

AKAI, S., CHIBA, S., NISHIZAWA, M. **Region Pointcut for AspectJ**. In: Workshop on Aspects, Components, and Patterns for Infrastructure Software, Charlottesville. (ACP4IS'09). ACM, New York, p. 43-48, 2009. ISBN: 978-1-60558-450-8 doi>10.1145/1509276.1509287.

ANDERSEN, L.O. **Program Analysis and Specialization for the C Programming Language**. PHD Thesis. DIKU, University of Copenhagen. Copenhagen, 1994.

AspectJ Team - **The AspectJ Programming Guide** - <http://eclipse.org/aspectj/>, Acessado em: 04/2009.

BERTELS, P., STROOBANDT, D. **Java and the Power of Multi-Core Processing**. In: Proceedings of the 2008 International Conference on Complex, Intelligent and Software Intensive Systems Washington: IEEE Computer Society (CISIS'08), Washington, 2008. ISBN: 978-0-7695-3109-0 doi>10.1109/CISIS.2008.121.

BRAVENBOER, M., SMARAGDAKIS, Y. **Exception Analysis and Points-to Analysis: Better Together**. In: Proceedings of the Eighteenth international Symposium on Software Testing and Analysis (ISSTA'09), ACM, New York, 2009. ISBN: 978-1-60558-338-9 doi>10.1145/1572272.1572274.

BUSCHMANN, F.; MEUNIER, R.; ROHNERT, H.; SOMMERLAD, P.; STAL, M. **Pattern-Oriented Software Architecture**, Volume 1: A System of Patterns. Wiley, 1996, 476p.

CACHO, N., FILHO, F. C., GARCIA, A., FIGUEIREDO, E. **EJFlow: Taming Exceptional Control Flows in Aspect-Oriented Programming**. In: Proceedings of the 7th international Conference on Aspect-Oriented Software Development (AOSD'08), New York, ACM, 2008. ISBN: 978-1-60558-044-9 doi>10.1145/1353482.1353492.

CHATTERJEE, R., RYDER, B. G., LANDI, W. A. **Complexity of Points-To Analysis of Java in the Presence of Exceptions**. IEEE Transactions Software Engineering, Piscataway, v. 27, n. 6, p.481-512, jun. 2001. doi>10.1109/32.926173.

COELHO, R., RASHID, A., GARCIA, A., FERRARI, F., CACHO, N., KULESZA, U., STAA, A., LUCENA, C. **Assessing the Impact of Aspects on Exception Flows: An Exploratory Study**. In: Proceedings of the 22nd European Conference on Object-Oriented Programming (ECOOP'08) Berlin: Springer-Verlag, 2008. ISBN: 978-3-540-70591-8 doi>10.1007/978-3-540-70592-5_10.

COELHO, R. **Analyzing the Exception Flows of Aspect- Oriented Programs**. Tese de Doutorado. Departamento de Informática da PUC. Rio de Janeiro, 2008.

DEMICHIEL, L. **Enterprise JavaBeans™ 2.1** - <http://jcp.org/en/jsr/detail?id=153>. Acessado em: 08/2009.

DOOREN, M.V., STEEGMANS, E. **Combining the Robustness of Checked Exceptions with the Flexibility of Unchecked Exceptions Using Anchored Exception Declarations**. In: Proceedings of the 20th annual ACM SIGPLAN conference on Object oriented programming systems languages and applications (OOPSLA'05), ACM Press, 2005, p.455–471. ISBN:1-59593-031-0 doi>10.1145/1094811.1094847.

EMANUELSSON, P., NILSSON, U. **A Comparative Study of Industrial Static Analysis Tools**. *Electronic Notes in Theoretical Computer Science*, v. 217, p. 5 - 21, 2008. doi>10.1016/j.entcs.2008.06.039.

FILHO, F. C., GARCIA, A., RUBIRA, C. M. **Error Handling as an Aspect**. In: Proceedings of the 2nd Workshop on Best Practices in Applying Aspect-Oriented Software Development (BPAOSD'07), Vancouver, ACM, 2007. ISBN: 978-1-59593-662-2 doi>10.1145/1229485.1229486.

FILHO, F. C., CACHO, N., FIGUEIREDO, E., MARANHÃO, R., GARCIA, A., AND RUBIRA, C. M. **Exceptions and Aspects: the Devil is in the Details**. In: Proceedings of the 14th ACM SIGSOFT international Symposium on Foundations of Software Engineering, 14, 2006, Portland. New York: ACM, 2006. ISBN:1-59593-468-5 doi>10.1145/1181775.1181794.

FLEURY, M., REVERBEL, F. **The JBoss Extensible Server**. In: Proceedings of the ACM/IFIP/USENIX international Conference on Middleware (Middleware '03), Rio de Janeiro, Berlin: Springer-Verlag, 2003.

FU, C.; MILANOVA, A.; RYDER, B.; WONNACOTT, D. **Robustness testing of Java Server Applications**. *IEEE Transactions on Software Engineering*, 31(4), p.292- 311, 2005.

FU, C.; RYDER, B.G. **Testing and Understanding Error Recovery Code in Java Applications**. In: Proceedings of the ECOOP2005 - Workshop on Exception Handling in Object Oriented Systems (EHOOS'2005), 2005.

FU, C., RYDER, B. G. **Exception-Chain Analysis: Revealing Exception Handling Architecture in Java Server Applications**. In: Proceedings of the 29th international Conference on Software Engineering Washington: IEEE Computer Society, 2007. ISBN:0-7695-2828-7 doi>10.1109/ICSE.2007.35.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns: Elements of Reusable Object-Oriented Software**. Addison-Wesley Longman Publishing Co., 1995, 416p.

GARCIA, A.F., RUBIRA, C.M.F. **A Comparative Study of Exception Handling Mechanisms for Building Dependable Object-Oriented Software**. Journal of Systems and Software, v 59, Issue 2, 15 November 2001, Pages 197-222. doi:10.1016/S0164-1212(01)00062-0.

GARCIA, A., SANT'ANNA, C., FIGUEIREDO, E., KULESZA, U., LUCENA, C., STAA, A.V. **Modularizing Design Patterns with Aspects: A Quantitative Study**. In: Proceedings of the Proceedings of the 4th international conference on Aspect-oriented software development (AOSD'05), Chicago, Illinois, ACM Press, 2005, p.3-14. ISBN:1-59593-042-6 doi>10.1145/1052898.1052899.

