Assignment series 1

Thijs Klaver & Carly Hill

Project

The project has 4 different modules, one for lines of code, one for the method metrics (unit size and complexity), one for the duplication and one for the use of the program. The last contains the methods to measure the metrics. The program should mostly explain themselves, and there is also some comments to help with understanding. For measuring LOC we used the SLOC measure, where everything except for comments and blank lines is measured, so including import statements and lines with only a bracket. For measuring the code duplication, we look at blocks of 6 rules. We check if these six rules are somewhere in a string with all of the lines from the complete program, and if this is true we add one line to the block, to find the biggest block. For every duplication the amount of duplicated lines are counted, and in the end the percentage is counted by dividing by the total lines of code.

Inspiration

UnitComplexity is programmed after the piece of source code in this article:
Landman, Davy, et al. "Empirical analysis of the relationship between CC and SLOC in a large corpus of Java methods and C functions." Journal of Software: Evolution and Process (2015).

Thresholds are programmed after the artile named in the reader:

I. Heitlager, T. Kuipers, and J. Visser. A Practical Model for Measuring Maintainability. In Quality of Information and Communications Technology, 2007. QUATIC 2007. 6th International Conference on the, pages 30–39, Sept 2007.

With backup from the slightly more extensive document online, which has some more tables:
http://docs.sonarqube.org/display/SONARQUBE45/SIG+Maintainability+Model+Plugin

Results

Smallsql project

```
rascal>getMetrics(|project://smallsql0.21_src|);
M3 Model built
--------------------------
Lines Of Code (LOC): 25047

Big methods:
   |java+method:///smallsql/database/DateTime/parse(java.lang.String)|
   |java+method:///smallsql/database/SQLParser/datatype(boolean)|
|java+method:///smallsql/database/SQLParser/function(smallsql.database.Command,smallsq
l.database.SQLToken,boolean)|
    |java+method:///smallsql/junit/TestDataTypes/runTest()|
    |java+method:///smallsql/database/SQLTokenizer/parseSQL(char%5B%5D)|

|java+method:///smallsql/database/StoreImpl/writeExpression(smallsql.database.Expression,
smallsql.database.Column)|
    |java+method:///smallsql/database/DateTime/toString(int)|

|java+method:///smallsql/junit/BasicTestCase/assertEqualsRsValue(java.lang.Object,java.sql
.ResultSet,boolean)|

|java+method:///smallsql/database/SQLParser/expressionSingle(smallsql.database.Comman
d,smallsql.database.SQLToken)|
    |java+method:///smallsql/database/ExpressionArithmetic/getBoolean()|

|java+method:///smallsql/database/ExpressionValue/accumulate(smallsql.database.Expressi
on)|

Complex methods:

|java+method:///smallsql/database/SQLParser/function(smallsql.database.Command,smallsq
l.database.SQLToken,boolean)|
    |java+method:///smallsql/junit/TestDataTypes/runTest()|
    |java+method:///smallsql/database/SQLTokenizer/parseSQL(char%5B%5D)|

|java+method:///smallsql/database/StoreImpl/writeExpression(smallsql.database.Expression,
smallsql.database.Column)|
    |java+method:///smallsql/database/ExpressionArithmetic/getBoolean()|

|java+method:///smallsql/database/ExpressionValue/accumulate(smallsql.database.Expressi
on)|

Percentages of LOC per risk category for Unit Size:
Percentage very high:   7.945063281000
Percentage high:        10.20481495000
Percentage medium:      33.98410987000
Percentage low:         32.08368268000
Percentage LOC outside methods: 15.782329219000

Percentages of LOC per risk category for Unit Complexity:
Percentage very high:   5.010580109000
Percentage high:        9.905377890000
Percentage medium:      6.986864694000
Percentage low:         62.31484809000
Percentage LOC outside methods: 15.782329217000

Percentage duplicated lines: 8.935201821000

Results
--------------------------
Volume:              ++
Unit Size:           --
Unit Complexity:     --
Code Duplication:     o

Analysability:  o
Changeability:  -
Testability:    --

Overall maintainability: -
```

