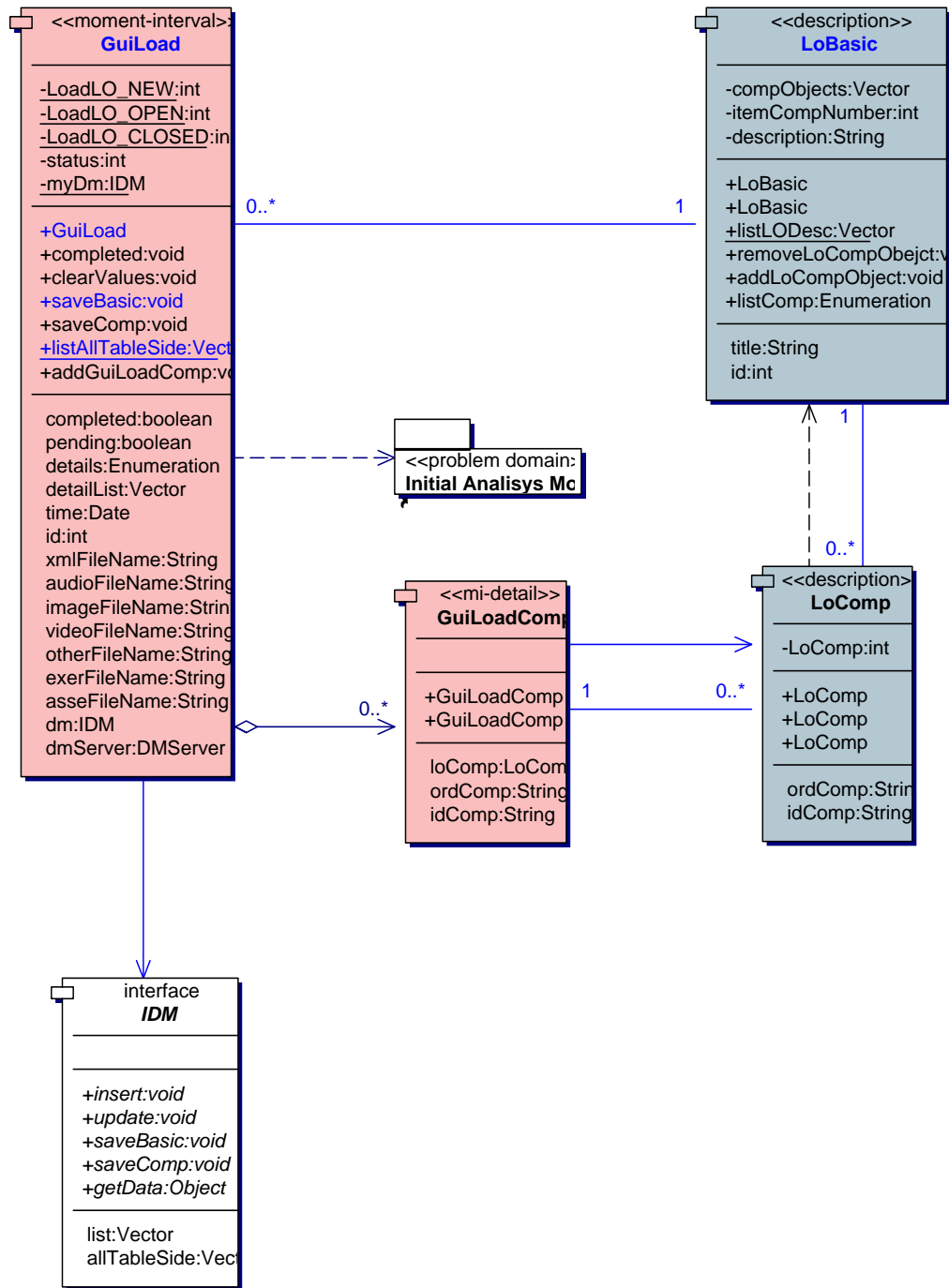
Interfaces

interface IDM

10.3.3.1. Class Diagrams

10.3.3.1.1. Class Diagram *problem_domain*

package: problem_domain



This diagram shows business-level classes.

type:

Problem Domain

Class Nodes

GuiLoad

GuiLoadComp

LoBasic

LoComp

Interface Nodes

IDM

**10.3.3.2.
Class Detail**

**10.3.3.2.1.
Class *GuiLoad***

package: **problem_domain**

public class GuiLoad

Class GuiLoad

Description Represents the load of a LO using a graphical user interface

<i>Field Summary</i>	
private String	asseFileName file name Assessment
private LoadLo	attributel
private String	audioFileName file name Audio
private Vector	detailList This is the list of components
private DMServer	dmServer
private String	exerFileName file name Exercise
private int	id This will be used as a unique identifier for LO objects.
private String	imageFileName file name Image

<i>Field Summary</i>	
private LoBasic	lnkLBasic
private final static int	LoadLO_CLOSED To follow if it is a load had finished
private final static int	LoadLO_NEW To follow if it is a new load
private final static int	LoadLO_OPEN To follow if it the load process had began and not yet finished
private static IDM	myDm Access to the Data Management Layer.
private String	otherFileName file name Other
private int	status States if the loading is in processing state or has been completed.
private Date	time Display current date
private String	videoFileName file name Video
private String	xmlFileName file name XML

<i>Constructor Summary</i>	
public	GuiLoad ()

<i>Method Summary</i>	
public void	addGuiLoadComp (GuiLoadComp aGuiLoadComp)

<i>Method Summary</i>	
public void	clearValues () Clear/reset the values
public void	completed () Marks GuiLoad as completed
public String	getAsseFileName ()
public String	getAudioFileName ()
public Vector	getDetailList ()
public Enumeration	getDetails ()
public String	getExerFileName ()
public int	getId ()
public String	getImageFileName ()
public String	getOtherFileName ()
public Date	getTime ()
public String	getVideoFileName ()
public String	getXmlFileName ()
public boolean	isCompleted () Returns if GuiLoad is completed or not yet.
public boolean	isPending () Indicate GuiLoad is pending
public static Vector	listAllTableSide () Returns a list of all titles loaded
public void	saveBasic () Persist basic info file about LO
public void	saveComp () Persist components info about LO
public void	setAsseFileName (String asseFileName)
public void	setAudioFileName (String audioFileName)

<i>Method Summary</i>	
public void	setCompleted (boolean bool)
public void	setDm (IDM dm)
public void	setDmServer (DMServer dmServer)
public void	setExerFileName (String exerFileName)
public void	setId (int id)
public void	setImageFileName (String imageFileName)
public void	setOtherFileName (String otherFileName)
public void	setTime (Date time)
public void	setVideoFileName (String videoFileName)
public void	setXmlFileName (String xmlFileName)

Field Detail

asseFileName

private String asseFileName

file name Assessment

attribute1

private LoadLo attribute1

audioFileName

private String audioFileName

file name Audio

detailList

private Vector detailList

This is the list of components

Associates:

problem_domain.GuiLoadComp

dmServer

```
private DMServer dmServer = null
```

exerFileName

```
private String exerFileName
```

file name Exercise

id

```
private int id = 0
```

This will be used as a unique identifier for LO objects.

imageFileName

```
private String imageFileName
```

file name Image

InkLBasic

```
private LoBasic InkLBasic
```

LoadLO_CLOSED

```
private final static int LoadLO_CLOSED = 2
```

To follow if it is a load had finished

LoadLO_NEW

```
private final static int LoadLO_NEW = 0
```

To follow if it is a new load

LoadLO_OPEN

```
private final static int LoadLO_OPEN = 1
```

To follow if it the load process had began and not yet finished

myDm

```
private static IDM myDm
```

Access to the Data Management Layer.

otherFileName

```
private String otherFileName
```

file name Other

status

```
private int status = LoadLO_NEW
```

States if the loading is in processing state or has been completed.

time

```
private Date time
```

Display current date

videoFileName

```
private String videoFileName
```

file name Video

xmlFileName

```
private String xmlFileName
```

file name XML

Constructor Detail**GuiLoad**

```
public GuiLoad()
```

Method Detail**addGuiLoadComp**

```
public void addGuiLoadComp(GuiLoadComp aGuiLoadComp)
```

clearValues

```
public void clearValues()
```

Clear/reset the values

completed

```
public void completed()
```

Marks GuiLoad as completed

getAsseFileName

```
public String getAsseFileName()
```

getAudioFileName

```
public String getAudioFileName()
```

getDetailList

```
public Vector getDetailList()
```

getDetails

```
public Enumeration getDetails()
```

getExerFileName

```
public String getExerFileName()
```

getId

```
public int getId()
```

getImageFileName

```
public String getImageFileName()
```

getOtherFileName

```
public String getOtherFileName()
```

getTime

```
public Date getTime()
```

getVideoFileName

```
public String getVideoFileName()
```

getXmlFileName

```
public String getXmlFileName()
```

isCompleted

```
public boolean isCompleted()
```

Returns if GuiLoad is completed or not yet.

isPending

```
public boolean isPending()
```

Indicate GuiLoad is pending

listAllTableSide

```
public static Vector listAllTableSide()
```

Returns a list of all titles loaded

saveBasic

```
public void saveBasic()
```

Persist basic info file about LO

Throws:

SQLException

saveComp

```
public void saveComp()
```

Persist components info about LO

Throws:

SQLException

setAsseFileName

```
public void setAsseFileName(String asseFileName)
```

setAudioFileName

```
public void setAudioFileName(String audioFileName)
```

setCompleted

```
public void setCompleted(boolean bool)
```

setDm

```
public void setDm(IDM dm)
```

setDmServer

```
public void setDmServer(DMServer dmServer)
```

setExerFileName

```
public void setExerFileName(String exerFileName)
```

setId

```
public void setId(int id)
```

setImageFileName

```
public void setImageFileName(String imageFileName)
```

setOtherFileName

```
public void setOtherFileName(String otherFileName)
```

setTime

```
public void setTime(Date time)
```

setVideoFileName

```
public void setVideoFileName(String videoFileName)
```

setXmlFileName

```
public void setXmlFileName(String xmlFileName)
```

10.3.3.2.2.**Class *GuiLoadComp***

```
package: problem_domain
```

```
public class GuiLoadComp
```

Class GuiLoadComp

Description Detail class for loading linking info between LO and it components.

<i>Field Summary</i>	
private String	idComp
private LoComp	loComp
private String	ordComp

<i>Constructor Summary</i>	
public	GuiLoadComp ()
public	GuiLoadComp (LoComp aLoComp)

<i>Method Summary</i>	
public String	getIdComp ()
public LoComp	getLoComp ()
public String	getOrdComp ()
public void	setIdComp (String idComp)
public void	setOrdComp (String ordComp)

Field Detail

idComp

private [String](#) idComp

loComp

private [LoComp](#) loComp

ordComp

private [String](#) ordComp

Constructor Detail**GuiLoadComp**

```
public GuiLoadComp()
```

GuiLoadComp

```
public GuiLoadComp(LoComp aLoComp)
```

Method Detail**getIdComp**

```
public String getIdComp()
```

getLoComp

```
public LoComp getLoComp()
```

getOrdComp

```
public String getOrdComp()
```

setIdComp

```
public void setIdComp(String idComp)
```

setOrdComp

```
public void setOrdComp(String ordComp)
```

10.3.3.2.3.**Class LoBasic**

package: **problem_domain**

```
public class LoBasic
```

Class LoBasic

Description Represents the set of basic information's about a Learning Object

<i>Field Summary</i>	
private LoBasic	attribute1
private Vector	CompObjects

<i>Field Summary</i>	
private <code>String</code>	description Literal description of a LO
private <code>int</code>	id Unique identifier of a LO
private <code>int</code>	itemCompNumber
private <code>String</code>	title

<i>Constructor Summary</i>	
public	LoBasic () All attributes are defaulted to 0
public	LoBasic (int aId, String aLO_xml)

<i>Method Summary</i>	
public <code>void</code>	addLoCompObject ()
public <code>int</code>	getId () Returns the value of the Id of a LO
public <code>String</code>	getTitle () Returns the value of the Title of a LO
public <code>Enumeration</code>	listComp () Returns the set of components linked to LO.
public static <code>Vector</code>	listLODesc () Method to return the relation of LO stored
public <code>void</code>	removeLoCompObejct ()
public <code>void</code>	setId (int aId) Updates the value of the Id of the LO

<i>Method Summary</i>	
<code>public void</code>	setTitle (<code>String</code> title) Updates the value of the Title of the LO.

Field Detail

attribute1

private `LoBasic` attribute1

compObjects

private `Vector` compObjects

description

private `String` description

Literal description of a LO

id

private `int` id

Unique identifier of a LO

itemCompNumber

private `int` itemCompNumber

title

private `String` title

Constructor Detail

LoBasic

`public LoBasic()`

All attributes are defaulted to 0

LoBasic

`public LoBasic(int aId, String aLO_xml)`

Method Detail**addLoCompObject**

```
public void addLoCompObject()
```

getId

```
public int getId()
```

Returns the value of the Id of a LO

getTitle

```
public String getTitle()
```

Returns the value of the Title of a LO

listComp

```
public Enumeration listComp()
```

Returns the set of components linked to LO.

listLODesc

```
public static Vector listLODesc()
```

Method to return the relation of LO stored

removeLoCompObejct

```
public void removeLoCompObejct()
```

setId

```
public void setId(int aId)
```

Updates the value of the Id of the LO

setTitle

```
public void setTitle(String title)
```

Updates the value of the Title of the LO.

10.3.3.2.4. Class *LoComp*

package: **problem_domain**

```
public class LoComp
```

Class *LoComp*

Description Its purpose is to persist LO components.

<i>Field Summary</i>	
private <code>String</code>	idComp
private <code>GuiLoadComp</code>	lnkLoComp
private <code>LoBasic</code>	lnkLoComp1
private <code>int</code>	LoComp
private <code>String</code>	ordComp

<i>Constructor Summary</i>	
public	LoComp ()
public	LoComp (<code>int</code> aId_xml, <code>int</code> aId_seq, <code>String</code> aLO_avi)
public	LoComp (<code>String</code> aOrdComp, <code>String</code> aIdComp)

<i>Method Summary</i>	
public <code>String</code>	getIdComp ()
public <code>String</code>	getOrdComp ()
public <code>void</code>	setIdComp (<code>String</code> aIdComp)
public <code>void</code>	setOrdComp (<code>String</code> aOrdComp)

Field Detail

idComp

private `String` idComp

InkLoComp

```
private GuiLoadComp InkLoComp
```

InkLoComp1

```
private LoBasic InkLoComp1
```

LoComp

```
private int LoComp
```

ordComp

```
private String ordComp
```

Constructor Detail**LoComp**

```
public LoComp ()
```

LoComp

```
public LoComp (int aId_xml, int aId_seq, String aLO_avi)
```

LoComp

```
public LoComp (String aOrdComp, String aIdComp)
```

Method Detail**getIdComp**

```
public String getIdComp ()
```

getOrdComp

```
public String getOrdComp ()
```

setIdComp

```
public void setIdComp (String aIdComp)
```

setOrdComp

```
public void setOrdComp (String aOrdComp)
```


10.3.3.3. Interface Detail

10.3.3.3.1. Interface *IDM*

package: **problem_domain**

All Known Implementing Classes:

[LoadLODM](#)

```
public interface IDM
```

Class IDM

Description The purpose of this interface is to separate general classes from those specific to DM.