GOODENOUGH, J. B. **Structured Exception Handling**. In: Proceedings of the 2nd ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL'75), Palo Alto. New York: ACM, 1975. doi>10.1145/512976.512997.

GOODENOUGH, J. B. **Exception Handling: Issues and a Proposed Notation**. Communications of ACM, v. 18, n. 12, p. 683-696, 1975. doi>10.1145/361227.361230.

GOSLING, J., JOY, B., STEELE, G.L. **The Java Language Specification**, 3rd Ed, Boston: Addison-Wesley Publishing Co, 1996, 688p.

GREENWOOD, P., BARTOLOMEI, T., FIGUEIREDO, E., DOSEA, M., GARCIA, A., CACHO, N., SANT'ANNA, C., SOARES, S., BORBA, P., KULESZA, U., RASHID, A. **On the Impact of Aspectual Decompositions on Design Stability: An Empirical Study**. In: Proceedings of the European Conference of Object-Oriented Programming (ECOOP'07), Springer-Verlag, 2007, p.176-200.

GROVE, D., CHAMBERS, C. **A Framework for Call Graph Construction Algorithms**. ACM Transactions on Programming Languages and Systems (TOPLAS), 23(6), p.685-746, 2001.

KICZALES, G. **Aspect-Oriented Programming**. ACM Computing Surveys, v. 28, n. 4, p.220-242, 1997.

HANNEMANN, J., KICZALES, G. **Design Pattern Implementation in Java and AspectJ**. In: Proceedings of the Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA'02), Seattle, Washington, USA, ACM Press, Volume 37, Issue 11, 2002, p.161-173. doi>10.1145/583854.582436.

HOFFMAN, K., EUGSTER, P. **Towards Reusable Components with Aspects: an Empirical Study on Modularity and Obliviousness.** In: Proceedings of the 30th international Conference on Software Engineering (ICSE'08), Leipzig. New York : ACM, 2008. ISBN: 978-1-60558-079-1 doi>10.1145/1368088.1368102.

HSQldb - **HyperSQL Database** - <http://hsqldb.org/>. Acessado em 06/2010

IBM T.J. WATSON RESEARCH CENTER - **Libraries for Analysis (WALA)** - <http://wala.sourceforge.net>. Acessado em: 04/2009

JAD - **Java Decompiler** - <http://www.varaneckas.com/jad> . Acessado em 06/2010

Jboss AS - **Jboss Application Server** - <http://jboss.org/jbossas/> Acessado em 04/2009.

Jboss AOP - **Jboss AOP Documentation** - Disponível em: <http://jboss.org/jbossaop/docs/index.html>. Acessado em 04/2009

JO, J., CHANG, B., YI, K., CHOE, K. **An Uncaught Exception Analysis for Java.** Journal of Systems and Software, v. 72, n. 1, p. 59-69, 2004. doi>10.1016/S0164-1212(03)00057-8.

LHOTÁK, O. HENDREN, L. **Scaling Java Points-to Analysis Using Spark.** In: COMPILER CONSTRUCTION, 12, 2003, local. Compiler Construction, 12th International Conference Berlin: Springer-Verlag, 2003.

LHOTÁK, O. **Program Analysis Using Binary Decision Diagrams.** PhD Thesis, Graduate Department of Computer Science, Universidade McGill, 2006.

LIPPERT, M.; LOPES, C.V. **A Study on Exception Detection and Handling Using Aspect-Oriented Programming.** In: Proceedings of the International Conference on Software Engineering (ICSE'00), ACM Press, 2000, p.418-427.

MATSUOKA, S., YONEZAWA, A. **Analysis of Inheritance Anomaly in Object-Oriented Concurrent Programming Languages.** Cambridge, MA: The MIT Press, 1993.

MILANOVA, A.; ROUNTEV, A.; RYDER, B. **Practical Object-sensitive Points-to Analysis for Java.** Disponível em: www.iei.pi.cnr.it/ISSTA/slides/07222002/sess2pr1.ppt Acessado em: 09/2009.

MILLER, R., TRIPHATHI, A. **Issues with Exception Handling in Object-Oriented Systems.** In: Proceedings of the European Conference on Object Oriented Programming (ECOOP'97). Berlin: Springer-Verlag, 1997. v 1241/1997, p. 85-103, DOI: 10.1007/BFb0053375.

MEZINI, M., OSTERMANN, K. **Conquering Aspects with Caesar.** In: Proceedings of the Proceedings of the 2nd International Conference on Aspect-oriented Software Development (AOSD'03), 2, 2003, Boston. New York: ACM, 2003. ISBN:1-58113-660-9 doi>10.1145/643603.643613.

MYERS, G.J. **The Art of Software Testing**. New York: John Wiley & Sons, 2004, 192p.

Soot - **Java Optimization Framework** - <http://www.sable.mcgill.ca/> Acessado em: 04/2009.

Spring AOP. Disponível em:

<http://static.springsource.org/spring/docs/2.0.x/reference/aop.html>. Acessado em: 05/2009

RASHID, A., CHITCHYAN, R. **Persistence as an Aspect**. In: Proceedings of the 2nd International Conference on Aspect-oriented Software Development, Boston, Massachusetts, ACM Press, 2003, p.120-129.

REPS, T., HORWITZ, S., SAGIV, M. **Precise Interprocedural Dataflow Analysis Via Graph Reachability**. In: Proceedings of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'95), 22, 1995, San Francisco. New York: ACM, 1995. ISBN: 0-89791-692-1 doi>10.1145/199448.199462.

ROBILLARD, M. P., MURPHY, G. C. **Analyzing Exception Flow in Java Programs**. In Proceedings of 7th European Software Engineering Conference and 7th ACM SIGSOFT Symposium on Foundations of Software Engineering, Springer-Verlag, 1999, p.322-337.

ROBILLARD, M. P., MURPHY, G. C. **Static Analysis to Support the Evolution of Exception Structure in Object-Oriented Systems**. ACM Transactions on Software Engineering and Methodology (TOSEM), v. 12, n. 2, p.191-221, 2003. doi>10.1145/941566.941569.