Hsqldb project(without duplication)

```
rascal>getMetricsWithoutDuplication(|project://hsqldb-2.3.1|);
Start building M3 model...
M3 Model built
--------------------------
Lines Of Code (LOC): 184414

Big methods:

|java+method:///org/hsqldb/jdbc/JDBCPreparedStatement/setParameter(int,java.lang.Object)|
   |java+method:///org/hsqldb/jdbc/JDBCClobClient/getAsciiStream()|
   |java+method:///org/hsqldb/ParserDDL/compileCreate()|

|java+method:///org/hsqldb/ExpressionOp/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/util/DatabaseManagerSwing/PopupListener/getCommandString()|

|java+method:///org/hsqldb/QuerySpecification/buildResult(org.hsqldb.Session,int%5B%5D)|
   |java+method:///org/hsqldb/Scanner/scanToken()|
   |java+method:///org/hsqldb/persist/Logger/open()|

|java+method:///org/hsqldb/index/IndexAVL/delete(org.hsqldb.persist.PersistentStore,org.hsqldb.index.NodeAVL)|
   |java+method:///org/hsqldb/cmdline/SqlFile/processSpecial(java.lang.String)|
   |java+method:///org/hsqldb/test/TestSchemaParse/multiPartKeywords(java.lang.String)|
    |java+method:///org/hsqldb/ParserDQL/XreadQueryExpression()|

|java+method:///org/hsqldb/util/DatabaseManager/actionPerformed(java.awt.event.ActionEvent)|

|java+method:///org/hsqldb/ParserDDL/getCompiledStatementBody(org.hsqldb.lib.HsqlList)|

|java+method:///org/hsqldb/index/IndexAVLMemory/delete(org.hsqldb.persist.PersistentStore,org.hsqldb.index.NodeAVL)|
    |java+method:///org/hsqldb/TableWorks/dropConstraint(java.lang.String,boolean)|

|java+method:///org/hsqldb/Session/executeCompiledStatement(org.hsqldb.Statement,java.lang.Object%5B%5D,int)|

|java+method:///org/hsqldb/auth/LdapAuthBean/authenticate(java.lang.String,java.lang.String)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_TYPEINFO(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/test/TestSqlPersistent/testSelectObject()|
    |java+method:///org/hsqldb/cmdline/SqlTool/objectMain(java.lang.String%5B%5D)|

|java+method:///org/hsqldb/StatementDML/update(org.hsqldb.Session,org.hsqldb.Table,org.hsqldb.navigator.RowSetNavigatorDataChange,org.hsqldb.navigator.RowSetNavigator)|

|java+method:///org/hsqldb/test/JDBCBench/createDatabase(java.lang.String,java.lang.String,java.lang.String)|

|java+method:///org/hsqldb/ParserDDL/readColumnConstraints(org.hsqldb.Table,org.hsqldb.ColumnSchema,org.hsqldb.lib.HsqlArrayList)|
    |java+method:///org/hsqldb/DatabaseURL/parseURL(java.lang.String,boolean,boolean)|
    |java+method:///org/hsqldb/persist/Logger/getPropertiesSQL(boolean)|

|java+method:///org/hsqldb/SchemaManager/removeSchemaObject(org.hsqldb.HsqlNameManager.HsqlName)|
    |java+method:///org/hsqldb/test/TestSchemaParse/testSanityCheck()|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/ELEMENT_TYPES(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/StatementSchemaDefinition/getResult(org.hsqldb.Session)|
    |java+method:///org/hsqldb/test/JDBCBench/main(java.lang.String%5B%5D)|

|java+method:///org/hsqldb/ParserDDL/compileCreateTableBody(org.hsqldb.Table,boolean)|

|java+method:///org/hsqldb/FunctionCustom/getValue(org.hsqldb.Session,java.lang.Object%5B%5D)|
    |java+method:///org/hsqldb/util/DatabaseManagerSwing/main()|

|java+method:///org/hsqldb/result/Result/newResult(org.hsqldb.Session,java.io.DataInput,org.hsqldb.rowio.RowInputBinary,int)|

|java+method:///org/hsqldb/ParserDML/compileMergeStatement(org.hsqldb.RangeGroup%5B%5D)|

|java+method:///org/hsqldb/result/Result/write(org.hsqldb.SessionInterface,org.hsqldb.lib.DataOutputStream,org.hsqldb.rowio.RowOutputInterface)|
    |java+method:///org/hsqldb/server/Server/printRequest(int,org.hsqldb.result.Result)|