<i>Method Summary</i>	
public Vector	getAllTableSide () Return all objects.
public Object	getData (CustResultSet rs)
public Vector	getList () This will return the Id and the Title of the LO
public void	insert (Object obj)
public void	saveBasic (Object obj)
public void	saveComp (Object obj)
public void	update (Object obj)

Method Detail

getAllTableSide

```
public Vector getAllTableSide ()
```

Return all objects.

Throws:

SQLException

getData

```
public Object getData(CustResultSet rs)
```

Throws:

SQLException

getList

```
public Vector getList()
```

This will return the Id and the Title of the LO

Throws:

SQLException

insert

```
public void insert(Object obj)
```

Throws:

SQLException

saveBasic

```
public void saveBasic(Object obj)
```

Throws:

SQLException

saveComp

```
public void saveComp(Object obj)
```

Throws:

SQLException

update

```
public void update(Object obj)
```

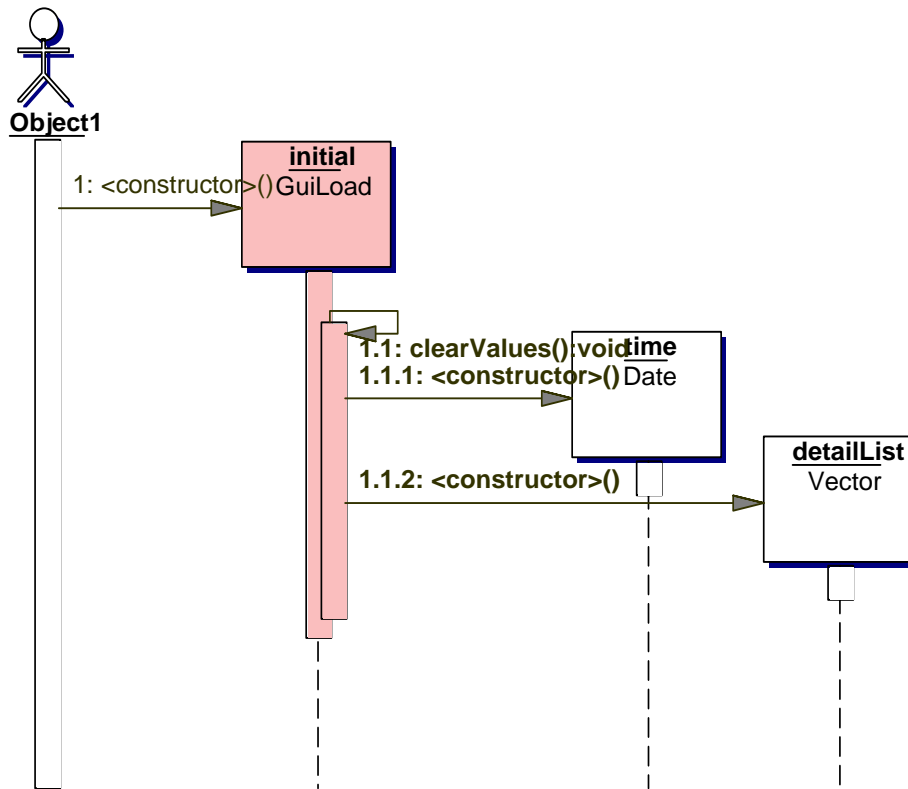
Throws:

SQLException

**10.3.3.4.
Interaction Diagrams**

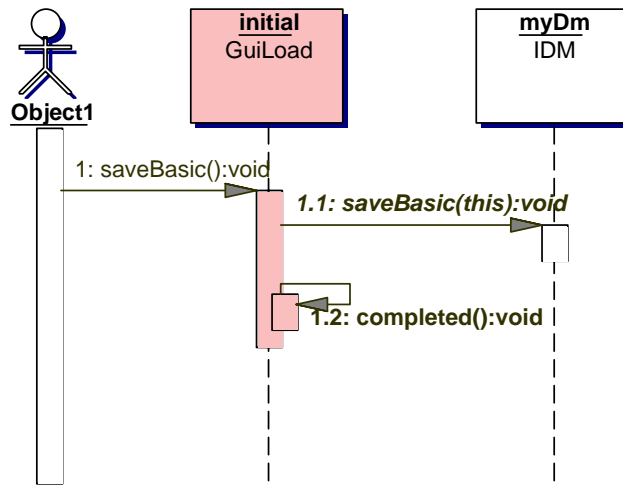
**10.3.3.4.1.
Sequence Diagram GuiLoad.GuiLoad(1)**

package: problem_domain



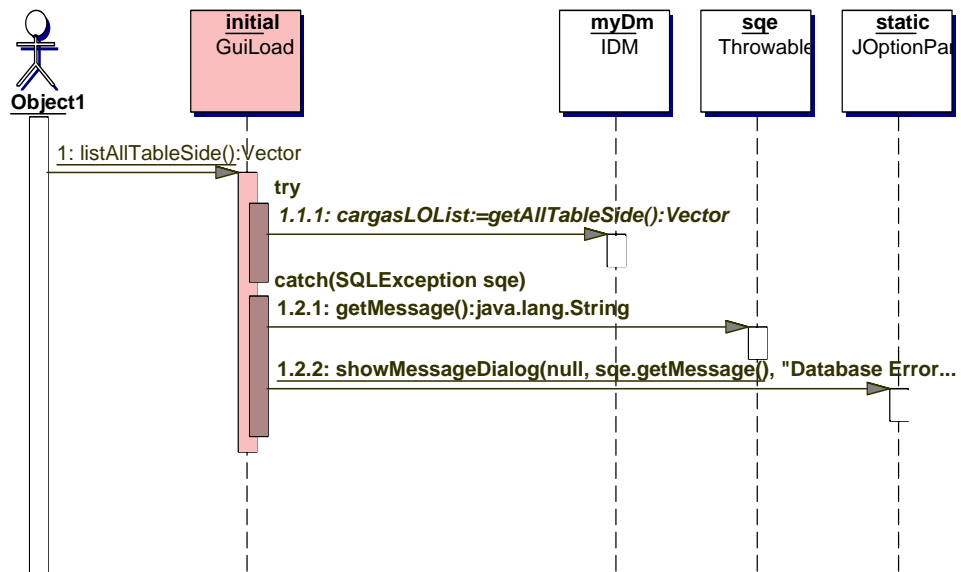
**10.3.3.4.2.
Sequence Diagram GuiLoad.saveBasic(1)**

package: problem_domain



10.3.3.4.3.
Sequence Diagram *List All Titles*

package: problem_domain



10.3.4.
Package *Requirements*

This diagram shows the various requirements artifacts.

Class Diagrams

diagram Requirements

Activity Diagrams

diagram Storing Activity

UseCase Diagrams

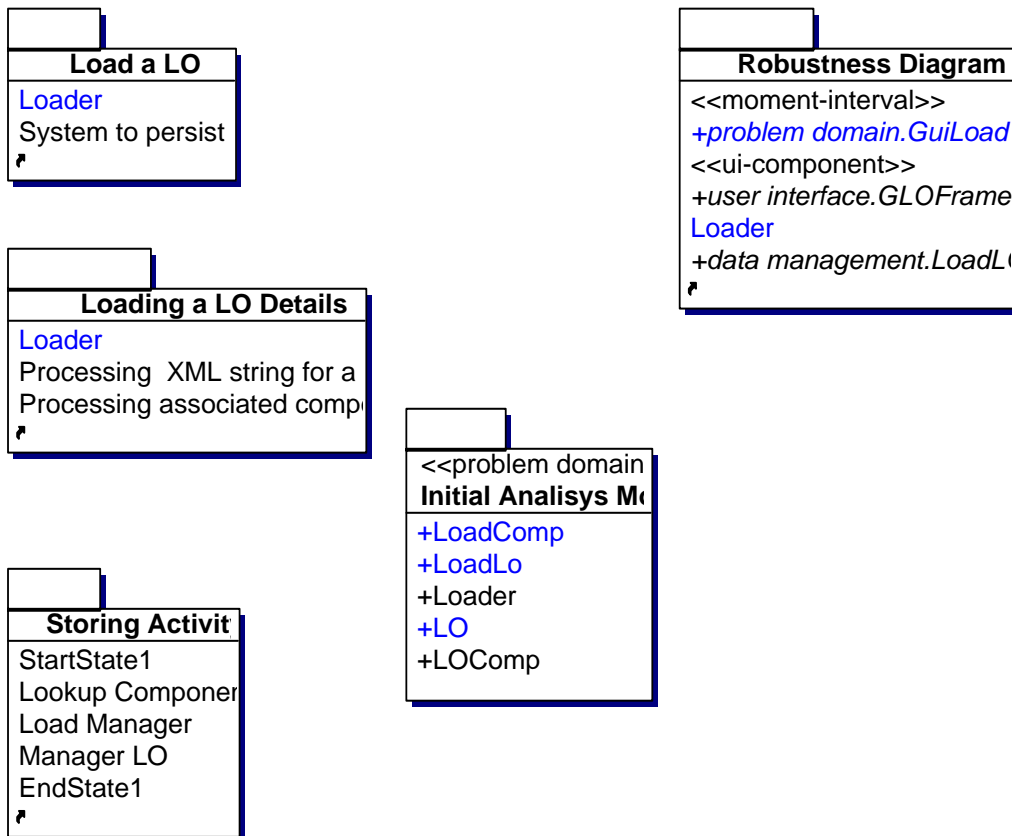
diagram Load a LO

diagram Loading a LO Details

**10.3.4.1.
Class Diagrams**

**10.3.4.1.1.
Class Diagrams Requirements**

package: Requirements



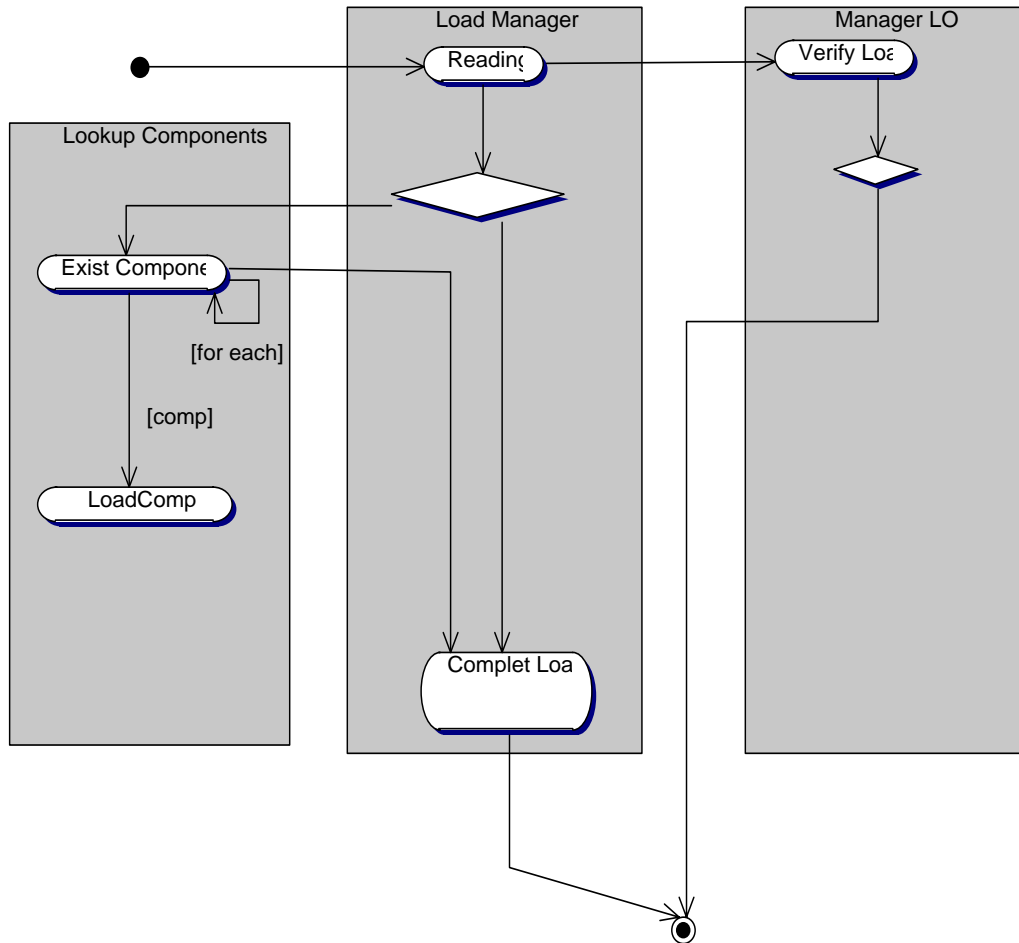
Package Nodes

analysis

**10.3.4.2.
Activity Diagrams**

**10.3.4.2.1.
Activity Diagram Storing Activity**

package: Requirements



This represents the activities required to carry out load of a learning object.