SACRAMENTO, P., CABRAL, B., MARQUES, P. **Unchecked Exceptions: Can the Programmer be Trusted to Document Exceptions?**. In: Proceedings of the International Conference on Innovative Views of .NET Technologies (IVNET'06), Florianópolis, Brazil, 2006.

SHANNON, B. - **Java Plataforma Enterprise Edition** - <http://jcp.org/en/jsr/detail?id=151> . Acessado em: 08/2009.

SINHA, S., SHAH, H.; GÖRG, C., JIANG, S., KIM, M., HARROLD, M. **Fault Localization and Repair for Java Runtime Exceptions**. In: Proceedings of the Eighteenth international Symposium on Software Testing and Analysis (ISSTA'09), Chicago. New York: 2009. ISBN: 978-1-60558-338-9 doi>10.1145/1572272.1572291.

SOARES, S., BORBA, P., LAUREANO, E. **Distribution and Persistence as Aspects**. Software Practice and Experience, v 36, Issue 7, June 2006, p.711-759, 2006. doi>10.1002.

TARDOS, E., KLEINBERG, J. **Algorithm Design**. 1st Ed, Boston: Addison-Wesley Publishing Co, 2005.

THUMMALAPENTA, S., XIE, T. **Mining Exception-Handling Rules as Sequence Association Rules**. In: Proceedings of the 2009 IEEE 31st international Conference on Software Engineering Washington, Washington. IEEE Computer Society, 2009.

TIRELO, F. et al. **Desenvolvimento de Software Orientado por Aspectos**. In: Congresso da Sociedade Brasileira de Computação, 24, 2004, Salvador. Anais. Salvador: Sociedade Brasileira de Computação, 2004, p. 57-96.

THOMAS, D. **The Deplorable State of Class Libraries**. In: Journal of Object Technology, v 1 Issue 1, 2002, p.21-27.

8

Anexos

8.1.Exemplo da Saída da Ferramenta

```
1. C-<edu.drexel.cms.actions.NewUserRegistrationAction:
    java.lang.String execute()>
2. C-<edu.drexel.cms.data.CmsDAO: boolean
    isUserExist(java.lang.String)>
3. C-<javax.naming.InitialContext: java.lang.Object
    lookup(java.lang.String)>
4. C-<javax.naming.InitialContext: javax.naming.Context
    getURLOrDefaultInitCtx(java.lang.String)>
5. C-<javax.naming.spi.NamingManager: javax.naming.Context
    getURLContext(java.lang.String,java.util.Hashtable)>
6. C-<javax.naming.spi.NamingManager: java.lang.Object
    getURLObject(java.lang.String,java.lang.Object,javax.naming
        .Name,javax.naming.Context,java.util.Hashtable)>
7. C-<org.jboss.ejb3.JndiProxyFactory: java.lang.Object
    getObjectInstance(java.lang.Object,javax.naming.Name,javax.
        naming.Context,java.util.Hashtable)>
8. C-<org.jboss.ejb3.stateful.StatefulLocalProxyFactory:
    java.lang.Object createProxy()>
9. C-<org.jboss.ejb3.cache.NoPassivationCache:
    org.jboss.ejb3.stateful.StatefulBeanContext create()>
10.C-<org.jboss.ejb3.StrictMaxPool: org.jboss.ejb3.BeanContext
    get()>
11.C-<org.jboss.ejb3.AbstractPool: org.jboss.ejb3.BeanContext
    create()>
12.C-<org.jboss.ejb3.EJBContainer: java.lang.Object
    construct()>
13.C-<org.jboss.aop.joinpoint.ConstructorInvocation:
    java.lang.Object invokeNext()>
14.C-<org.jboss.aspects.tx.TxInterceptor$Required:
    java.lang.Object
    invoke(org.jboss.aop.joinpoint.Invocation)>
15.C-<org.jboss.aspects.tx.TxPolicy: java.lang.Object
```

```

        invokeInOurTx(org.jboss.aop.joinpoint.Invocation, javax.transaction.TransactionManager)>
16.C-<org.jboss.aspects.tx.TxPolicy: void
    endTransaction(javax.transaction.TransactionManager, javax.transaction.Transaction)>
17.C-<org.jboss.tm.TxManager: void commit()>
18.C-<org.jboss.tm.TransactionImpl: void commit()>
19.C-<org.jboss.tm.TransactionImpl: void beforePrepare()>
20.C-<org.jboss.tm.TransactionImpl: void doBeforeCompletion()>
21.C-<org.jboss.ejb.txtimer.TimerImpl: void
    beforeCompletion()>
22.C-<org.jboss.ejb.txtimer.TimerServiceImpl: void
    removeTimer(org.jboss.ejb.txtimer.TimerImpl)>
23.C-<org.jboss.ejb.txtimer.DatabasePersistencePolicy: void
    deleteTimer(java.lang.String, org.jboss.ejb.txtimer.TimedObjectId)>
24.C-
    <org.jboss.ejb.txtimer.GeneralPurposeDatabasePersistencePlugin: void
    deleteTimer(java.lang.String, org.jboss.ejb.txtimer.TimedObjectId)>
25.C-<org.hsqldb.jdbc.jdbcDataSource: java.sql.Connection
    getConnection()>
26.C-<org.hsqldb.jdbc.jdbcDataSource: java.sql.Connection
    getConnection(java.lang.String, java.lang.String)>
27.C-<org.hsqldb.jdbc.Driver: java.sql.Connection
    getConnection(java.lang.String, java.util.Properties)>
28.C-<org.hsqldb.jdbc.jdbcConnection: void
    <init>(org.hsqldb.persist.HsqlProperties)>
29.C-<org.hsqldb.DatabaseManager: org.hsqldb.Session
    newSession(java.lang.String, java.lang.String, java.lang.String, java.lang.String, org.hsqldb.persist.HsqlProperties)>
30.C-<org.hsqldb.DatabaseManager: org.hsqldb.Database
    getDatabase(java.lang.String, java.lang.String, org.hsqldb.persist.HsqlProperties)>
31.C-<org.hsqldb.Database: void open()>
32.C-<org.hsqldb.Database: void reopen()>
33.C-<org.hsqldb.persist.Logger: void
    openLog(org.hsqldb.Database)>
34.C-<org.hsqldb.persist.Log: void open()>
35.C-<org.hsqldb.persist.Log: void processScript()>