|java+method:///org/hsqldb/ExpressionLike/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/ParserDDL/readSequenceOptions(org.hsqldb.NumberSequence,boolean,boolean,boolean)|
    |java+method:///org/hsqldb/server/ServerConnection/odbcConnect(int)|
    |java+method:///org/hsqldb/test/TestJDBCSavepoints/testJDBCSavepoints()|
    |java+method:///org/hsqldb/persist/DataFileCache/open(boolean)|
    |java+method:///org/hsqldb/util/ConnectionDialogSwing/create()|
    |java+method:///org/hsqldb/ParserCommand/compileSetTable()|
    |java+method:///org/hsqldb/RangeVariableResolver/expandConditions()|

|java+method:///org/hsqldb/ParserDML/compileInsertStatement(org.hsqldb.RangeGroup%5B%5D)|
    |java+method:///org/hsqldb/ParserDQL/XreadSimpleValueExpressionPrimary()|

|java+constructor:///org/hsqldb/StatementCommand/StatementCommand(int,java.lang.Object%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D)|
    |java+method:///org/hsqldb/ExpressionOp/getValue(org.hsqldb.Session)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/COLUMNS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/ParserCommand/compileSetDatabaseProperty()|
    |java+method:///org/hsqldb/SetFunction/getValue(org.hsqldb.Session)|

|java+method:///org/hsqldb/ParserDDL/readColumnDefinitionOrNull(org.hsqldb.Table,org.hsqldb.HsqlNameManager.HsqlName,org.hsqldb.lib.HsqlArrayList)|
    |java+method:///org/hsqldb/ParserDQL/readTypeDefinition(boolean,boolean)|

|java+method:///org/hsqldb/cmdline/SqlFile/processBuffHist(org.hsqldb.cmdline.sqltool.Token)|
|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_BESTROWIDENTIFIER(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/SchemaManager/getSQLArray()|
    |java+method:///org/hsqldb/ParserDQL/XreadPredicateRightPart(org.hsqldb.Expression)|
    |java+method:///org/hsqldb/cmdline/SqlFile/importDsv(java.lang.String,java.lang.String)|

|java+method:///org/hsqldb/cmdline/SqlFile/displaySqlResults(java.sql.Statement,java.sql.ResultSet,int%5B%5D,java.lang.String,boolean)|

|java+method:///org/hsqldb/RangeVariableResolver/setIndexConditions(org.hsqldb.RangeVariable.RangeVariableConditions,org.hsqldb.lib.HsqlList,int,boolean)|
    |java+method:///org/hsqldb/StatementCommand/getResult(org.hsqldb.Session)|

|java+method:///org/hsqldb/StatementDML/performReferentialActions(org.hsqldb.Session,org.hsqldb.Table,org.hsqldb.navigator.RowSetNavigatorDataChange,org.hsqldb.Row,java.lang.Object%5B%5D,int%5B%5D,org.hsqldb.lib.HashSet)|
    |java+method:///org/hsqldb/cmdline/SqlFile/processPL()|
    |java+method:///org/hsqldb/test/TestDataStructures/testLists()|

|java+method:///org/hsqldb/test/TestBench/createDatabase(java.lang.String,java.lang.String,java.lang.String)|

|java+method:///org/hsqldb/ExpressionLogical/getAllAnyValue(org.hsqldb.Session,java.lang.Object%5B%5D,org.hsqldb.TableDerived)|

|java+method:///org/hsqldb/cmdline/SqlFile/processBlock(org.hsqldb.cmdline.sqltool.Token)|
    |java+method:///org/hsqldb/util/TransferTable/transferStructure()|
 |java+method:///org/hsqldb/cmdline/SqlFile/listTables(char,java.lang.String)|

|java+method:///org/hsqldb/FunctionSQL/getValue(org.hsqldb.Session,java.lang.Object%5B%5D)|

|java+method:///org/hsqldb/cmdline/SqlFile/scanpass(org.hsqldb.cmdline.sqltool.TokenSource)|
    |java+method:///org/hsqldb/ExpressionLogical/testMatchCondition(org.hsqldb.Session)|
    |java+method:///org/hsqldb/ParserDDL/compileCreateTrigger(boolean)|
    |java+constructor:///org/hsqldb/lib/RCData/RCData(java.io.File,java.lang.String)|

|java+method:///org/hsqldb/persist/Logger/backupInternal(java.lang.String,boolean,boolean,boolean,boolean,boolean)|