Diagram Contents Summary

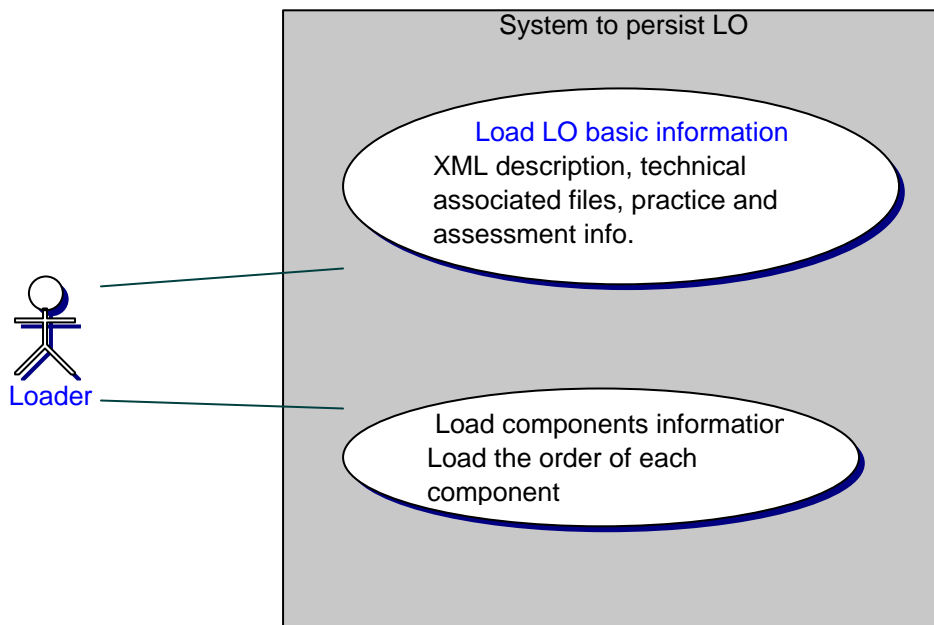
- ▣ **Swim Lane Load Manager**
 - **Activity Compleet Load**
 - **Activity Reading**
 - ◇ **Decision SGBD AVI**
- ▣ **Swim Lane Lookup Components**

- Activity Exist Component
- Activity LoadComp
- ▣ Swim Lane Manager LO
- Activity Verify Load
- ◇ Decision Decision1
- EndState EndState1
- StartState StartState1

**10.3.4.3.
UseCase Diagrams**

**10.3.4.3.1.
UseCase Diagram Load a LO**

package: Requirements



UseCase Load components information

explanation:

Load the order of each component

UseCase Load LO basic information

explanation:

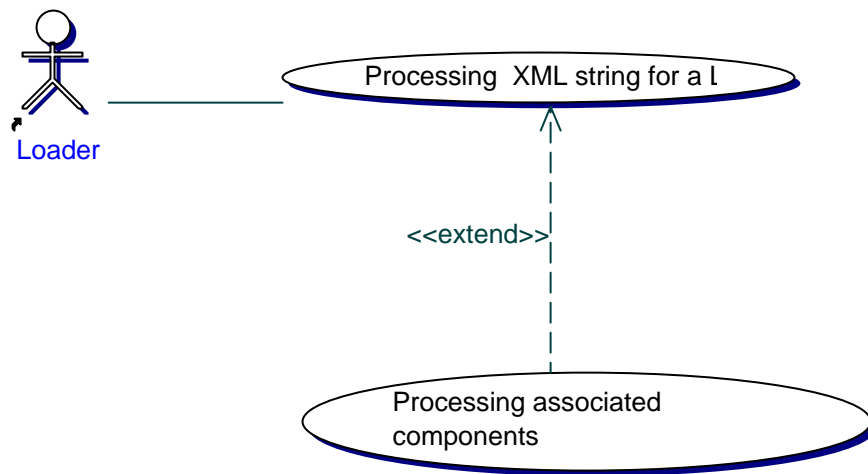
XML description, technical associated files, practice and assessment info.

preconditions:

SGBD must enable AVI support

10.3.4.3.2. UseCase Diagram Loading a LO Details

package: Requirements



10.3.5. Package *Requirements.analysis*

Class Diagrams

diagram analysis

Classes

class LO

class LoadComp

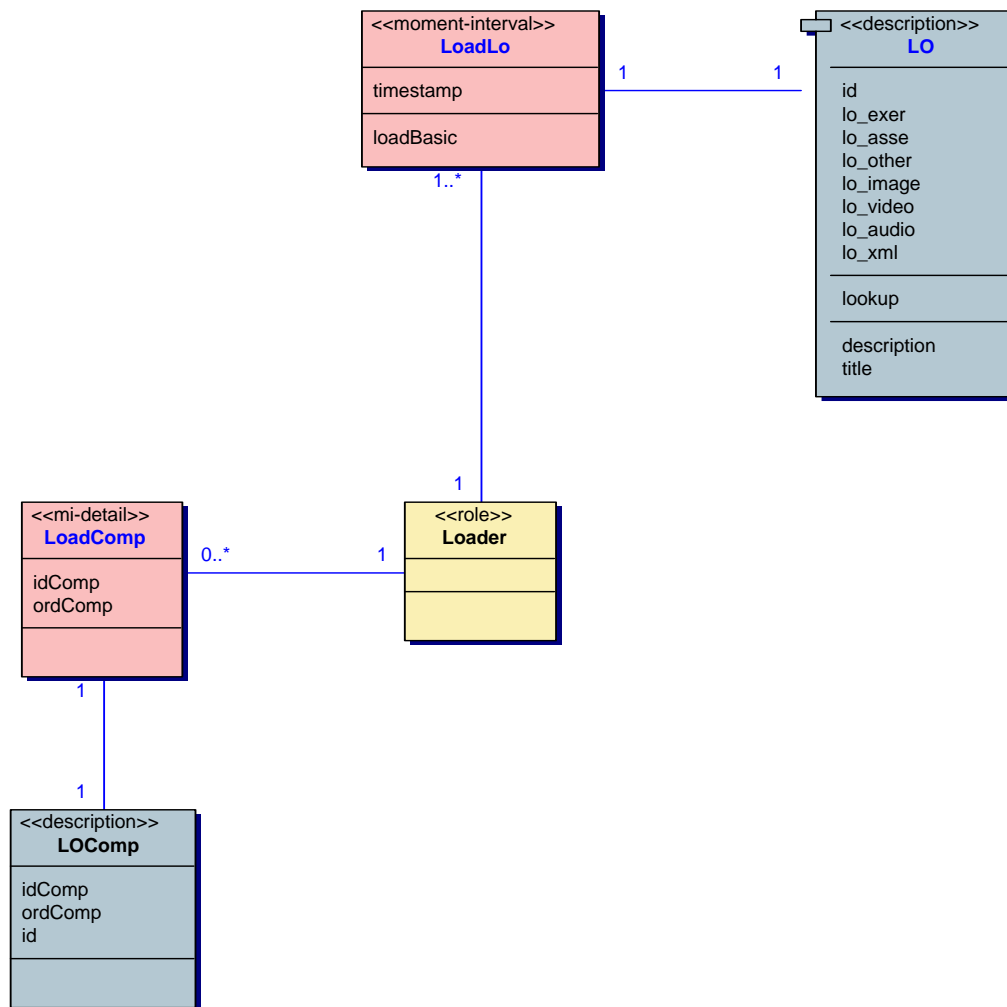
class Loader

class LoadLo
class LOComp

**10.3.5.1.
 Class Diagrams**

**10.3.5.1.1.
 Class Diagrams *analysis***

package: Requirements.analysis



Class Nodes

- LO
- LoadComp
- Loader
- LoadLo

LOComp

10.3.5.2. Class Detail

10.3.5.2.1. Class *LO*

package: **Requirements.analysis**

```
public class LO
```

Class *LO*

Description Main class describing a Learning Object

<i>Field Summary</i>	
private LO	attributel
private int	id
private int	lo_asse
private int	lo_audio
private int	lo_exer
private int	lo_image
private int	lo_other
private int	lo_video
private String	lo_xml
private IDString	title

<i>Method Summary</i>	
public IDString	getDescription()
public IDString	getTitle()
public void	lookup(int ident_xml)
public void	setTitle(IDString title)

Field Detail**attribute1**

private `LO` attribute1

id

private `int` id

lo_asse

private `int` lo_asse

lo_audio

private `int` lo_audio

lo_exer

private `int` lo_exer

lo_image

private `int` lo_image

lo_other

private `int` lo_other

lo_video

private `int` lo_video

lo_xml

private `String` lo_xml

title

private `IDString` title

Method Detail**getDescription**

public `IDString` getDescription()

getTitle

public `IDString` getTitle()

lookup

```
public void lookup(int ident_xml)
```

setTitle

```
public void setTitle(IDString title)
```

10.3.5.2.2.**Class LoadComp**

package: **Requirements.analysis**

```
public class LoadComp
```

Class LoadComp

Description The purpose is to describe the loading of a LO component as a simple transaction, using the main's LO and its component's identifiers, and the execution order of the component

<i>Field Summary</i>	
private GuiLoadComp	attribute1
private int	idComp
private LOComp	lnkLOComp
private int	ordComp

Field Detail**attribute1**

```
private GuiLoadComp attribute1
```

idComp

```
private int idComp
```

lnkLOComp

```
private LOComp lnkLOComp
```

ordComp

```
private int ordComp
```

**10.3.5.2.3.
Class Loader**

package: **Requirements.analysis**

```
public class Loader
```

Class Loader

Description Perhaps, in the future, it will be used to identify the operator of the loading operation

<i>Field Summary</i>	
private LoadLo	LnkGuiLoad
private LoadComp	LnkLoComp

Field Detail

LnkGuiLoad

```
private LoadLo LnkGuiLoad
```

LnkLoComp

```
private LoadComp LnkLoComp
```

**10.3.5.2.4.
Class LoadLo**

package: **Requirements.analysis**

```
public class LoadLo
```

Class LoadLo

Description This will describe the loading basic information about a LO, generating an unique identifier.

<i>Field Summary</i>	
private LoadLo	attribute1
private LO	LnkLoBasic

<i>Field Summary</i>	
private int	Timestamp

<i>Method Summary</i>	
public void	loadBasic()

Field Detail

attribute1

private LoadLo attribute1

InkLoBasic

private LO InkLoBasic

timestamp

private int timestamp

Method Detail

loadBasic

public void loadBasic()

10.3.5.2.5.

Class LOComp

package: **Requirements.analysis**

public class LOComp

<i>Field Summary</i>	
private int	id
private int	idComp
private int	ordComp

Field Detail**id**

private int id

idComp

private int idComp

ordComp

private int ordComp

10.3.6.**Package Server***Class Diagrams***diagram** server*Classes***class** CustResultSet**class** Database**class** DMServer**class** KeyTable**10.3.6.1.****Class Diagrams****10.3.6.1.1.****Class Diagrams Server****package:** server



Class Nodes

- CustResultSet
- Database
- DMServer
- KeyTable

10.3.6.2. Class Detail

10.3.6.2.1. Class *CustResultSet*

```

package: server

public class CustResultSet

Class CustResultSet
    
```

Description This will be used to work with set of data retrieved from db

<i>Field Summary</i>	
ResultSet	resultSet

<i>Constructor Summary</i>	
public	CustResultSet (ResultSet rs)

<i>Method Summary</i>	
public boolean	getBoolean (String columnName)
public Date	getDate (String columnName)
public double	getDouble (String columnName)
public int	getInt (String columnName)
public String	getString (String columnName)
public boolean	next ()

Field Detail

resultSet

[ResultSet](#) resultSet = null

Constructor Detail

CustResultSet

public CustResultSet ([ResultSet](#) rs)

Method Detail

getBoolean

public [boolean](#) getBoolean ([String](#) columnName)

Throws:

SQLException

getDate

public [Date](#) getDate ([String](#) columnName)

Throws:

SQLException

getDouble

```
public double getDouble(String columnName)
```

Throws:

SQLException

getInt

```
public int getInt(String columnName)
```

Throws:

SQLException

getString

```
public String getString(String columnName)
```

Throws:

SQLException

next

```
public boolean next()
```

Throws:

SQLException

10.3.6.2.2.

Class *Database*

```
package: server
```

```
public abstract class Database
```

Class Database

Description This is an abstract class that contains general db activities

<i>Field Summary</i>	
private Connection	connection
private String	tableName
private String	tableNameComp
private String	tableNameSide

<i>Constructor Summary</i>	
protected	Database (Connection connection, String tableName)
protected	Database (Connection connection, String tableName, String tableNameComp)
protected	Database (Connection connection, String tableName, String tableNameComp, String tableNameSide)

<i>Method Summary</i>	
public void	commit ()
protected void	delete (int id)
protected void	delete (String whereClause)
protected void	executeUpdate (String query)
protected Object	get (String columnNames, int id)
protected Object	get (String columnNames, String whereClause)
private String	getColumnNameString (String columnNames)
protected abstract Object	getData (CustResultSet rs)
protected String	getDBDate (Date date)
protected String	getDBString (String str)
protected IDString	getIDString (int id)

<i>Method Summary</i>	
protected <code>IDString</code>	getIDString (<code>String</code> idColumn, <code>String</code> textColumn, <code>String</code> whereClause)
protected <code>IDString</code>	getIDString (<code>ResultSet</code> rs, <code>String</code> idColumnName, <code>String</code> textColumnName)
private <code>IDString</code>	getIDString (<code>CustResultSet</code> rs, <code>String</code> idColumnName, <code>String</code> textColumnName)
protected <code>Vector</code>	getIDStringList ()
protected <code>Vector</code>	getIDStringList (<code>String</code> idColumnName, <code>String</code> textColumnName)
protected <code>Vector</code>	getIDStringList (<code>String</code> idColumnName, <code>String</code> textColumnName, <code>String</code> whereClause)
protected <code>Vector</code>	getStringList (<code>String</code> textColumnName, <code>boolean</code> selectDistinct)
protected <code>Vector</code>	getStringList (<code>String</code> textColumnName, <code>String</code> whereClause, <code>boolean</code> selectDistinct)
protected <code>String</code>	getTableName ()
protected <code>String</code>	getTableNameComp ()
protected <code>void</code>	insert (<code>String</code> columnNames, <code>String</code> columnValues)
protected <code>void</code>	insertComp (<code>String</code> columnNames, <code>String</code> columnValues)
public <code>void</code>	rollback ()
protected <code>CustResultSet</code>	select (<code>String</code> query)

<i>Method Summary</i>	
protected <code>Vector</code>	<code>selectList</code> (<code>String</code> columnNames, <code>String</code> whereClause)
protected <code>Vector</code>	<code>selectList</code> (<code>String</code> columnNames, <code>String</code> whereClause, <code>String</code> orderBy)
protected <code>void</code>	<code>setTableName</code> (<code>String</code> tableName)
protected <code>void</code>	<code>setTableNameComp</code> (<code>String</code> tableNameComp)
protected <code>void</code>	<code>update</code> (<code>String</code> columnNames, <code>String</code> columnValues, <code>String</code> whereClause)

*Field Detail***connection**

```
private Connection connection = null
```

tableName

```
private String tableName = ""
```

tableNameComp

```
private String tableNameComp = ""
```

tableNameSide

```
private String tableNameSide = ""
```

*Constructor Detail***Database**

```
protected Database(Connection connection, String tableName)
```

Database

```
protected Database(Connection connection, String tableName, String tableNameComp)
```

Database

```
protected Database(Connection connection, String tableName, String  
tableNameComp, String tableNameSide)
```

Method Detail**commit**

```
public void commit()
```

Throws:

SQLException

delete

```
protected void delete(int id)
```

Throws:

SQLException

delete

```
protected void delete(String whereClause)
```

Throws:

SQLException

executeUpdate

```
protected void executeUpdate(String query)
```

Throws:

SQLException

get

```
protected Object get(String columnNames, int id)
```

Throws:

SQLException

get

```
protected Object get(String columnNames, String whereClause)
```

Throws:

SQLException

getColumnNameString

```
private String getColumnNameString(String columnNames)
```

getData

```
protected abstract Object getData(CustResultSet rs)
```

Throws:

SQLException

getDBDate

```
protected String getDBDate(Date date)
```

getDBString

```
protected String getDBString(String str)
```

getIDString

```
private IDString getIDString(CustResultSet rs, String idColumnName,  
String textColumnName)
```

Throws:

SQLException

getIDString

```
protected IDString getIDString(int id)
```

Throws:

SQLException

getIDString

```
protected IDString getIDString(ResultSet rs, String idColumnName,  
String textColumnName)
```

Throws:

SQLException

getIDString

```
protected IDString getIDString(String idColumn, String textColumn,  
String whereClause)
```

Throws:

SQLException

getIDStringList

```
protected Vector getIDStringList(String idColumnName, String  
textColumnName)
```

Throws:

SQLException

getIDStringList

```
protected Vector getIDStringList()
```

Throws:

SQLException

getIDStringList

```
protected Vector getIDStringList(String idColumnName, String  
textColumnName, String whereClause)
```

Throws:

SQLException

getStringList

```
protected Vector getStringList(String textColumnName, String  
whereClause, boolean selectDistinct)
```

Throws:

SQLException

getStringList

```
protected Vector getStringList(String textColumnName, boolean  
selectDistinct)
```


Throws:

SQLException

getTableName

protected `String` getTableName()

getTableNameComp

protected `String` getTableNameComp()

insert

protected `void` insert(`String` columnNames, `String` columnValues)

Throws:

SQLException

insertComp

protected `void` insertComp(`String` columnNames, `String` columnValues)

Throws:

SQLException

rollback

public `void` rollback()

Throws:

SQLException

select

protected `CustResultSet` select(`String` query)

Throws:

SQLException

selectList

protected `Vector` selectList(`String` columnNames, `String` whereClause, `String` orderBy)

Throws:

SQLException

selectList

protected `Vector` selectList(`String` columnNames, `String` whereClause)

Throws:

SQLException

setTableName

protected `void` setTableName(`String` tableName)

setTableNameComp

protected `void` setTableNameComp(`String` tableNameComp)

update

protected `void` update(`String` columnNames, `String` columnValues, `String` whereClause)

Throws:

SQLException

10.3.6.2.3.

Class *DMServer*

package: **server**

```
public class DMServer
```

Class DMServer

Description For the purpose to manage connectivity to the database,

<i>Field Summary</i>	
private <code>KeyTable</code>	keyTable
<code>Connection</code>	systemConnection
<code>Connection</code>	userConnection

<i>Constructor Summary</i>	
public	DMServer () Defines the driver name and database name

<i>Method Summary</i>	
public int	getNextKey (String columnName, String tableName) Method that returns the next key (internal identifier used by the system)
public Connection	getUserConnection ()

Field Detail

keyTable

private KeyTable keyTable = null

systemConnection

Connection systemConnection = null

userConnection

Connection userConnection = null

Constructor Detail

DMServer

public DMServer ()

Defines the driver name and database name

Throws:

SQLException

ClassNotFoundException

Method Detail**getNextKey**

```
public int getNextKey(String columnName, String tableName)
```

Method that returns the next key (internal identifier used by the system)

Throws:

SQLException

getUserConnection

```
public Connection getUserConnection()
```

10.3.6.2.4.**Class KeyTable**

package: **server**

```
public class KeyTable
```

Class KeyTable

Description This is used to make primary key generation relatively simple and independent of a DBMS solution.

Field Summary

	<code>Connection</code>	<code>connection</code>
--	-------------------------	-------------------------

Constructor Summary

	<code>public</code>	<code>KeyTable (Connection dbConnection)</code>
--	---------------------	---

Method Summary

	<code>public int</code>	<code>getNextKey (String columnName, String tableName)</code>
--	-------------------------	---

Field Detail**connection**

`Connection` connection = null

Constructor Detail**KeyTable**

```
public KeyTable(Connection dbConnection)
```

Method Detail**getNextKey**

```
public int getNextKey(String columnName, String tableName)
```

Throws:

SQLException

10.3.7.**Package user interface*****Class Diagrams***

diagram user_interface

Interaction Diagrams

diagram Initalizing GUI

Classes

class GLOFrame

class GLOFrame_AboutBox

class GuiLoadLoApp

class TableDataModel

**10.3.7.1.
Class Diagrams**

**10.3.7.1.1.
Class Diagram user interface**

package: user_interface

GLOFrame_AboutBox

GuiLoadLoApp

TableDataModel

10.3.7.2.

Class Detail

10.3.7.2.1.

Class *GLOFrame*

package: **user_interface**

java.lang.Object

|

+-- java.awt.Component

|

+-- java.awt.Container

|

+-- java.awt.Window

|

+-- java.awt.Frame

|

+-- javax.swing.JFrame

|

+--

user_interface.GLOFrame

```
public class GLOFrame
```

Extends:

[javax.swing.JFrame](#)

Class GLOFrame

Description This is a sample interface to load the learning object basic info files and also to load learning object components and they execution order within a learning objective

<i>Field Summary</i>	
final static int	ADDED_DETAIL
BorderLayout	borderLayout1
BorderLayout	borderLayout2
BorderLayout	borderLayout3
private GuiLoad	currentGuiLoad Problem Domain Object
SimpleDateFormat	dateFormat Format helper
private DMServer	dmServer Database handle
GridLayout	gridLayout1
ImageIcon	imageHelp
JButton	jButtonCancel
JButton	jButtonLinkComp
JButton	jButtonLoadBasic
private JButton	jButtonLoadComp
private JButton	jButtonNewCompSet
JButton	jButtonNewInfo
JLabel	jLabel1
private JLabel	jLabel10
private JLabel	jLabel11
private JLabel	jLabel12
private JLabel	jLabel13
private JLabel	jLabel14
private JLabel	jLabel15
private JLabel	jLabel16
private JLabel	jLabel17
private JLabel	jLabel18
JLabel	jLabel2
JLabel	jLabel3

<i>Field Summary</i>	
private JLabel	jLabel4
private JLabel	jLabel5
JLabel	jLabel6
private JLabel	jLabel7
JLabel	jLabel8
private JLabel	jLabel9
JLabel	jLabelTimeStamp
JPanel	jPanel1
JPanel	jPanelBasicInfo
JPanel	jPanelComp
JPanel	jPanelDetails
JScrollPane	jScrollPaneDetails
JTable	jTableCompDetails
JTextField	jTextFieldAssFile
JTextField	jTextFieldCompId
JTextField	jTextFieldExerFile
private JTextField	jTextFieldLOID
JTextField	jTextFieldLomFile
private JTextField	jTextFieldOrd
private JTextField	jTextFieldOtherTechFile
private JTextField	jTextFieldTechAudioFile
private JTextField	jTextFieldTechImageFile
JTextField	jTextFieldTechVideoFile
JMenuBar	menuBar1
JMenu	menuFile
JMenuItem	menuFileExit
JMenu	menuHelp
JMenuItem	menuHelpAbout
JMenu	menuView
JMenuItem	menuViewListAll

<i>Field Summary</i>	
<code>NumberFormat</code>	numberFormat Format helper
<code>final static int</code>	REMOVED_DETAIL
<code>JLabel</code>	statusBar
<code>JToolBar</code>	toolBar
<code>public static boolean</code>	USE_DB Use this to trigger the database on/off 'false' turns DB functions OFF 'true' turns DB functions ON

<i>Constructor Summary</i>	
<code>public</code>	GLOFrame () Construct the frame

<i>Method Summary</i>	
<code>private void</code>	addDetailToTable () Update the table model to update the table UI.
<code>private void</code>	cancelCarga () User has decided to cancel a load.
<code>private void</code>	collectBasic () Collect the basic info associated with a LO
<code>private void</code>	collectComp () Collect a component set
<code>private boolean</code>	emptyString (String string) Helper function to verify is a string is null.

<i>Method Summary</i>	
public void	fileExit_actionPerformed (ActionEvent e) File Exit action performed
public void	helpAbout_actionPerformed (ActionEvent e) Help About action performed
private void	initButtons () Keep buttons properly grayed out at the main start.
private void	initButtonsBasic () Keep buttons properly grayed out at the start of basic info files data-entry.
private void	initButtonsCompSet () Keep buttons properly grayed out at the start of component set data-entry.
private void	initDatabase () Verify if database odbc was properly defined
private void	initDetailTable () Init components table
private void	initialize () Component initialization
void	jButtonCancel_actionPerformed (ActionEvent e)
void	jButtonLinkComp_actionPerformed (ActionEvent e)
void	jButtonLoadBasic_actionPerformed (ActionEvent e)

<i>Method Summary</i>	
<code>void</code>	<code>jButtonLoadComp_actionPerformed(ActionEvent e)</code>
<code>void</code>	<code>jButtonNewCompSet_actionPerformed(ActionEvent e)</code>
<code>void</code>	<code>jButtonNewInfo_actionPerformed(ActionEvent e)</code>
<code>private void</code>	<code>listAllTableSide()</code> List all titles.
<code>private void</code>	<code>loadComp()</code> Load the component's set collected
<code>private void</code>	<code>loadGUIWithPD()</code> Load the UI widgets with "data"
<code>private void</code>	<code>loadGUIWithPDBasic()</code> Load the UI widgets with "data"
<code>private void</code>	<code>loadGUIWithPDCompSet()</code> Load the UI widgets with "data"
<code>private void</code>	<code>newBasic()</code> Start a new load of basic information.
<code>private void</code>	<code>newCompSet()</code> Start a new component set data entry
<code>private int</code>	<code>preCancelCheck()</code> Verify if a cancel operation was issued during a data entry operation
<code>public void</code>	<code>viewListAll_actionPerformed(ActionEvent e)</code> View ListAll action performed

Field Detail**ADDED_DETAIL**

```
final static int ADDED_DETAIL = 0
```

borderLayout1

```
BorderLayout borderLayout1 = new BorderLayout()
```

borderLayout2

```
BorderLayout borderLayout2 = new BorderLayout()
```

borderLayout3

```
BorderLayout borderLayout3 = new BorderLayout()
```

currentGuiLoad

```
private GuiLoad currentGuiLoad = new GuiLoad()
```

Problem Domain Object

dateFormat

```
SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy.MM.dd" )
```

Format helper

dmServer

```
private DMServer dmServer = null
```

Database handle

gridLayout1

```
GridLayout gridLayout1 = new GridLayout()
```

imageHelp

```
ImageIcon imageHelp
```

jButtonCancel

```
JButton jButtonCancel = new JButton()
```

jButtonLinkComp

```
JButton jButtonLinkComp = new JButton()
```

jButtonLoadBasic

```
JButton jButtonLoadBasic = new JButton()
```

jButtonLoadComp

```
private JButton jButtonLoadComp = new JButton()
```

jButtonNewCompSet

```
private JButton jButtonNewCompSet = new JButton()
```

jButtonNewInfo

```
JButton jButtonNewInfo = new JButton()
```

jLabel1

```
JLabel jLabel1 = new JLabel()
```

jLabel10

```
private JLabel jLabel10 = new JLabel()
```

jLabel11

```
private JLabel jLabel11 = new JLabel()
```

jLabel12

```
private JLabel jLabel12 = new JLabel()
```

jLabel13

```
private JLabel jLabel13 = new JLabel()
```

jLabel14

```
private JLabel jLabel14 = new JLabel()
```

jLabel15

```
private JLabel jLabel15 = new JLabel()
```

jLabel16

```
private JLabel jLabel16 = new JLabel()
```

jLabel17

```
private JLabel jLabel17 = new JLabel()
```

jLabel18

```
private JLabel jLabel18 = new JLabel()
```

jLabel2

```
JLabel jLabel2 = new JLabel()
```

jLabel3

```
JLabel jLabel3 = new JLabel()
```

jLabel4

```
private JLabel jLabel4 = new JLabel()
```

jLabel5

```
private JLabel jLabel5 = new JLabel()
```

jLabel6

```
JLabel jLabel6 = new JLabel()
```

jLabel7

```
private JLabel jLabel7 = new JLabel()
```

jLabel8

```
JLabel jLabel8 = new JLabel()
```

jLabel9

```
private JLabel jLabel9 = new JLabel()
```

jLabelTimeStamp

```
JLabel jLabelTimeStamp = new JLabel()
```

jPanel1

```
JPanel jPanel1 = new JPanel()
```

jPanelBasicInfo

```
JPanel jPanelBasicInfo = new JPanel()
```

jPanelComp

```
JPanel jPanelComp = new JPanel()
```

jPanelDetails

```
JPanel jPanelDetails = new JPanel(new GridBagLayout())
```

jScrollPaneDetails

```
JScrollPane jScrollPaneDetails = new JScrollPane()
```

jTableCompDetails

```
JTable jTableCompDetails = new JTable()
```

jTextFieldAssFile

```
JTextField jTextFieldAssFile = new JTextField()
```

jTextFieldCompId

```
JTextField jTextFieldCompId = new JTextField()
```

jTextFieldExerFile

```
JTextField jTextFieldExerFile = new JTextField()
```

jTextFieldLOID

```
private JTextField jTextFieldLOID = new JTextField()
```

jTextFieldLomFile

```
JTextField jTextFieldLomFile = new JTextField()
```

jTextFieldOrd

```
private JTextField jTextFieldOrd = new JTextField()
```

jTextFieldOtherTechFile

```
private JTextField jTextFieldOtherTechFile = new JTextField()
```

jTextFieldTechAudioFile

```
private JTextField jTextFieldTechAudioFile = new JTextField()
```

jTextFieldTechImageFile

```
private JTextField jTextFieldTechImageFile = new JTextField()
```

jTextFieldTechVideoFile

```
JTextField jTextFieldTechVideoFile = new JTextField()
```

menuBar1

```
JMenuBar menuBar1 = new JMenuBar()
```

menuFile

```
JMenu menuFile = new JMenu()
```

menuFileExit

```
JMenuItem menuFileExit = new JMenuItem()
```

menuHelp

```
JMenu menuHelp = new JMenu()
```

menuHelpAbout

```
JMenuItem menuHelpAbout = new JMenuItem()
```

menuView

```
JMenu menuView = new JMenu()
```

menuViewListAll

```
JMenuItem menuViewListAll = new JMenuItem()
```

numberFormat

```
NumberFormat numberFormat = NumberFormat.getInstance()
```

Format helper

REMOVED_DETAIL

```
final static int REMOVED_DETAIL = 1
```

statusBar

```
JLabel statusBar = new JLabel()
```

toolBar

```
JToolBar toolBar = new JToolBar()
```

USE_DB

```
public static boolean USE_DB = true
```

Use this to trigger the database on/off 'false' turns DB functions OFF
'true' turns DB functions ON