```

```

36.C-<org.hsqldb.scriptio.ScriptReaderBase: void
    readAll(org.hsqldb.Session)>
37.C-<org.hsqldb.scriptio.ScriptReaderBinary: void
    readDDL(org.hsqldb.Session)>
38.C-<org.hsqldb.Session: org.hsqldb.Result
    sqlExecuteDirectNoPreChecks(java.lang.String)>
39.C-<org.hsqldb.DatabaseCommandInterpreter: org.hsqldb.Result
    execute(java.lang.String)>
40.C-<org.hsqldb.DatabaseCommandInterpreter: org.hsqldb.Result
    executePart(int,java.lang.String)>
41.C-<org.hsqldb.Parser: org.hsqldb.CompiledStatement
    compileSelectStatement(int)>
42.C-<org.hsqldb.Parser: org.hsqldb.Select
    parseSelect(int,boolean,boolean,boolean,boolean)>
43.C-<org.hsqldb.Parser: void
    parseLimit(java.lang.String,org.hsqldb.Select,boolean)>
44.C-<org.hsqldb.Parser: org.hsqldb.Expression readTerm()>
45.C-<org.hsqldb.Parser: org.hsqldb.Expression
    readColumnExpression()>
46.C-<org.hsqldb.Parser: org.hsqldb.Expression readOr()>
47.C-<org.hsqldb.Parser: org.hsqldb.Expression readAnd()>
48.C-<org.hsqldb.Parser: org.hsqldb.Expression
    readCondition()>
49.C-<org.hsqldb.Parser: org.hsqldb.SubQuery
    parseSubquery(int,org.hsqldb.HsqlNameManager$HsqlName[],boolean,int)>
50.C-<org.hsqldb.Select: boolean
    resolveAll(org.hsqldb.Session,boolean)>
51.C-<org.hsqldb.Select: void resolve(org.hsqldb.Session)>
52.C-<org.hsqldb.Select: void
    resolveTypes(org.hsqldb.Session)>
53.C-<org.hsqldb.Expression: void
    resolveTypes(org.hsqldb.Session)>
54.C-<org.hsqldb.Expression: java.lang.Object
    getValue(org.hsqldb.Session,int)>
55.C-<org.hsqldb.Expression: java.lang.Object
    getValue(org.hsqldb.Session)>
56.C-<org.hsqldb.Select: java.lang.Object
    getValue(org.hsqldb.Session,int)>
57.C-<org.hsqldb.Select: org.hsqldb.Result
    getResult(org.hsqldb.Session,int)>

```

```

58.C-<org.hsqldb.Select: org.hsqldb.Result
    getSingleResult(org.hsqldb.Session,int)>
59.C-<org.hsqldb.Select: org.hsqldb.Result
    buildResult(org.hsqldb.Session,int)>
60.C-<org.hsqldb.Expression: boolean
    testCondition(org.hsqldb.Session)>
61.C-<org.hsqldb.Expression: java.lang.Boolean
    test(org.hsqldb.Session)>
62.C-<org.hsqldb.Expression: java.lang.Boolean
    testInCondition(org.hsqldb.Session,java.lang.Object)>
63.C-<org.hsqldb.SubQuery: void
    populateTable(org.hsqldb.Session)>
64.C-<org.hsqldb.Table: int
    insertResult(org.hsqldb.Session,org.hsqldb.Result)>
65.C-<org.hsqldb.Table: void
    insertData(org.hsqldb.Session,java.lang.Object[])>
66.C-<org.hsqldb.Table: void
    indexRow(org.hsqldb.Session,org.hsqldb.Row)>
67.C-<org.hsqldb.Index: void
    insert(org.hsqldb.Session,org.hsqldb.Row,int)>
68.C-<org.hsqldb.Index: void
    set(org.hsqldb.Node,boolean,org.hsqldb.Node)>
69.C-<org.hsqldb.DiskNode: org.hsqldb.Node getUpdatedNode()>
70.C-<org.hsqldb.CachedDataRow: org.hsqldb.Row
    getUpdatedRow()>
71.C-<org.hsqldb.Table: org.hsqldb.CachedRow
    getRow(int,org.hsqldb.Node)>
72.C-<org.hsqldb.Table$RowStore:
    org.hsqldb.persist.CachedObject get(int)>
73.C-<org.hsqldb.persist.DataFileCache:
    org.hsqldb.persist.CachedObject
    get(int,org.hsqldb.persist.PersistentStore,boolean)>
74.C-<org.hsqldb.Table$RowStore:
    org.hsqldb.persist.CachedObject
    get(org.hsqldb.rowio.RowInputInterface)>
75.C-<org.hsqldb.CachedDataRow:                                void
    <init>(org.hsqldb.Table,org.hsqldb.rowio.RowInputInterface)
    >
76.C-<org.hsqldb.rowio.RowInputBase:                            java.lang.Object[]
    readData(int[])>
77.C-<org.hsqldb.rowio.RowInputTextLog:                        java.lang.String

```

```

    readChar(int)>
78.C-<org.hsqldb.rowio.RowInputTextLog:      java.lang.String
    readString()>
79.C-<org.hsqldb.rowio.RowInputTextLog:      java.lang.String
    readField()>
80.C-<org.hsqldb.Tokenizer: int getType()>

```

8.2.Lista das Exceções da Aplicação CMS

Exceção	Número de Fluxos
edu.drexel.cms.commons.CmsDataValidationException	224
java.beans.IntrospectionException	12
java.io.EOFException	429
java.io.FileNotFoundException	2
java.io.IOException	32284
java.io.InvalidObjectException	53
java.io.UTFDataFormatException	128
java.lang.AssertionError	1971
java.lang.ClassCastException	56
java.lang.ClassNotFoundException	364
java.lang.Error	45289
java.lang.Exception	1504
java.lang.IllegalArgumentException	4,57052E+11
java.lang.IllegalStateException	1,70032E+11
java.lang.IndexOutOfBoundsException	1,11266E+14
java.lang.InternalError	49251
java.lang.InterruptedException	456
java.lang.NegativeArraySizeException	24191
java.lang.NullPointerException	4129579
java.lang.RuntimeException	8,92238E+15
java.lang.SecurityException	18002
java.lang.UnsupportedOperationException	8,88778E+12
java.lang.reflect.InvocationTargetException	2
java.net.ConnectException	375
java.net.MalformedURLException	202
java.net.SocketException	1845
java.net.SocketTimeoutException	210
java.net.UnknownHostException	10