|java+method:///org/hsqldb/index/IndexAVL/findNode(org.hsqldb.Session,org.hsqldb.persist.PersistentStore,java.lang.Object%5B%5D,int%5B%5D,int,int,int,boolean)|
    |java+method:///org/hsqldb/Session/execute(org.hsqldb.result.Result)|
    |java+method:///org/hsqldb/cmdline/SqlFile/describe(java.lang.String,java.lang.String)|
    |java+method:///org/hsqldb/util/ConnectionDialog/create()|
    |java+method:///org/hsqldb/FunctionCustom/getSQL()|

|java+method:///org/hsqldb/FunctionSQL/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/util/DatabaseManagerSwing/actionPerformed(java.awt.event.ActionEvent)|
    |java+method:///org/hsqldb/StatementSchema/getResult(org.hsqldb.Session)|

|java+method:///org/hsqldb/ExpressionArithmetic/resolveTypesForArithmetic(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/ParserDDL/readFKReferences(org.hsqldb.Table,org.hsqldb.HsqlNameManager.HsqlName,org.hsqldb.lib.OrderedHashSet)|
    |java+method:///org/hsqldb/ExpressionLogical/getSQL()|

|java+method:///org/hsqldb/jdbc/JDBCUtil/sqlException(java.lang.String,java.lang.String,int,java.lang.Throwable)|

|java+method:///org/hsqldb/ParserDQL/XreadTableReference(org.hsqldb.QuerySpecification)|

|java+method:///org/hsqldb/Routine/getMethod(java.lang.String,org.hsqldb.Routine,boolean%5B%5D,boolean)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/CONSTRAINT_COLUMN_USAGE(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/ExpressionColumn/resolveColumnReferences(org.hsqldb.Session,org.hsqldb.RangeGroup,int,org.hsqldb.RangeGroup%5B%5D,org.hsqldb.lib.HsqlList,boolean)|
    |java+method:///org/hsqldb/test/TestSqlPersistent/doTestDoubleNan(boolean)|
    |java+method:///org/hsqldb/Scanner/scanNumber()|

|java+method:///org/hsqldb/types/NumberType/convertToType(org.hsqldb.SessionInterface,java.lang.Object,org.hsqldb.types.Type)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_INDEXINFO(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/persist/ScriptRunner/runScript(org.hsqldb.Database,org.hsqldb.scriptio.ScriptReaderBase)|
    |java+method:///org/hsqldb/QuerySpecification/setUpdatability()|
|java+method:///org/hsqldb/util/TransferSQLText/parseFileForTheRest(org.hsqldb.util.TransferTable,org.hsqldb.util.DataAccessPoint)|
    |java+method:///org/hsqldb/test/TestSql/testMetaData()|

|java+method:///org/hsqldb/util/TransferDb/getTableStructure(org.hsqldb.util.TransferTable,org.hsqldb.util.DataAccessPoint)|
    |java+method:///org/hsqldb/rights/Grantee/getRightsSQL()|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_PROCEDURECOLUMNS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/StatementDML/delete(org.hsqldb.Session,org.hsqldb.Table,org.hsqldb.navigator.RowSetNavigatorDataChange)|
    |java+method:///org/hsqldb/util/CodeSwitcher/processFile(java.lang.String)|
    |java+method:///org/hsqldb/util/Transfer/_main(java.lang.String%5B%5D)|
    |java+method:///org/hsqldb/test/TestAllTypes/testFillUp()|

|java+method:///org/hsqldb/types/DateTimeType/convertToType(org.hsqldb.SessionInterface,java.lang.Object,org.hsqldb.types.Type)|
    |java+method:///org/hsqldb/ParserCommand/compilePart(int)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/SYSTEM_SESSIONS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/cmdline/sqltool/SqlFileScanner/yylex()|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_CROSSREFERENCE(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/map/BaseHashMap/addOrRemove(long,long,java.lang.Object,java.lang.Object,boolean)|
    |java+method:///org/hsqldb/util/DatabaseManagerSwing/directRefreshTree()|