Constructor Detail**GLOFrame**

```
public GLOFrame ()
```

Construct the frame

Method Detail**addDetailToTable**

```
private void addDetailToTable ()
```

Update the table model to update the table UI.

cancelCarga

```
private void cancelCarga ()
```

User has decided to cancel a load. Start a new load.

collectBasic

```
private void collectBasic ()
```

Collect the basic info associated with a LO

collectComp

```
private void collectComp ()
```

Collect a component set

emptyString

```
private boolean emptyString (String string)
```

Helper function to verify is a string is null.

fileExit_actionPerformed

```
public void fileExit_actionPerformed (ActionEvent e)
```

File | Exit action performed

helpAbout_actionPerformed

```
public void helpAbout_actionPerformed (ActionEvent e)
```

Help | About action performed

initButtons

```
private void initButtons()
```

Keep buttons properly grayed out at the main start.

initButtonsBasic

```
private void initButtonsBasic()
```

Keep buttons properly grayed out at the start of basic info files data-entry.

initButtonsCompSet

```
private void initButtonsCompSet()
```

Keep buttons properly grayed out at the start of component set data-entry.

initDatabase

```
private void initDatabase()
```

Verify if database odbc was properly defined

initDetailTable

```
private void initDetailTable()
```

Init components table

initialize

```
private void initialize()
```

Component initialization

Throws:

Exception

jButtonCancel_actionPerformed

```
void jButtonCancel_actionPerformed(ActionEvent e)
```

jButtonLinkComp_actionPerformed

```
void jButtonLinkComp_actionPerformed(ActionEvent e)
```

jButtonLoadBasic_actionPerformed

```
void jButtonLoadBasic_actionPerformed(ActionEvent e)
```

jButtonLoadComp_actionPerformed

```
void jButtonLoadComp_actionPerformed(ActionEvent e)
```

jButtonNewCompSet_actionPerformed

```
void jButtonNewCompSet_actionPerformed(ActionEvent e)
```

jButtonNewInfo_actionPerformed

```
void jButtonNewInfo_actionPerformed(ActionEvent e)
```

listAllTableSide

```
private void listAllTableSide()
```

List all titles.

loadComp

```
private void loadComp()
```

Load the component's set collected

loadGUIWithPD

```
private void loadGUIWithPD()
```

Load the UI widgets with "data"

loadGUIWithPDBasic

```
private void loadGUIWithPDBasic()
```

Load the UI widgets with "data"

loadGUIWithPDCompSet

```
private void loadGUIWithPDCompSet()
```

Load the UI widgets with "data"

newBasic

```
private void newBasic()
```

Start a new load of basic information.

newCompSet

```
private void newCompSet ()
```

Start a new component set data entry

preCancelCheck

```
private int preCancelCheck ()
```

Verify if a cancel operation was issued during a data entry operation

viewListAll_actionPerformed

```
public void viewListAll_actionPerformed(ActionEvent e)
```

View | ListAll action performed

10.3.7.2.2.

Class *GLOFrame_AboutBox*

package: **user_interface**

```
java.lang.Object
```

```
|
```

```
+-- java.awt.Component
```

```
|
```

```
+-- java.awt.Container
```

```
|
```

```
+-- java.awt.Window
```

```
|
```

```
+-- java.awt.Dialog
```

```
|
```

```
+-- javax.swing.JDialog
```

```
|
```

```
+--
```

user_interface.GLOFrame_AboutBox

```
public class GLOFrame_AboutBox
```

Extends:`javax.swing.JDialog`**Implements:**`java.awt.event.ActionListener`

Class GLOFrame_AboutBox

Description It is a sample presenting how to load basic learning objects information files using DB2/IBM extenders for audio, video, image and xml

<i>Field Summary</i>	
<code>BorderLayout</code>	<code>borderLayout1</code>
<code>BorderLayout</code>	<code>borderLayout2</code>
<code>JButton</code>	<code>button1</code>
<code>String</code>	<code>comments</code>
<code>String</code>	<code>copyright</code>
<code>FlowLayout</code>	<code>flowLayout1</code>
<code>FlowLayout</code>	<code>flowLayout2</code>
<code>GridLayout</code>	<code>gridLayout1</code>
<code>JLabel</code>	<code>imageControl1</code>
<code>ImageIcon</code>	<code>imageIcon</code>
<code>JPanel</code>	<code>insertsPanel1</code>
<code>JPanel</code>	<code>insertsPanel2</code>
<code>JPanel</code>	<code>insertsPanel3</code>
<code>JLabel</code>	<code>label1</code>
<code>JLabel</code>	<code>label2</code>
<code>JLabel</code>	<code>label3</code>
<code>JLabel</code>	<code>label4</code>
<code>JPanel</code>	<code>panel1</code>
<code>JPanel</code>	<code>panel2</code>
<code>String</code>	<code>product</code>
<code>String</code>	<code>version</code>

<i>Constructor Summary</i>	
public	GLOFrame_AboutBox (Frame parent)

<i>Method Summary</i>	
public void	actionPerformed (ActionEvent e)
void	cancel ()
private void	initialize ()

Field Detail

borderLayout1

[BorderLayout](#) borderLayout1 = new BorderLayout()

borderLayout2

[BorderLayout](#) borderLayout2 = new BorderLayout()

button1

[JButton](#) button1 = new JButton()

comments

[String](#) comments = "This program uses DB2 extenders for audio, " +
"video, image and XML"

copyright

[String](#) copyright = "Copyright (c) 2002"

flowLayout1

[FlowLayout](#) flowLayout1 = new FlowLayout()

flowLayout2

[FlowLayout](#) flowLayout2 = new FlowLayout()

gridLayout1

[GridLayout](#) gridLayout1 = new GridLayout()

imageControl1

```
JLabel imageControl1 = new JLabel()
```

imageIcon

```
ImageIcon imageIcon
```

insertsPanel1

```
JPanel insertsPanel1 = new JPanel()
```

insertsPanel2

```
JPanel insertsPanel2 = new JPanel()
```

insertsPanel3

```
JPanel insertsPanel3 = new JPanel()
```

label1

```
JLabel label1 = new JLabel()
```

label2

```
JLabel label2 = new JLabel()
```

label3

```
JLabel label3 = new JLabel()
```

label4

```
JLabel label4 = new JLabel()
```

panel1

```
JPanel panel1 = new JPanel()
```

panel2

```
JPanel panel2 = new JPanel()
```

product

```
String product = "Learning Objects: Loading basic files " +  
                "and associated components"
```

version

```
String version = "Version 1"
```

Constructor Detail**GLOFrame_AboutBox**

```
public GLOFrame_AboutBox(Frame parent)
```

Method Detail**actionPerformed**

```
public void actionPerformed(ActionEvent e)
```

cancel

```
void cancel()
```

initialize

```
private void initialize()
```

Throws:

Exception

10.3.7.2.3.**Class *GuiLoadLoApp***

package: **user_interface**

```
public class GuiLoadLoApp
```

Class **GuiLoadApp**

Description This is the start class for the application

Field Summary

<code>boolean</code>	packFrame
----------------------	------------------

Constructor Summary

<code>public</code>	GuiLoadLoApp ()
---------------------	------------------------

Method Summary

<code>public static void</code>	main (String args)
---------------------------------	---------------------------

Field Detail**packFrame**

```
boolean packFrame = true
```

Constructor Detail**GuiLoadLoApp**

```
public GuiLoadLoApp()
```

Method Detail**main**

```
public static void main(String args)
```

10.3.7.2.4.**Class *TableModel***

```
package: user_interface
```

```
java.lang.Object
```

```
|
```

```
+- javax.swing.table.AbstractTableModel
```

```
|
```

```
+- user_interface.TableDataModel
```

```
public class TableDataModel
```

Extends:

```
javax.swing.table.AbstractTableModel
```

Class *TableModel*

Description Create a model of the data will used in tabular forms

Field Summary

private final String	colNames Table Column Header
--------------------------------------	--

<i>Field Summary</i>	
private <code>GuiLoad</code>	currentGuiLoad Class that will be used in tabular format

<i>Constructor Summary</i>	
public	TableModel (<code>GuiLoad</code> currentGuiLoad)

<i>Method Summary</i>	
public <code>void</code>	dataChanged (<code>int</code> changeType, <code>int</code> index) Update the table with the changed data model
public <code>Class</code>	getColumnClass (<code>int</code> c) JTable uses this method to determine the default render/ editor for each cell to improve the display.
public <code>int</code>	getColumnCount () This method always needs to be implemented.
public <code>String</code>	getColumnName (<code>int</code> column) Return the corresponding string for a column number
public <code>int</code>	getRowCount () This method always needs to be implemented.
public <code>Object</code>	getValueAt (<code>int</code> row, <code>int</code> col) This method always needs to be implemented.

<i>Method Summary</i>	
public boolean	isCellEditable (int row, int col) Set all cells non editable

Field Detail

colNames

```
private final String colNames = { "Sequential" , "Execution Order" , "Component ID" }
```

Table Column Header

currentGuiLoad

```
private GuiLoad currentGuiLoad
```

Class that will be used in tabular format

Constructor Detail

TableModel

```
public TableDataModel(GuiLoad currentGuiLoad)
```

Method Detail

dataChanged

```
public void dataChanged(int changeType, int index)
```

Update the table with the changed data model

getColumnClass

```
public Class getColumnClass(int c)
```

JTable uses this method to determine the default render/ editor for each cell to improve the display.

getColumnCount

```
public int getColumnCount()
```

This method always needs to be implemented.

getColumnName

```
public String getColumnName(int column)
```

Return the corresponding string for a column number

getRowCount

```
public int getRowCount()
```

This method always needs to be implemented.

getValueAt

```
public Object getValueAt(int row, int col)
```

This method always needs to be implemented.

isCellEditable

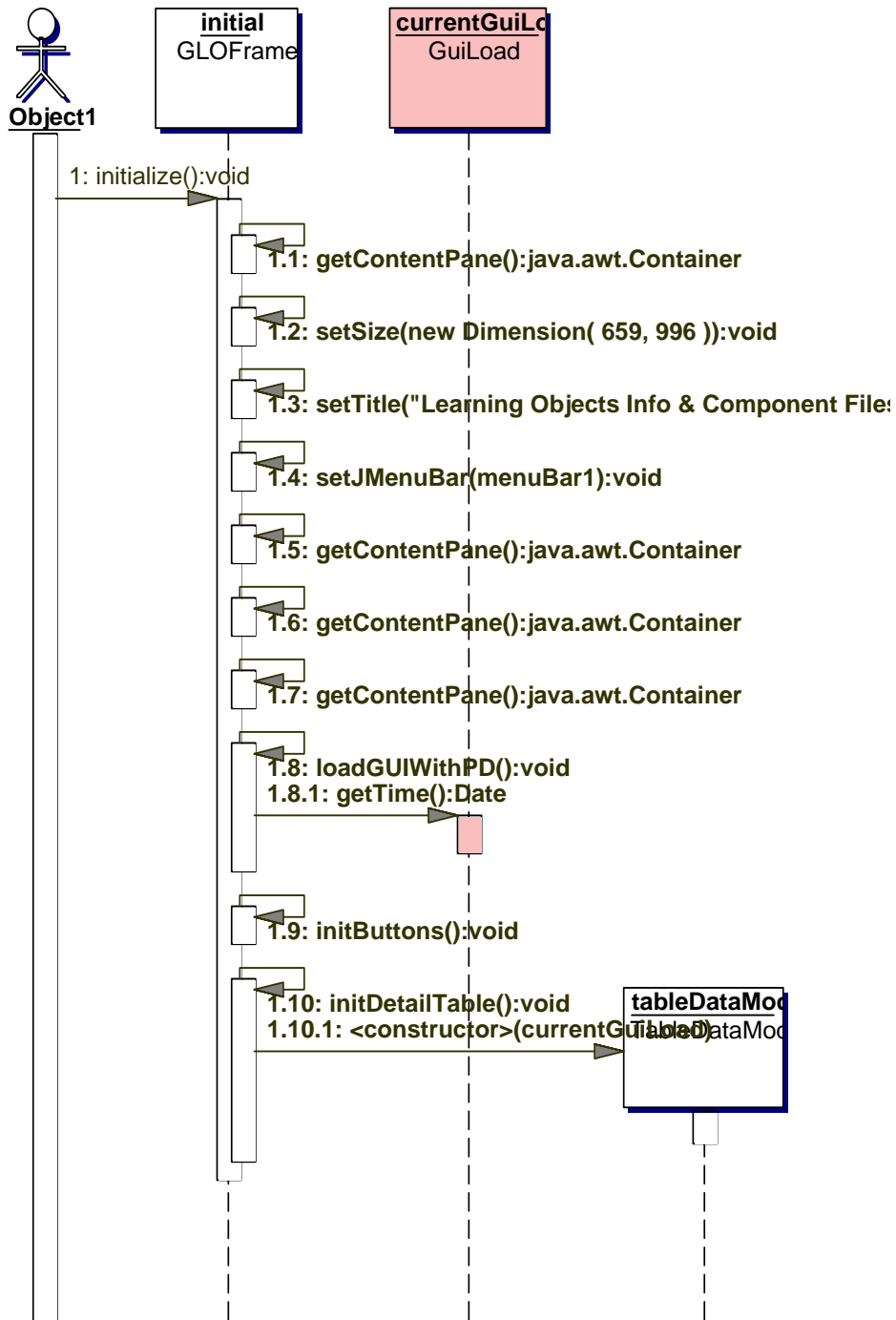
```
public boolean isCellEditable(int row, int col)
```

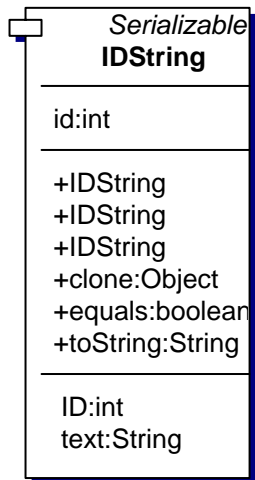
Set all cells non editable

10.3.7.3. Interaction Diagrams

10.3.7.3.1. Sequence Diagram *Initilizing GUI*

package: **user_interface**



10.3.8.**Package util***Class Diagrams***diagram util***Classes***class IDString****10.3.8.1.****Class Diagrams****10.3.8.1.1.****Class Diagrams util****package:** user_interface**package:** util*Class Nodes***IDString****10.3.8.2.****Class Detail**

**10.3.8.2.1.
Class *IDString***

package: **util**

```
public class IDString
```

Implements:

`java.io.Serializable`

Class *IDString*

Description This is used as aid-in box to return the database index and the name string for a given object in the database table. It is very useful when doing retrieves of complex objects to populate a drop-down list box of choices, for example.