java.rmi.ConnectException	2
java.rmi.MarshalException	104
java.rmi.NoSuchObjectException	262
java.rmi.RemoteException	89561
java.rmi.ServerException	100494
java.security.InvalidAlgorithmParameterException	21
java.security.InvalidKeyException	54
java.security.InvalidParameterException	8
java.security.NoSuchAlgorithmException	368
java.security.NoSuchProviderException	72
java.security.cert.CertificateException	41
java.security.cert.CertificateParsingException	2
java.sql.SQLException	521571
java.util.NoSuchElementException	5,78671E+12
java.util.jar.JarException	20
javassist.CannotCompileException	1379430
javassist.NotFoundException	15775
javassist.bytecode.AlignmentException	7
javassist.bytecode.BadBytecode	3207
javassist.bytecode.CodeAttribute\$RuntimeCopyException	7,68406E+12
javassist.bytecode.DuplicateMemberException	6774
javassist.bytecode.StackMapTable\$RuntimeCopyException	2,48131E+12
	8122565
javassist.compiler.CompileError	56
javassist.compiler.NoFieldException	5385
javassist.compiler.SyntaxError	3259
javax.crypto.SunJCE_e\$P	407
javax.ejb.ConcurrentAccessException	2
javax.ejb.EJBAccessException	6
javax.ejb.EJBException	512157
javax.ejb.EJBTransactionRequiredException	2
javax.ejb.EJBTransactionRolledbackException	6
javax.ejb.NoSuchEJBException	12688
javax.ejb.NoSuchEntityException	270
javax.ejb.NoSuchObjectLocalException	279
javax.ejb.RemoveException	2
javax.management.AttributeNotFoundException	196
javax.management.InstanceNotFoundException	56
javax.management.IntrospectionException	10

javax.management.InvalidAttributeValueException	3314
javax.management.JMRuntimeException	4
javax.management.ListenerNotFoundException	643
javax.management.MBeanException	3824
javax.management.MalformedObjectNameException	737
javax.management.NotCompliantMBeanException	804
javax.management.ReflectionException	10
javax.management.RuntimeErrorException	16
javax.management.RuntimeMBeanException	10
javax.management.RuntimeOperationsException	42063
javax.management.openmbean.InvalidKeyException	66
javax.management.openmbean.InvalidOpenTypeException	18
javax.management.openmbean.KeyAlreadyExistsException	18
javax.management.openmbean.OpenDataException	227
javax.naming.CannotProceedException	80
javax.naming.CommunicationException	5337
javax.naming.ConfigurationException	198
javax.naming.InvalidNameException	56252
javax.naming.MalformedLinkException	198
javax.naming.NameAlreadyBoundException	47
javax.naming.NameNotFoundException	531
javax.naming.NamingException	333325
javax.naming.NoInitialContextException	2621
javax.naming.NotContextException	546
javax.naming.OperationNotSupportedException	1282
javax.net.ssl.SSLException	272
javax.persistence.TransactionRequiredException	1491
javax.resource.ResourceException	836
javax.security.auth.login.LoginException	6
javax.transaction.HeuristicMixedException	32
javax.transaction.HeuristicRollbackException	32
javax.transaction.NotSupportedException	28
javax.transaction.RollbackException	325
javax.transaction.SystemException	25
javax.transaction.xa.XAException	1546
org.hsqldb.HsqlException	10608536917
org.hsqldb.HsqlInternalException	611944
org.jboss.annotation.factory.AnnotationValidationException	13
org.jboss.aop.NotFoundInDispatcherException	40

org.jboss.aop.advice.InvalidAdviceException	2770
org.jboss.aop.advice.NoMatchingAdviceException	1881
org.jboss.aspects.versioned.LockAttemptFailure	65
org.jboss.aspects.versioned.OptimisticLockFailure	31
org.jboss.deployment.DeploymentException	242
org.jboss.ejb.plugins.cmp.jdbc2.schema.Cache\$RemoveException	4
org.jboss.ejb.plugins.lock.NonReentrantLock\$ReentranceException	13
org.jboss.invocation.JBossLazyUnmarshallingException	964
org.jboss.kernel.spi.registry.KernelRegistryEntryNotFoundException	2
org.jboss.remoting.CannotConnectException	282
org.jboss.remoting.ConnectionFailedException	189
org.jboss.remoting.InvalidConfigurationException	193
org.jboss.remoting.ServerInvoker\$InvalidStateException	36
org.jboss.remoting.callback.HandleCallbackException	1134
org.jboss.remoting.marshall.InvalidMarshallingResource	14
org.jboss.remoting.samples.chat.exceptions.NameInUseException	55
org.jboss.remoting.samples.chat.exceptions.ShuttingDownException	179
org.jboss.remoting.security.SSLSocketBuilder\$NullStoreURLErrorException	48
org.jboss.remoting.transport.bisocket.BisocketServerInvoker\$ClientUnavailableException	8
org.jboss.remoting.transport.http.WebServerError	14
org.jboss.resource.JBossResourceException	544
org.jboss.resource.connectionmanager.JBossLocalXAException	84
org.jboss.tm.JBossRollbackException	6
org.jboss.tm.JBossTransactionRolledbackException	170
org.jboss.tm.JBossTransactionRolledbackLocalException	170
org.jboss.util.NestedError	432
org.jboss.util.NestedRuntimeException	171227
org.jboss.util.NestedSQLException	7041
org.jboss.util.deadlock.ApplicationDeadlockException	231
org.postgresql.util.PSQLException	3648
org.postgresql.xa.PGXException	268
org.springframework.orm.jdo.JdoObjectRetrievalFailureException	110

n	
org.springframework.orm.jdo.JdoOptimisticLockingFailureException	110
org.w3c.dom.DOMException	365118
sun.security.validator.ValidatorException	114