|java+method:///org/hsqldb/ParserRoutine/readRoutineCharacteristics(org.hsqldb.Routine)|

|java+method:///org/hsqldb/TableWorks/addColumn(org.hsqldb.ColumnSchema,int,org.hsqldb.lib.HsqlArrayList)|
    |java+method:///org/hsqldb/persist/Logger/setVariables()|

|java+method:///org/hsqldb/Expression/resolveCheckOrGenExpression(org.hsqldb.Session,org.hsqldb.RangeGroup,boolean)|

|java+method:///org/hsqldb/rowio/RowOutputBinary/getSize(java.lang.Object,org.hsqldb.types.Type)|
    |java+method:///org/hsqldb/test/TestDbBackup/testTarFileNames()|
    |java+method:///org/hsqldb/ParserCommand/compileSetFilesProperty()|
    |java+method:///org/hsqldb/server/ServerConnection/receiveOdbcPacket(char)|
    |java+method:///org/hsqldb/ParserCommand/compileSet()|
    |java+method:///org/hsqldb/ParserRoutine/readDefaultClause(org.hsqldb.types.Type)|
    |java+method:///org/hsqldb/test/TestSql/testDoubleNaN()|
    |java+method:///org/hsqldb/util/TransferSQLText/parseFileForData(java.lang.String)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationMain/SYSTEM_COLUMNS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/QuerySpecification/resolveGroups()|
    |java+method:///org/hsqldb/FunctionSQL/getSQL()|
    |java+method:///org/hsqldb/SetFunction/add(org.hsqldb.Session,java.lang.Object)|

|java+method:///org/hsqldb/ParserRoutine/compileSQLProcedureStatementOrNull(org.hsqldb.Routine,org.hsqldb.StatementCompound)|

|java+method:///org/hsqldb/map/BaseHashMap/addOrRemoveMultiVal(long,long,java.lang.Object,java.lang.Object,boolean,boolean)|
    |java+method:///org/hsqldb/test/TestDatabaseMetaData/testOne()|
    |java+constructor:///org/hsqldb/FunctionSQL/FunctionSQL(int)|
 |java+method:///org/hsqldb/ParserDDL/compileRightGrantOrRevoke(boolean)|
    |java+method:///org/hsqldb/StatementSession/getResult(org.hsqldb.Session)|
    |java+method:///org/hsqldb/ExpressionLogical/getValue(org.hsqldb.Session)|

|java+method:///org/hsqldb/ExpressionLogical/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/PARAMETERS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/USER_DEFINED_TYPES(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+method:///org/hsqldb/ParserDDL/compileDrop()|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/ROUTINES(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|

|java+method:///org/hsqldb/ExpressionLogical/resolveTypesForComparison(org.hsqldb.Session,org.hsqldb.Expression)|
    |java+method:///org/hsqldb/test/TestLikePredicateOptimizations/test()|
    |java+method:///org/hsqldb/cmdline/sqltool/Calculator/reduce(int,boolean)|

|java+method:///org/hsqldb/cmdline/SqlFile/processMacro(org.hsqldb.cmdline.sqltool.Token)|

|java+method:///org/hsqldb/FunctionCustom/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|
|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/DOMAINS(org.hsqldb.Session,org.hsqldb.persist.PersistentStore)|
    |java+constructor:///org/hibernate/dialect/HSQLDialect/HSQLDialect()|

|java+constructor:///org/hsqldb/StatementSchema/StatementSchema(java.lang.String,int,java.lang.Object%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D)|

|java+method:///org/hsqldb/RangeVariableResolver/reorderRanges(org.hsqldb.lib.HsqlArrayList,org.hsqldb.lib.HsqlArrayList)|
    |java+constructor:///org/hsqldb/FunctionCustom/FunctionCustom(int)|
    |java+method:///org/hsqldb/persist/Logger/getValueStringForProperty(java.lang.String)|
    |java+method:///org/hsqldb/Routine/getMethods(java.lang.String)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/generateTable(org.hsqldb.Session,org.hsqldb.persist.PersistentStore,int)|
    |java+method:///org/hsqldb/test/TestAcl/commonSetup()|
```