<i>Field Summary</i>	
<code>int</code>	id Object ID number.
<code>String</code>	text This is the Object's name.

<i>Constructor Summary</i>	
<code>public</code>	IDString ()
<code>public</code>	IDString (int id, String text)
<code>public</code>	IDString (IDString data)

<i>Method Summary</i>	
<code>public Object</code>	clone ()
<code>public boolean</code>	equals (Object obj)
<code>public int</code>	getID ()
<code>public String</code>	getText ()
<code>public void</code>	setID (int value)
<code>public void</code>	setText (String value)
<code>public String</code>	toString ()

Field Detail

id

```
int id = 0
```

Object ID number.

Defaults to 0 for a new object not yet in the database. This '0' is used by persistence mechanism to know whether an object is new, and needs to be inserted, or it already had been included and now it will updated.

text

```
String text = ""
```

This is the Object's name.

Constructor Detail

IDString

```
public IDString()
```

IDString

```
public IDString(int id, String text)
```

IDString

```
public IDString(IDString data)
```

Method Detail

clone

```
public Object clone()
```

equals

```
public boolean equals(Object obj)
```

getID

```
public int getID()
```

getText

```
public String getText ()
```

setID

```
public void setID(int value)
```

setText

```
public void setText (String value)
```

toString

```
public String toString()
```

10.4. Exemplo de Macro Net.Data

A seguir é apresentado um exemplo de codificação Net.Data.

```

%{ ----- % }
%{ ----- % }
%{ Definitions section % }
%{ ----- % }
%define{
DATABASE="LO_DB_FD"
SHOWSQL="yes"
LOGIN="db2admin"
PASSWORD="db2kpl"
% }
%{ ----- % }
%{ SQL functions % }
%{ ----- % }
%function (DTW_SQL) startHereSQL(){
select id, lo_title, lo_description from db2admin.fv_lo_side_tab
%REPORT{
<table border="2" bgcolor="#b1b1b1">
<tr><th>Id <th> Title <th> Description </tr>
%ROW{ <tr><td><a href="addThumbs?id=$(V_id)">$(V_id)</a> <td> $(V_lo_title)
<td> $(V_lo_description) </tr>
% }
</table>
% }
% }
%function (DTW_SQL) addThumbsSQL(){
select
db2xml.extractVarchar (LO_XML,'/lom/general/title/langstring'),
db2xml.extractVarchar (LO_XML,'/lom/general/description/langstring') ,
db2xml.extractVarchar (LO_XML,'/lom/rights/description/langstring') ,
cast(mmdbsys.thumbnail_i(lo_image) as blob(10000)),
cast(mmdbsys.thumbnail_v(lo_video) as blob(10000)),
mmdbsys.comment(lo_audio), id
from db2admin.x_lo_tab
where id = '$(id)'

```

```

%REPORT{
<h2>$(V1)</h2>
<h4>Description:</h4> <blockquote>$(V2) </blockquote>
<h4>Copyrights:</h4> <blockquote>$(V3) </blockquote>
<a href="showImage?id=$(V_id)"></a>
<a href="getVideo?id=$(V_id)"></a>
<a href="getAudio?id=$(V_id)&filename=$V6">[Listen]</a>
% }
% }
%function (DTW_SQL) showImageSQL(){
select      cast(mmdbsys.content_i(lo_image,      'GIF')      as      blob(150000)),
mmdbsys.format_i(lo_image)
from db2admin.x_lo_tab
where id = '$(id)'
%REPORT{
%ROW{  <br><br><b>Original image format: $(V2)</b>% }
% }
% }
%function (DTW_SQL) showVideoSQL(){
select      mmdbsys.comment(lo_video),      mmdbsys.content_v(lo_video),
mmdbsys.comment(lo_video), 1),
mmdbsys.format_v(lo_video)
from db2admin.x_lo_tab
where id = '$(id)'
%REPORT{
%ROW{ <a href="/tmplobs/$(V1)"><i><b> Play Video Clip</b></i></a>
<br><br><b>Format: $(V3)
% }
% }
% }
%function (DTW_SQL) showAudioSQL(){
select      mmdbsys.comment(lo_audio),      mmdbsys.content(lo_audio),
mmdbsys.comment(lo_audio), 1),
mmdbsys.format(lo_audio)
from db2admin.x_lo_tab
where id = '$(id)'
%REPORT{
%ROW{ <a href="/tmplobs/$(V1) "><i><b>Play Audio Clip</b></i></a>
<br><br><b>Format: $(V3)</b>
% }
% }

```

```

% }
% }
% { ----- % }
% { HTML sections % }
% { -----% }
% }
%HTML(startHere){
<html>
<head><title>LO_DB: Simple Row Listing</title></head>
<body bgcolor="#ffffff">
<br>@startHereSQL()
<br><b>Click <a href="addThumbs"><i>here</i></a> to display thumbnails
and links to image/audio/video data.</b>
</body>
</html>
% }
%HTML(addThumbs){
<html>
<head><title>Lo Description</title></head>
<body bgcolor="#ffffff">
<font color="#3300ff" size="3"><b>This page shows some LO details and adds image
thumbnails
and links to display the multimedia content of the database.
</b></font> @addThumbsSQL()
<br><b>Click <a href="startHere"><i>here</i></a> to go back to the first page.</b>
</body>
</html>
% }
%HTML(showImage){
<html>
<head><title>LO_DB: Image contained in a Learning Object $(id)</title></head>
<body bgcolor="#ffffff">
<font color="#3300ff" size="3"><b>From this page, you can view an image:
</b></font><br><br>
<table width="400" border="2" bgcolor="#b1b1b1" cellpadding="5">
<tr><td align=center> @showImageSQL()
<tr><td align=center> <b>Id: $(id)</b>
</table>
<br><b>Go <a href="addThumbs?id=$(id)"><i>back</i></a>.</b>
</body>

```

```

</html>
% }
%HTML(getVideo){
<html>
<head><title>LO_DB: Video clip contained in a Learning Object $(id)</title></head>
<body bgcolor="#ffffff">
<font color="#3300ff" size="3"><b>From this page, you can view a video:
</b></font><br><br>
<table width="400" border="2" bgcolor="#b1b1b1" cellpadding="5">
<tr><td align=center> @showVideoSQL()
<tr><td align=center> <b>Id: $(id)</b>
</table>
<br><b>Go <a href="addThumbs?id=$(id)"><i>back</i></a>.</b>
</body>
</html>
% }
%HTML(getAudio){
<html>
<head><title>LO_DB: Audio clip contained in a Learning Object $(id)</title></head>
<body bgcolor="#ffffff">
<font color="#3300ff" size="3"><b>From this page, you can listen to an audio:
</b></font><br><br>
<table width="400" border="2" bgcolor="#b1b1b1" cellpadding="5">
<tr><td align=center> @showAudioSQL()
<tr><td align=center> <b>Id: $(id)</b>
</table>
<br><b>Go <a href="addThumbs?id=$(id)"><i>back</i></a>.</b>
</body>
</html>
% }

```

Figura 49. Macro Net.Data

10.5. Modelo LOM

10.5.1. Descrição da Estrutura

O elemento **<lom>** engloba o conjunto total das descrições. Êle é decomposto nos seguintes elementos:

1. Elemento **<general>**

Forma um conjunto único contendo as descrições gerais sobre o objeto de aprendizado. Este elemento subdivide-se nos seguintes elementos:

1.1. Elemento **<identifier>**

Para conter a identificação do objeto de aprendizado, segundo um esquema de identificação, como ISBN, por exemplo:

x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475

1.2. Elemento **<title>**

Para conter o nome atribuído ao objeto de aprendizado, por exemplo:
Microsoft SQL Server 7.0: Implementing a Database - Part 1

1.3. Elemento **<catalogentry>**

Forma uma lista com até 10 valores contendo informações para catalogação. A lista é subdividida em dois elementos:

1.3.1. Elemento **<catalog>**

É uma cadeia de até 1000 caracteres contendo o nome ou identificador do esquema de identificação, por exemplo:

“ARIADNE”, “ISBN”, “URI”, por exemplo:

<http://www.netg.com/catalog1.html>

1.3.2. Elemento **<entry>**

Valor do identificador, segundo o esquema especificado, por exemplo:

72475

1.4. Elemento **<language>**

Compreende uma lista de até 10 itens descrevendo o código do idioma, em que é expresso o conteúdo do objeto de aprendizado, segundo a norma ISO 639.1988 e código do país especificado conforme a ISO

3166.1977. A utilização sem o código do país indica a audiência alvo.

Exemplos: “pt-BR”, “en”;

1.5. Elemento <description>

Lista de até 10 itens de até 2000 caracteres descrevendo o conteúdo do objeto de aprendizado, por exemplo:

’This is the first course in a five part series that will provide students with the knowledge to implement a database solution with Microsoft SQL Server 7.0”

1.6. Elemento <keyword>

Lista de até 10 itens de até 1000 caracteres relacionando as palavras chave associadas ao objeto de aprendizado, por exemplo: “Database”;

1.7. Elemento <coverage>

Lista de até 10 itens de até 1000 caracteres identificando as abrangências temporal, culturais, geográficas ou regionais do objeto de aprendizado;

1.8. Elemento <structure>

Apresenta o grau de composição e o tipo de relacionamento entre os componentes do objeto de aprendizado, expresso pelas palavras chaves restritas constantes do conjunto: “collection”, “mixed”, “linear”, “hierarquical”, “networked”, “branched”, “parceled”, “atomic”.

1.9. Elemento <aggregationlevel>

Descreve o tamanho funcional do objeto de aprendizado, através dos valores de 1 a 4:

Nível 1: menor nível de agregação, indicando conteúdo simples ou elementar;

Nível 2: refere-se a uma coleção de elementos atômicos, como uma página HTML com figuras ou uma lição;

Nível 3: indica uma coleção de recursos do nível 1, como por exemplo uma coleção de páginas HTML, com página índice permitindo a navegação pelas demais páginas;

Nível 4 indica o mais alto nível de agregação, como por exemplo, um curso.

2. Elemento <lifecycle>

Uma instância única descrevendo o histórico e o estado atual do objeto de aprendizado expressa através dos seguintes elementos:

2.1. Elemento <version>

Indica a edição do objeto de aprendizado.

2.2. Elemento <status>

Indica a condição de edição do objeto de aprendizado, expressa por meio de uma das seguintes opções: “draft”, “final”, “revised”, “unavailable”

2.3. Elemento <contribute>

Lista de até 30 itens descrevendo pessoas ou organizações que contribuíram para elaborar o objeto de aprendizado:

2.3.1. Elemento <role>

Descreve o tipo de contribuição, através de uma das palavras sugeridas no conjunto: author, Publisher, unknown, initiator, terminator, validator, editor, graphical designer, technical implementer, content provider, technical validator, educational validator, script writer, instructional designer;

2.3.2. Elemento <entity>

Lista de até 40 itens relacionando as entidades envolvidas na elaboração do objeto de aprendizado

2.3.3. Elemento <date>

Data da contribuição.

3. Elemento <metametadata>

Contém características associadas às descrições sobre elementos:

3.1. Elemento <identifier>

Contém o rótulo utilizado para o elemento

3.2. Elemento <catalogentry>

Lista de até 10 itens designando a instância de metadados

3.2.1. Elemento <catalog>

Fonte a que se refere a informação

3.2.2. Elemento <entry>

Valor de referencia a que se refere a informação

3.3. Elemento <contribute>

Lista de até 10 itens relacionando as pessoas e organizações que contribuíram para elaborar os metadados

3.3.1. Elemento <role>

Descreve o tipo de contribuição, através de uma das palavras sugeridas no conjunto: creator, validator;

3.3.2. Elemento <entity>

Lista de até 10 itens relacionando as entidades participantes da elaboração dos metadados;

3.3.3. Elemento <date>

Data da contribuição

3.4. Elemento <metadatascheme>

Relação de até 10 itens identificando a estrutura de metadados, por exemplo: LOM-1.0

3.5. Elemento <language>

Linguagem em que está descrita a instância de metadados

4. Elemento <technical>

Contem as descrições relativas aos aspectos técnicos do objeto de aprendizado:

4.1. Elemento <format>

Lista de até 40 itens descrevendo o objeto de aprendizado, por exemplo: vídeo/mpeg, texto/html

4.2. Elemento <size>

Descreve o tamanho do objeto de aprendizado utilizando como unidade byte.

4.3. Elemento <location>

Lista de até 10 itens descrevendo onde pode ser localizado o objeto de aprendizado

4.4. Elemento <requirements>

Lista de até 10 itens descrevendo os requisitos para acessar o objeto de aprendizado

4.4.1. Elemento <type>

Tipo de requisito, compreendendo as palavras sugeridas: operating system, browser

4.4.2. Elemento <name>

Nome do requisito, conforme lista sugerida: PC-DOS, MS-Windows, Maços, Unix, Multi-OS, Other, None, Any, Opera, Microsoft Internet Explorer, Netscape Communicator;

4.4.3. Elemento <minimumversion>

Versão mínima necessária

4.4.4. Elemento <maximumversion>

Versão máxima necessária

4.5. Elemento <installationremarks>

Descrição sobre como instalar o objeto de aprendizado

4.6. Elemento <otherplatformrequirements>

Informação sobre outros requisitos de programas e equipamentos necessários

4.7. Elemento <duration>

Tempo em segundos quando executado na velocidade prevista

5. Elemento <educational>

Contem características educacionais ou pedagógicas do objeto de aprendizado

5.1. Elemento <interactivitytype>

Descreve o tipo de interatividade suportado pelo objeto de aprendizado, expresso por uma das palavras: active, expositive, mixed, undefined

5.2. Elemento <learningresourcetype>

Lista descrevendo o tipo de recurso empregado no objeto de aprendizado selecionada entre as palavras sugeridas: exercise, simulation, questionnaire, diagram, figure, graph, index, slide, table, narrative text, exam, experiment, problemstatement, selfassessment

5.3. Elemento <interactivitylevel>

Descrição do grau de interatividade entre o usuário final e o objeto de aprendizado, expresso por meio das palavras sugeridas: very low, low, médium, high, very high

5.4. Elemento <semanticdensity>

Avaliação subjetiva do grau de utilidade do objeto de aprendizado comparada com a sua duração, expressa pelas palavras sugeridas: very low, low, médium, high, very high

5.5. Elemento <intendedenduserrole>

Lista de até 4 itens descrevendo o usuário do objeto de aprendizado, escolhido entre as palavras: teacher, author, learner, manager

5.6. Elemento <context>

Lista de até 10 itens descrevendo o ambiente usual onde o objeto de aprendizado será utilizado