8.3.Lista das Exceções da Aplicação WordNet

Exceções	Número de Fluxos
java.beans.IntrospectionException	18
java.io.EOFException	429
java.io.FileNotFoundException	2
java.io.IOException	90276
java.io.InvalidObjectException	53
java.io.UTFDataFormatException	128
java.lang.AssertionError	3378
java.lang.ClassCastException	56
java.lang.ClassNotFoundException	531
java.lang.Error	11855
java.lang.Exception	1545
java.lang.IllegalArgumentException	4,55674E+11
java.lang.IllegalStateException	1,19838E+12
java.lang.IndexOutOfBoundsException	1,1092E+14
java.lang.InterruptedExecution	450
java.lang.NegativeArraySizeException	24191
java.lang.NullPointerException	2663173
java.lang.RuntimeException	2,59704E+15
java.lang.SecurityException	9214
java.lang.UnsupportedOperationException	5,8296E+12
java.lang.reflect.InvocationTargetException	2
java.net.ConnectException	705
java.net.MalformedURLException	202
java.net.SocketException	2915
java.net.SocketTimeoutException	375
java.net.UnknownHostException	10
java.rmi.ConnectException	2
java.rmi.MarshalException	233
java.rmi.NoSuchObjectException	420

java.rmi.RemoteException	64560
java.rmi.ServerException	71520
java.security.InvalidAlgorithmParameterException	21
java.security.InvalidKeyException	54
java.security.InvalidParameterException	8
java.security.NoSuchAlgorithmException	368
java.security.NoSuchProviderException	72
java.security.cert.CertificateException	41
java.security.cert.CertificateParsingException	2
java.sql.SQLException	137288
java.util.NoSuchElementException	5,76966E+12
java.util.jar.JarException	20
javassist.CannotCompileException	1588099
javassist.NotFoundException	14134
javassist.bytecode.AlignmentException	7
javassist.bytecode.BadBytecode	3207
javassist.bytecode.CodeAttribute\$RuntimeCopyException	5,01811E+12
javassist.bytecode.DuplicateMemberException	5955
javassist.bytecode.StackMapTable\$RuntimeCopyException	1,62043E+12
javassist.compiler.CompileError	1001818727
javassist.compiler.NoFieldException	6383901
javassist.compiler.SyntaxError	3259
javax.crypto.SunJCE_e\$P	407
javax.ejb.ConcurrentAccessException	2
javax.ejb.EJBAccessException	6
javax.ejb.EJBException	894347
javax.ejb.EJBTransactionRequiredException	2
javax.ejb.EJBTransactionRolledbackException	6
javax.ejb.NoSuchEJBException	195
javax.ejb.NoSuchEntityException	260
javax.ejb.NoSuchObjectLocalException	405
javax.ejb.RemoveException	2
javax.imageio.metadata.IIOMException	94899
javax.management.AttributeNotFoundException	136
javax.management.InstanceNotFoundException	56
javax.management.InvalidAttributeValueException	69
javax.management.JMRuntimeException	4
javax.management.ListenerNotFoundException	517
javax.management.MBeanException	833

javax.management.MalformedObjectNameException	737
javax.management.NotCompliantMBeanException	12
javax.management.ReflectionException	10
javax.management.RuntimeErrorException	16
javax.management.RuntimeMBeanException	10
javax.management.RuntimeOperationsException	11926
javax.management.openmbean.InvalidKeyException	31
javax.management.openmbean.InvalidOpenTypeException	8
javax.management.openmbean.KeyAlreadyExistsException	8
javax.management.openmbean.OpenDataException	126
javax.naming.CannotProceedException	88
javax.naming.CommunicationException	7035
javax.naming.ConfigurationException	224
javax.naming.InvalidNameException	41369
javax.naming.MalformedLinkException	88
javax.naming.NameAlreadyBoundException	47
javax.naming.NameNotFoundException	613
javax.naming.NamingException	130556
javax.naming.NoInitialContextException	1915
javax.naming.NotContextException	709
javax.naming.OperationNotSupportedException	1745
javax.net.ssl.SSLException	518
javax.persistence.TransactionRequiredException	7
javax.resource.ResourceException	836
javax.security.auth.login.LoginException	6
javax.servlet.ServletException	8
javax.transaction.HeuristicMixedException	32
javax.transaction.HeuristicRollbackException	32
javax.transaction.NotSupportedException	28
javax.transaction.RollbackException	47873
javax.transaction.SystemException	10
javax.transaction.xa.XAException	1299
javax.xml.parsers.FactoryConfigurationError	420
javax.xml.parsers.FactoryFinder\$ConfigurationError	4
javax.xml.transform.FactoryFinder\$ConfigurationError	2
javax.xml.transform.TransformerFactoryConfigurationError	136
org.dom4j.DocumentException	50
org.dom4j.IllegalAddException	3343399
org.dom4j.datatype.InvalidSchemaException	182016

org.hsqldb.HsqlException	10608536917
org.hsqldb.HsqlInternalException	611944
org.jboss.annotation.factory.AnnotationValidationException	13
org.jboss.aop.NotFoundInDispatcherException	40
org.jboss.aop.advice.InvalidAdviceException	2286
org.jboss.aop.advice.NoMatchingAdviceException	1551
org.jboss.aspects.versioned.LockAttemptFailure	65
org.jboss.aspects.versioned.OptimisticLockFailure	31
org.jboss.deployment.DeploymentException	242
org.jboss.dom4j.DocumentException	54
org.jboss.dom4j.IllegalAddException	3343399
org.jboss.dom4j.datatype.InvalidSchemaException	181205
org.jboss.ejb.plugins.cmp.jdbc2.schema.Cache\$RemoveException	4
org.jboss.ejb.plugins.lock.NonReentrantLock\$ReentranceException	13
org.jboss.invocation.JBossLazyUnmarshallingException	5638
org.jboss.kernel.spi.registry.KernelRegistryEntryNotFoundException	2
org.jboss.remoting.CannotConnectException	566
org.jboss.remoting.ConnectionFailedException	2
org.jboss.remoting.InvalidConfigurationException	193
org.jboss.remoting.ServerInvoker\$InvalidStateException	36
org.jboss.remoting.callback.HandleCallbackException	1261
org.jboss.remoting.marshal.InvalidMarshallingResource	32
org.jboss.remoting.samples.chat.exceptions.NameInUseException	55
org.jboss.remoting.samples.chat.exceptions.ShuttingDownException	179
org.jboss.remoting.security.SSLSocketBuilder\$NullStoreURLException	48
org.jboss.remoting.transport.bisocket.BisocketServerInvoker\$ClientUnavailableException	18
org.jboss.remoting.transport.http.WebServerError	32
org.jboss.resource.JBossResourceException	544
org.jboss.resource.connectionmanager.JBossLocalXAException	81
org.jboss.tm.JBossRollbackException	6
org.jboss.tm.JBossTransactionRolledbackException	170
org.jboss.tm.JBossTransactionRolledbackLocalException	170

org.jboss.util.NestedError	390
org.jboss.util.NestedRuntimeException	157266
org.jboss.util.NestedSQLException	3117
org.jboss.util.deadlock.ApplicationDeadlockException	218
org.postgresql.util.PSQLException	1586
org.postgresql.xa.PGXAException	259
org.w3c.dom.DOMException	3658578
org.xml.sax.SAXException	53
org.xml.sax.SAXNotRecognizedException	20
org.xml.sax.SAXNotSupportedException	41
sun.security.validator.ValidatorException	114

8.4.Lista das Exceções da Aplicação Health Care

Exceção	Número de Fluxos
com.mysql.jdbc.AssertionFailedException	3504
com.mysql.jdbc.CommunicationsException	1091753
com.mysql.jdbc.ConnectionFeatureNotAvailableException	370
com.mysql.jdbc.PacketTooBigException	6670
com.mysql.jdbc.RowDataDynamic\$OperationNotSupportedException	78
java.io.EOFException	226
java.io.IOException	280
java.io.InvalidObjectException	26
java.lang.AssertionError	1233
java.lang.ClassNotFoundException	4
java.lang.Error	27
java.lang.Exception	192
java.lang.IllegalArgumentException	2147854984
java.lang.IllegalStateException	11325496475
java.lang.IndexOutOfBoundsException	214673419
java.lang.InterruptedException	82
java.lang.NullPointerException	936482923
java.lang.NumberFormatException	26
java.lang.RuntimeException	5,15972E+14
java.lang.SecurityException	44

java.lang.UnsupportedOperationException	8,64469E+12
java.net.ConnectException	4
java.net.MalformedURLException	18
java.net.SocketException	20
java.rmi.RemoteException	11
java.rmi.ServerException	58
java.security.NoSuchAlgorithmException	6
java.sql.SQLException	1553590
java.sql.SQLWarning	2
java.util.NoSuchElementException	1817756 80
javassist.CannotCompileException	484677
javassist.NotFoundException	4342
javassist.bytecode.AlignmentException	7
javassist.bytecode.BadBytecode	3207
javassist.bytecode.CodeAttribute\$RuntimeCopyException	7,3583E+12
javassist.bytecode.DuplicateMemberException	1645
javassist.bytecode.StackMapTable\$RuntimeCopyException	2,37612E+12
javassist.compiler.CompileError	5016432973
javassist.compiler.NoFieldException	348859
javassist.compiler.SyntaxError	3259
javax.ejb.ConcurrentAccessException	2
javax.ejb.EJBAccessException	6
javax.ejb.EJBException	4261335
javax.ejb.EJBTransactionRequiredException	2
javax.ejb.EJBTransactionRolledbackException	6
javax.ejb.NoSuchEJBException	7344103
javax.management.AttributeNotFoundException	16
javax.management.InstanceNotFoundException	2
javax.management.JMRuntimeException	2
javax.management.MBeanException	33
javax.management.MalformedObjectNameException	249
javax.management.NotCompliantMBeanException	12
javax.management.ReflectionException	2
javax.management.RuntimeErrorException	4
javax.management.RuntimeMBeanException	2
javax.management.RuntimeOperationsException	394
javax.management.openmbean.InvalidKeyException	15
javax.management.openmbean.InvalidOpenTypeException	4

javax.management.openmbean.KeyAlreadyExistsException	4
javax.management.openmbean.OpenDataException	126
javax.naming.CannotProceedException	44
javax.naming.CommunicationException	579
javax.naming.ConfigurationException	199
javax.naming.InvalidNameException	51510
javax.naming.NameAlreadyBoundException	30
javax.naming.NameNotFoundException	307
javax.naming.NamingException	2292677
javax.naming.NoInitialContextException	5117
javax.naming.NotContextException	359
javax.naming.OperationNotSupportedException	2995
javax.persistence.TransactionRequiredException	7
javax.transaction.HeuristicMixedException	32
javax.transaction.HeuristicRollbackException	32
javax.transaction.NotSupportedException	6
javax.transaction.RollbackException	229
org.jboss.annotation.factory.AnnotationValidationException	13
org.jboss.aop.NotFoundInDispatcherException	4
org.jboss.aop.advice.InvalidAdviceException	473
org.jboss.aop.advice.NoMatchingAdviceException	321
org.jboss.aspects.versioned.LockAttemptFailure	29
org.jboss.aspects.versioned.OptimisticLockFailure	21
org.jboss.tm.JBossRollbackException	6
org.jboss.util.NestedRuntimeException	20513767
org.jboss.util.deadlock.ApplicationDeadlockException	2

8.5. Classe Simples Após o Weaving

```

1. public class Pojo implements Advised{
2.     public static class Pojo_1_MByMInvocation
3. extends MethodCalledByMethodInvocation
4.     implements Untransformable {
5.         public Object[] getArguments() {
6.             }
7.
8.         final void enforceArgsConsistency(){

```

```

9.         }
10.        public Object invokeTarget() throws Throwable {
11.            enforceArgsConsistency();
12.            typedTargetObject.m();
13.            return null;
14.        }
15.        public Invocation copy(){
16.        }
17.        public Pojo typedTargetObject;
18.    }
19.    public void m()    {
20.        System.out.println("method m invoked");
21.    }
22.    public static void main(String args[])    {
23.        Pojo pojo = new Pojo();
24.        MethodByMethodInfo methodbymethodinfo =
        (MethodByMethodInfo)
25.        aop$methodCall__N_1585215547985831794com_meo_
26.        safe_teste_after_Pojo_N_8891172967806401140.
27.        get();
28.        if(methodbymethodinfo.getInterceptors() !=
        (Interceptor[])null)
29.        {
30.            Pojo_1_MByMInvocation pojo_1_mbyminvocation =
            new Pojo_1_MByMInvocation(methodbymethodinfo, null, pojo, new
            Object[0], methodbymethodinfo.getInterceptors());
31.            pojo_1_mbyminvocation.typedTargetObject = pojo;
32.            pojo_1_mbyminvocation.setTargetObject(pojo);
33.            Object obj =
            pojo_1_mbyminvocation.invokeNext();
34.        } else {
35.            pojo.m();
36.            Object obj1 = null;
37.        }
38.    }
39.    public Advisor _getAdvisor() {
40.        return aop$classAdvisor$aop;
41.    }
42.    public InstanceAdvisor _getInstanceAdvisor() {
43.        synchronized(this) {
44.            if(_instanceAdvisor == null)