5.7. Elemento <typicalagerange>

Lista de até 5 itens descrevendo a idade usual do usuário

5.8. Elemento <difficulty>

Descrição da dificuldade para trabalhar com o objeto de aprendizado submetido para a audiência prevista, expressa através de uma das palavras: very easy, easy, médium, difficult, very difficult

5.9. Elemento <typicallearningtime>

Tempo aproximado para trabalhar com o objeto de aprendizado

5.10. Elemento <description>

Comentários sobre como utilizar o objeto de aprendizado

5.11. Elemento <language>

Idioma do usuário

6. Elemento <rights>

Contem elementos descrevendo as condições para utilização

6.1. Elemento <cost>

Identifica se existe ou não custo associado à utilização do objeto de aprendizado expresso através das palavras: yes, no

6.2. Elemento <copyrightandotherrestrictions>

Identifica se existe ou não direitos autorais ou outras restrições aplicáveis à utilização do objeto de aprendizado expresso através das palavras: yes, no

6.3. Elemento <description>

Comentários sobre as condições de utilização do objeto de aprendizado

7. Elemento <relation>

Lista de até 100 itens que descreve o relacionamento deste objeto de aprendizado com outros objetos

7.1. Elemento <kind>

Natureza do relacionamento

7.2. Elemento <resource>

Identificação do recurso associado nesta relação

7.2.1. Elemento <identifier>

Identificação unívoca do recurso associado

7.2.2. Elemento <description>

Descrição do outro recurso

7.2.3. Elemento <catalogentry>

Descrição do outro recurso

8. Elemento <annotation>

Lista de até 30 itens que descreve anotações de carácter educational associadas ao objeto de aprendizagem

8.1. Elemento <person>

Identificação de quem fez anotações

8.2. Elemento <date>

Data em que as anotações foram criadas

8.3. Elemento <description>

Compreende o conteúdo da anotação

9. Elemento <classification>

Lista de até 40 itens que descreve aspectos classificatórios em termos de taxonomia

9.1. Elemento <purpose>

Características do objeto de aprendizagem com algumas palavras sugeridas: discipline, idea, prerequisite, educational objective, accessibility restrictions, educational level, skill level, security level

9.2. Elemento <taxonpath>

Caminho taxonomico segundo a classificação

9.2.1. Elemento <source>

Identificação de uma taxonomia, por exemplo: ACM, MESH, ARIADNE

9.2.2. Elemento <taxon>

Lista de até 15 itens descrevendo a classificação, por exemplo: física/acústica/instrumento/estetoscópio, medicina/diagnóstico/instrumento/estetoscópio

9.2.2.1. Elemento <id>

Identificador taxonomico

9.2.2.2. Elemento <entry>

Nome taxonomico ou rótulo

9.3. Elemento <description>

Descrição do objeto de aprendizado em relação ao propósito estabelecido

9.4. Elemento <keyword>

Lista de até 40 itens relacionando as palavras chave que descrevam o objetivo de aprendizado em relação à finalidade proposta

10.5.2. Descrição em Diagramas

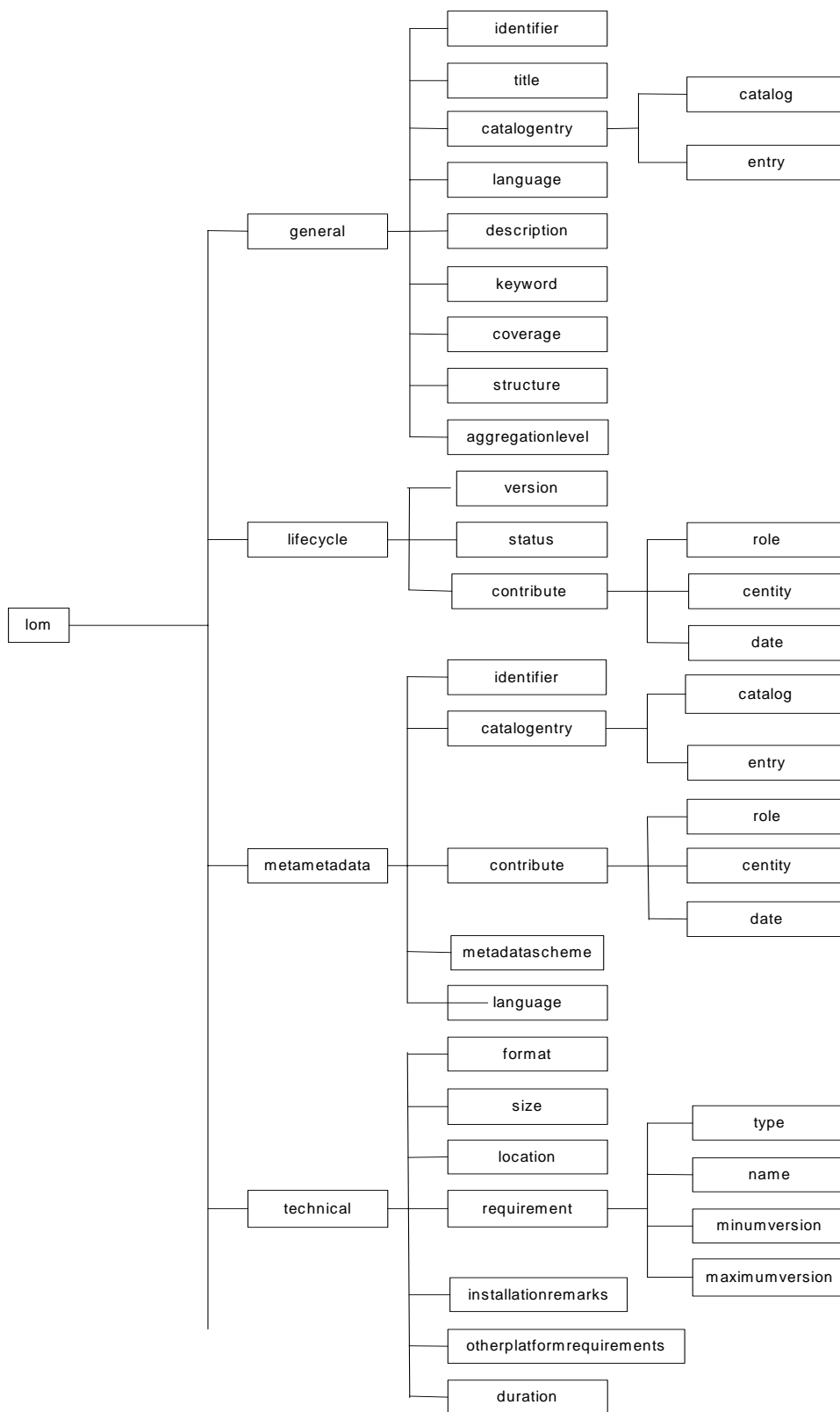


Figura 50. Estrutura do LOM (1/2)

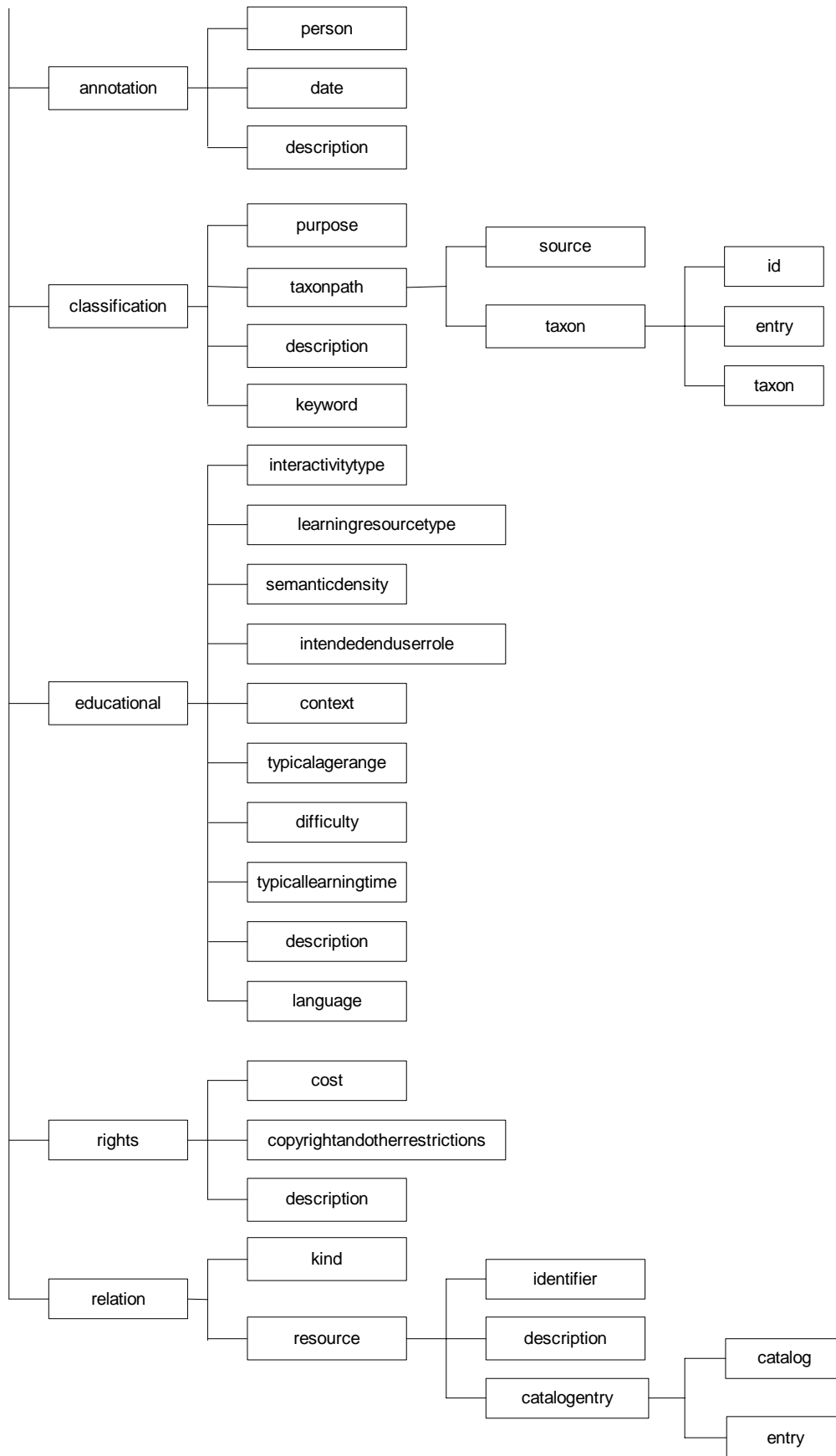


Figura 50. Estrutura do LOM (2/2)

10.5.3. Exemplo Completo de Uso

Um exemplo de descrição, em XML, de um objeto de aprendizado, segundo o padrão IMS, é apresentado a seguir. Este exemplo foi extraído do arquivo MD_NETg_em_US_72475.xml, de metadav1p2p2ex, acessado em <http://www.imsproject.org>.

Este exemplo apresenta a descrição de um curso para implementar banco de dados, utilizando o produto Microsoft SQL Server.

O exemplo apresenta um cabeçalho de identificação e a seguir pode ser decomposto conforme a estrutura do LOM, observando-se a freqüente utilização múltipla dos itens visando enriquecer a descrição.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v3.5 (http://www.xmlspy.com) by Boyd W Nielsen (NETg) -->
<lom xmlns="http://www.imsglobal.org/xsd/imsmid_v1p2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmid_v1p2 imsmid_v1p2p2.xsd">
  <general>
    <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475</identifier>
    <title>
      <langstring xml:lang="en-US">Microsoft SQL Server 7.0: Implementing a Database -
Part 1</langstring>
    </title>
    <catalogentry>
      <catalog>http://www.netg.com/catalog1.html</catalog>
      <entry>
        <langstring xml:lang="en-US">72475</langstring>
      </entry>
    </catalogentry>
    <language>en</language>
    <description>
      <langstring xml:lang="en-US">This is the first course in a five part series that will
provide students with the knowledge to implement a database solution with Microsoft SQL Server 7.0.</langstring>
    </description>
    <keyword>
      <langstring xml:lang="en-US">Windows NT</langstring>
    </keyword>
    <keyword>
      <langstring xml:lang="en-US">Microsoft</langstring>
    </keyword>
    <keyword>
      <langstring xml:lang="en-US">Database</langstring>
    </keyword>
```

```

<keyword>
  <langstring xml:lang="en-US">SQL Server 7.0</langstring>
</keyword>
<keyword>
  <langstring xml:lang="en-US">Microsoft BackOffice</langstring>
</keyword>
<structure>
  <source>
    <langstring xml:lang="x-none">LOMv1.0</langstring>
  </source>
  <value>
    <langstring xml:lang="x-none">Collection</langstring>
  </value>
</structure>
<aggregationlevel>
  <source>
    <langstring xml:lang="x-none">LOMv1.0</langstring>
  </source>
  <value>
    <langstring xml:lang="x-none">3</langstring>
  </value>
</aggregationlevel>
</general>
<lifecycle> ←
  <version>
    <langstring xml:lang="en-US">72475-9907-10</langstring>
  </version>
  <status>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Final</langstring>
    </value>
  </status>
  <contribute>
    <role>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">Publisher</langstring>
      </value>
    </role>
    <centity>
      <vcard>BEGIN:VCARD VERSION:2.1 FN:National Education Training
Group, Inc. ORG:National Education Training Group, Inc. ADR;WORK;;;1751 W. Diehl
Road;Naperville;IL;60563;United States of America LABEL;WORKK;ENCODING=QUOTED-PRINTABLE:1751 W.
Diehl Road=0D=0ANaperville, IL 60563=0D=0AUnited States of America REV:20000413T191459Z
END:VCARD</vcard>
    </centity>
  </contribute>
</lifecycle>

```

```

        <date>
            <datetime>1999-11-04</datetime>
            <description>
                <langstring xml:lang="en-US">Last revision</langstring>
            </description>
        </date>
    </contribute>
<contribute>
    <role>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Author</langstring>
        </value>
    </role>
    <centity>
        <vcard>BEGIN:VCARD VERSION:2.1 FN:National Education Training
Group, Inc. ORG:National Education Training Group, Inc. ADR;WORK:;;1751 W. Diehl
Road;Naperville,IL;60563;United States of America LABEL;WORKK;ENCODING=QUOTED-PRINTABLE:1751 W.
Diehl Road=0D=0ANaperville, IL 60563=0D=0AUnited States of America REV:20000413T191459Z
END:VCARD</vcard>
    </centity>
    <date>
        <datetime>1999-11-04</datetime>
        <description>
            <langstring xml:lang="en-US">Last revision</langstring>
        </description>
    </date>
</contribute>
</lifecycle>
<metametadata> ←
    <metadatascheme>IMS Metadata 1.2</metadatascheme>
    <language>en</language>
</metametadata>
<technical> ←
    <format>HTML</format>
    <location>en_US_72475.html</location>
    <requirement>
        <type>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">Browser</langstring>
            </value>
        </type>
    <name>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>

```

```

        <value>
            <langstring xml:lang="x-none">Microsoft Internet
Explorer</langstring>
        </value>
    </name>
    <minimumversion>4</minimumversion>
</requirement>
<requirement>
    <type>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Browser</langstring>
        </value>
    </type>
    <name>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring
Communicator</langstring>                xml:lang="x-none">Netscape
        </value>
    </name>
    <minimumversion>4</minimumversion>
</requirement>
<requirement>
    <type>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Operating System</langstring>
        </value>
    </type>
    <name>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">MS-Windows</langstring>
        </value>
    </name>
    <minimumversion>95</minimumversion>
</requirement>
<requirement>
    <type>
        <source>
            <langstring xml:lang="x-none">User defined</langstring>
        </source>
    </type>

```

```

        <value>
            <langstring xml:lang="x-none">Virtual Machine</langstring>
        </value>
    </type>
    <name>
        <source>
            <langstring xml:lang="x-none">User defined</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Java Virtual Machine</langstring>
        </value>
    </name>
    <minimumversion>1.1</minimumversion>
</requirement>
<otherplatformrequirements>
    <langstring xml:lang="en-US">Sound card, Min. RAM: 16Mb, Video card and display:
at least 800 X 600 pixels x 256 colors</langstring>
</otherplatformrequirements>
</technical>
<educational> ←—————
    <interactivitytype>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Mixed</langstring>
        </value>
    </interactivitytype>
    <learningresourcetype>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Narrative Text</langstring>
        </value>
    </learningresourcetype>
    <learningresourcetype>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Simulation</langstring>
        </value>
    </learningresourcetype>
    <learningresourcetype>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Exam</langstring>
        </value>
    </learningresourcetype>

```

```

</learningresourcetype>
<interactivitylevel>
  <source>
    <langstring xml:lang="x-none">LOMv1.0</langstring>
  </source>
  <value>
    <langstring xml:lang="x-none">high</langstring>
  </value>
</interactivitylevel>
<intendedenduserrole>
  <source>
    <langstring xml:lang="x-none">LOMv1.0</langstring>
  </source>
  <value>
    <langstring xml:lang="x-none">Learner</langstring>
  </value>
</intendedenduserrole>
<description>
  <langstring xml:lang="en-US">This course is designed for IT professionals responsible
for implementing a database solution with Microsoft SQL Server 7.0</langstring>
</description>
</educational>
<rights> ←—————
  <cost>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">yes</langstring>
    </value>
  </cost>
  <copyrightandotherrestrictions>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">yes</langstring>
    </value>
  </copyrightandotherrestrictions>
  <description>
    <langstring xml:lang="en-US">Copyright © 1999 National Education Training
Group, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any
form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information
storage and retrieval system, without permission in writing from National Education Training Group, Inc. Skill Builder is a
registered trademark of National Education Training Group, Inc. Microsoft, SQL Server, MS, MS-DOS, Visual Basic,
BackOffice, Windows and the Windows logo are registered trademarks. Visual C++ and Windows NT are trademarks of
Microsoft Corporation. All other trademarks referenced are the trademark, service mark, or registered trademark of their
respective holders. National Education Training Group, Inc. is not affiliated with any products or vendors mentioned in
this course and its accompanying materials. The software and technology used to implement this course contain trade
secrets that NETg considers to be confidential and proprietary information, and your right to use this material is subject to
the restrictions in the license agreement under which you obtained it.</langstring>

```



```

        </source>
        <value>
            <langstring xml:lang="x-none">IsBasedOn</langstring>
        </value>
    </kind>
    <resource>
        <description>
            <langstring xml:lang="en-US">Microsoft MCSE</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_021</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Overview</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_311</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Unit 3 Overview and
Objectives</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>

```

```

        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_312</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Data Storage
Characteristics</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_313</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Transaction
Logs</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_321</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Creating a
Database</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>

```

```

        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_322</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Database
Options</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_331</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Managing Data and Log
File Growth</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>
        </value>
    </kind>
    <resource>
        <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_332</identifier>
        <description>
            <langstring xml:lang="en-US">Course 72475 Topic: Expanding a Transaction
Log</langstring>
        </description>
    </resource>
</relation>
<relation>
    <kind>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">HasPart</langstring>

```

```

        </value>
      </kind>
    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_333</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Topic: Shrinking a
Database</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">HasPart</langstring>
      </value>
    </kind>
    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_334</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Topic: Shrinking a
File</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">HasPart</langstring>
      </value>
    </kind>
    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_335</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Topic: Dropping a
Database</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">HasPart</langstring>

```



```

</kind>
<resource>
  <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_343</identifier>
  <description>
    <langstring xml:lang="en-US">Course 72475 Topic: Dropping User-Defined
Data Types</langstring>
  </description>
</resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>
  <resource>
    <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_351</identifier>
    <description>
      <langstring xml:lang="en-US">Course 72475 Topic: Creating a
Table</langstring>
    </description>
  </resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>
  <resource>
    <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_352</identifier>
    <description>
      <langstring xml:lang="en-US">Course 72475 Topic: Dropping a
Table</langstring>
    </description>
  </resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>

```

```

        </kind>
        <resource>
            <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_353</identifier>
            <description>
                <langstring xml:lang="en-US">Course 72475 Topic: Adding a
Column</langstring>
            </description>
        </resource>
    </relation>
    <relation>
        <kind>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">HasPart</langstring>
            </value>
        </kind>
        <resource>
            <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_354</identifier>
            <description>
                <langstring xml:lang="en-US">Course 72475 Topic: Dropping a
Column</langstring>
            </description>
        </resource>
    </relation>
    <relation>
        <kind>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">HasPart</langstring>
            </value>
        </kind>
        <resource>
            <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_355</identifier>
            <description>
                <langstring xml:lang="en-US">Course 72475 Topic: Generating Column
Values</langstring>
            </description>
        </resource>
    </relation>
    <relation>
        <kind>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">HasPart</langstring>
            </value>
        </kind>

```

```

</kind>
<resource>
  <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_361</identifier>
  <description>
    <langstring xml:lang="en-US">Course 72475 Topic: Schema
Purposes</langstring>
  </description>
</resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>
  <resource>
    <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_362</identifier>
    <description>
      <langstring xml:lang="en-US">Course 72475 Topic: Schema
Considerations</langstring>
    </description>
  </resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>
  <resource>
    <identifier>x-ims-plirid-v0.DUNS.05-107-9929.nloid.en_US_72475_363</identifier>
    <description>
      <langstring xml:lang="en-US">Course 72475 Unit 3 Summary</langstring>
    </description>
  </resource>
</relation>
<relation>
  <kind>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">HasPart</langstring>
    </value>
  </kind>

```



```

    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-
9929.nloid.en_US_72475_Assmt3_Post</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Unit 3 Mastery</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">HasPart</langstring>
      </value>
    </kind>
    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-
9929.nloid.en_US_72475_Assmt3_Pre</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Unit 3
Preassessment</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">HasPart</langstring>
      </value>
    </kind>
    <resource>
      <identifier>x-ims-plirid-v0.DUNS.05-107-9929.hlpid.en_US_72475_Help</identifier>
      <description>
        <langstring xml:lang="en-US">Course 72475 Help</langstring>
      </description>
    </resource>
  </relation>
  <relation>
    <kind>
      <source>
        <langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
        <langstring xml:lang="x-none">Requires</langstring>
      </value>
    </kind>

```



```

        <entry>
          <langstring xml:lang="en-US">Worker
Requirements</langstring>
        </entry>
        <taxon>
          <id>II.C</id>
        <entry>
          <langstring xml:lang="en-
US">Knowledge</langstring>
        </entry>
        <taxon>
          <id>II.C.3</id>
        <entry>
          <langstring xml:lang="en-US">Engineering
and Technology</langstring>
        </entry>
        <taxon>
          <id>II.C.3.a</id>
        <entry>
          <langstring xml:lang="en-
US">Computers and Electronics</langstring>
        </entry>
        </taxon>
      </taxon>
    </taxon>
  </taxonpath>
  <description>
    <langstring xml:lang="en-US">Knowledge of electric circuit boards, processors, chips,
and computer hardware and software, including applications and programming</langstring>
  </description>
</classification>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Skill Level</langstring>
    </value>
  </purpose>
  <taxonpath>
    <source>
      <langstring xml:lang="en-US">O*NET 98</langstring>
    </source>
  </taxon>
  <id>II</id>
  <entry>
    <langstring xml:lang="en-US">Worker
Requirements</langstring>
  </entry>

```

```

        <taxon>
            <id>II.B</id>
            <entry>
                <langstring xml:lang="en-US">Cross-Functional
Skills</langstring>
            </entry>
        </taxon>
        <taxon>
            <id>II.B.3</id>
            <entry>
                <langstring xml:lang="en-US">Technical
Skills</langstring>
            </entry>
        </taxon>
        <taxon>
            <id>II.B.3.a</id>
            <entry>
                <langstring xml:lang="en-
US">Operations Analysis</langstring>
            </entry>
        </taxon>
    </taxon>
</taxon>
</taxon>
</taxonpath>
<description>
    <langstring xml:lang="en-US">Analyzing needs and product requirements to create a
design</langstring>
</description>
</classification>
<classification>
    <purpose>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Skill Level</langstring>
        </value>
    </purpose>
</taxonpath>
<source>
    <langstring xml:lang="en-US">O*NET 98</langstring>
</source>
<taxon>
    <id>II</id>
    <entry>
        <langstring xml:lang="en-US">Worker
Requirements</langstring>
    </entry>
</taxon>
    <id>II.B</id>
    <entry>

```

```

Skills</langstring>
    <langstring xml:lang="en-US">Cross-Functional
</entry>
<taxon>
    <id>II.B.3</id>
<entry>
    <langstring xml:lang="en-US">Technical
Skills</langstring>
</entry>
<taxon>
    <id>II.B.3.b</id>
<entry>
    <langstring xml:lang="en-
US">Technology Design</langstring>
</entry>
</taxon>
</taxon>
</taxon>
</taxon>
</taxonpath>
<description>
    <langstring xml:lang="en-US">Generating or adapting equipment and technology to
serve user needs</langstring>
</description>
</classification>
<classification>
    <purpose>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Skill Level</langstring>
        </value>
    </purpose>
</taxonpath>
    <source>
        <langstring xml:lang="en-US">O*NET 98</langstring>
    </source>
</taxon>
    <id>II</id>
<entry>
    <langstring xml:lang="en-US">Worker
Requirements</langstring>
</entry>
<taxon>
    <id>II.B</id>
<entry>
    <langstring xml:lang="en-US">Cross-Functional
Skills</langstring>
</entry>
</taxon>

```

```

        <id>II.B.3</id>
        <entry>
            <langstring xml:lang="en-US">Technical
Skills</langstring>
        </entry>
    </taxon>
    <taxon>
        <id>II.B.3.c</id>
        <entry>
            <langstring xml:lang="en-
US">Equipment Selection</langstring>
        </entry>
    </taxon>
</taxon>
</taxon>
</taxon>
</taxonpath>
<description>
    <langstring xml:lang="en-US">Determining the kind of tools and equipment needed to
do a job</langstring>
</description>
</classification>
<classification>
    <purpose>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Skill Level</langstring>
        </value>
    </purpose>
    <taxonpath>
        <source>
            <langstring xml:lang="en-US">O*NET 98</langstring>
        </source>
    </taxon>
    <id>II</id>
    <entry>
        <langstring xml:lang="en-US">Worker
Requirements</langstring>
    </entry>
</taxon>
    <id>II.B</id>
    <entry>
        <langstring xml:lang="en-US">Cross-Functional
Skills</langstring>
    </entry>
</taxon>
    <id>II.B.3</id>
    <entry>
        <langstring xml:lang="en-US">Technical
Skills</langstring>
    </entry>

```

```

        </entry>
        <taxon>
            <id>II.B.3.e</id>
            <entry>
                <langstring xml:lang="en-
US">Programming</langstring>
            </entry>
        </taxon>
    </taxon>
</taxon>
</taxon>
</taxon>
</taxonpath>
<description>
    <langstring xml:lang="en-US">Writing computer programs for various
purposes</langstring>
</description>
</classification>
<classification>
    <purpose>
        <source>
            <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
            <langstring xml:lang="x-none">Skill Level</langstring>
        </value>
    </purpose>
    <taxonpath>
        <source>
            <langstring xml:lang="en-US">O*NET 98</langstring>
        </source>
    </taxon>
    <id>II</id>
    <entry>
        <langstring xml:lang="en-US">Worker
Requirements</langstring>
    </entry>
</taxon>
    <id>II.B</id>
    <entry>
        <langstring xml:lang="en-US">Cross-Functional
Skills</langstring>
    </entry>
</taxon>
    <id>II.B.3</id>
    <entry>
        <langstring xml:lang="en-US">Technical
Skills</langstring>
    </entry>
</taxon>
    <id>II.B.3.f</id>
    <entry>

```

```

US">Testing</langstring>
</entry>
</taxon>
</taxon>
</taxon>
</taxon>
</taxonpath>
<description>
  <langstring xml:lang="en-US">Conducting tests to determine whether equipment,
software, or procedures are operating as expected</langstring>
</description>
</classification>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Prerequisite</langstring>
    </value>
  </purpose>
  <description>
    <langstring xml:lang="en-US">NETg Course 72470: Microsoft SQL Server 7.0: System
Administration - Part 1</langstring>
  </description>
</classification>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Prerequisite</langstring>
    </value>
  </purpose>
  <description>
    <langstring xml:lang="en-US">NETg Course 72471: Microsoft SQL Server 7.0: System
Administration - Part 2</langstring>
  </description>
</classification>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Prerequisite</langstring>
    </value>
  </purpose>
  <description>

```



```

        <langstring xml:lang="en-US">NETg Course 72472: Microsoft SQL Server 7.0: System
Administration - Part 3</langstring>
        </description>
    </classification>
    <classification>
        <purpose>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">Prerequisite</langstring>
            </value>
        </purpose>
    </description>
    <langstring xml:lang="en-US">NETg Course 72473: Microsoft SQL Server 7.0: System
Administration - Part 4</langstring>
    </description>
    </classification>
    <classification>
        <purpose>
            <source>
                <langstring xml:lang="x-none">LOMv1.0</langstring>
            </source>
            <value>
                <langstring xml:lang="x-none">Prerequisite</langstring>
            </value>
        </purpose>
    </description>
    <langstring xml:lang="en-US">NETg Course 72474: Microsoft SQL Server 7.0: System
Administration - Part 5</langstring>
    </description>
    </classification>
</lom>

```