```

```

45.         _instanceAdvisor = new
           ClassInstanceAdvisor(this);
46.         return _instanceAdvisor;
47.     }
48. }
49.
50.     public void _setInstanceAdvisor(InstanceAdvisor
           instanceadvisor) {
51.         synchronized(this){
52.             _instanceAdvisor =
               (ClassInstanceAdvisor)instanceadvisor;
53.         }
54.     }
55.     private static ClassAdvisor aop$classAdvisor$aop;
56.     protected transient ClassInstanceAdvisor
           _instanceAdvisor;
57.     private static WeakReference
           aop$methodCall__N_1585215547985831794com_meo_safe_teste_after_
58.     Pojo_N_8891172967806401140;
59.     static {
60.         aop$classAdvisor$aop =
           AspectManager.instance().getAdvisor(Class.forName(
61.     "com.meo.safe.teste.after.Pojo"));
62.
           aop$methodCall__N_1585215547985831794com_meo_safe_teste_after_
63.     Pojo_N_8891172967806401140 =
           aop$classAdvisor$aop.resolveCallerMethodInfoAsWeakReference(
64.     0xea002edbale0d88eL, "com.meo.safe.teste.after.Pojo",
           0x849c354b1158058cL);
65.     }
66. }

```

8.6.Código modificado pela instrumentação Generated Advisor Weaving

```

1. class Pojo implements Advised {
2.     public static class PojoAdvisor extends
3.         GeneratedClassAdvisor implements
4.         Untransformable {
5.         public void aop$mc1(Pojo pojo)

```

```

6.      {
7.
      aop$mc1.getInterceptorChainReadWriteLock().readLock().lock();
8.          if(jpmb1 == null && aop$mc1 != null &&
      aop$mc1.hasAdvices() && (jpmb1 == null && aop$mc1 != null &&
      aop$mc1.hasAdvices()))
9.              super.generateJoinPointClass(aop$mc1);
10.             if(jpmb1 == null)
11.                 pojo.m();
12.             else
13.                 jpmb1.invokeJoinpoint(pojo);
14.             break MISSING_BLOCK_LABEL_127;
15.             Exception exception;
16.             exception;
17.
      aop$mc1.getInterceptorChainReadWriteLock().readLock().unlock(
      );
18.             throw exception;
19.
      aop$mc1.getInterceptorChainReadWriteLock().readLock().unlock(
      );
20.             return;
21.         }
22.         protected Domain domain;
23.         protected MethodByMethodInfo aop$mc1;
24.         public JP1 jpmb1;
25.     }
26.     public static class PojoInstanceAdvisor extends
      PojoAdvisor implements Untransformable,InstanceAdvisor {
27.         public void aop$mc1(Pojo pojo) {
28.             checkVersion();
29.             if(!aop$mc1_updated)
30.                 break MISSING_BLOCK_LABEL_99;
31.
      super.mcN15.getInterceptorChainReadWriteLock().
32. writeLock().lock();
33.             super.mcN15.setInterceptors(
34. instanceAdvisorMixin.getWrappers(
35. parent.mcN15.getInterceptors()));
36.             super.jpmb1 = null;
37.             aop$mc1_updated = false;

```

```

38.        super.rebindJoinPointWithInstanceInformation(super.mcN15);
39.            break MISSING_BLOCK_LABEL_86;
40.            Exception exception;
41.            exception;
42.
43.        super.mcN15.getInterceptorChainReadWriteLock().writeLock().un
lock();
44.            throw exception;
45.        super.mcN15(pojo);
46.        return;
47.    }
48.        GeneratedInstanceAdvisorMixin
instanceAdvisorMixin;
49.        PojoAdvisor parent;
50.        protected boolean aop$mcl_updated;
51.        public PojoInstanceAdvisor(
52.            Object obj,
53.            PojoAdvisor pojoadvisor) {
54.            super(pojoadvisor);
55.            parent = null;
56.            instanceAdvisorMixin = new
GeneratedInstanceAdvisorMixin(
57.                obj, pojoadvisor);
58.            parent = pojoadvisor;
59.        }
60.    }
61.    public static class JP1 extends
62.        MethodCalledByMethodInvocation
implements
63.        Untransformable {
64.        protected void invokeJoinpoint(Pojo pojo)
65.            throws Throwable {
66.        }
67.
68.        protected Pojo typedTargetObject;
69.        protected MethodByMethodInfo info;
70.        public JP1(

```

```
71.                                     MethodByMethodInfo
    methodbymethodinfo) {
72.                                     super(methodbymethodinfo, null, null,
73.
        methodbymethodinfo.getInterceptors());
74.                                     info = methodbymethodinfo;
75.                                     }
76.                                     protected JP1(
77.                                     JP1 jp1, Pojo pojo) {
78.                                     this(jp1.info);
79.                                     super.targetObject = pojo;
80.                                     typedTargetObject = pojo;
81.                                     }
82.                                     }
83.                                     public Pojo() {
84.                                     currentAdvisor$aop = _getAdvisor();
85.                                     }
86.                                     public void m() {
87.                                     System.out.println("method m invoked");
88.                                     }
89.                                     public static void main(String args[]) {
90.                                     Pojo pojo = new Pojo();
91.                                     ((PojoAdvisor) aop$classAdvisor$aop)
92.                                     .mcN15(pojo);
93.                                     }
94.                                     private static transient Advisor
        aop$classAdvisor$aop = new PojoAdvisor();
95.                                     protected volatile transient Advisor
        currentAdvisor$aop;
96.                                     protected transient InstanceAdvisor
        instanceAdvisor$aop;
97.                                     }
```