Complex methods:

```
|java+method:///org/hsqldb/jdbc/JDBCPreparedStatement/setParameter(int,java.lang.Object)|
   |java+method:///org/hsqldb/Scanner/scanToken()|
   |java+method:///org/hsqldb/cmdline/SqlFile/processSpecial(java.lang.String)|

|java+method:///org/hsqldb/util/DatabaseManager/actionPerformed(java.awt.event.ActionEvent)|
   |java+method:///org/hsqldb/cmdline/SqlTool/objectMain(java.lang.String%5B%5D)|
|java+method:///org/hsqldb/DatabaseURL/parseURL(java.lang.String,boolean,boolean)|
   |java+method:///org/hsqldb/persist/Logger/getPropertiesSQL(boolean)|

|java+method:///org/hsqldb/FunctionCustom/getValue(org.hsqldb.Session,java.lang.Object%5B%5D)|

|java+method:///org/hsqldb/ParserDML/compileInsertStatement(org.hsqldb.RangeGroup%5B%5D)|
   |java+method:///org/hsqldb/ParserDQL/XreadSimpleValueExpressionPrimary()|

|java+constructor:///org/hsqldb/StatementCommand/StatementCommand(int,java.lang.Object%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D)|
   |java+method:///org/hsqldb/ExpressionOp/getValue(org.hsqldb.Session)|

|java+method:///org/hsqldb/ParserDDL/readColumnDefinitionOrNull(org.hsqldb.Table,org.hsqldb.HsqlNameManager.HsqlName,org.hsqldb.lib.HsqlArrayList)|
    |java+method:///org/hsqldb/ParserDQL/readTypeDefinition(boolean,boolean)|

|java+method:///org/hsqldb/cmdline/SqlFile/processBuffHist(org.hsqldb.cmdline.sqltool.Token)|
    |java+method:///org/hsqldb/cmdline/SqlFile/importDsv(java.lang.String,java.lang.String)|

|java+method:///org/hsqldb/cmdline/SqlFile/displaySqlResults(java.sql.Statement,java.sql.ResultSet,int%5B%5D,java.lang.String,boolean)|
    |java+method:///org/hsqldb/StatementCommand/getResult(org.hsqldb.Session)|
    |java+method:///org/hsqldb/cmdline/SqlFile/processPL()|

|java+method:///org/hsqldb/cmdline/SqlFile/processBlock(org.hsqldb.cmdline.sqltool.Token)|
    |java+method:///org/hsqldb/cmdline/SqlFile/listTables(char,java.lang.String)|

|java+method:///org/hsqldb/FunctionSQL/getValue(org.hsqldb.Session,java.lang.Object%5B%5D)|

|java+method:///org/hsqldb/cmdline/SqlFile/scanpass(org.hsqldb.cmdline.sqltool.TokenSource)|
    |java+method:///org/hsqldb/ParserDDL/compileCreateTrigger(boolean)|
    |java+method:///org/hsqldb/FunctionCustom/getSQL()|

|java+method:///org/hsqldb/FunctionSQL/resolveTypes(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/util/DatabaseManagerSwing/actionPerformed(java.awt.event.ActionEvent)|
    |java+method:///org/hsqldb/StatementSchema/getResult(org.hsqldb.Session)|

|java+method:///org/hsqldb/ExpressionArithmetic/resolveTypesForArithmetic(org.hsqldb.Session,org.hsqldb.Expression)|

|java+method:///org/hsqldb/util/TransferDb/getTableStructure(org.hsqldb.util.TransferTable,org.hsqldb.util.DataAccessPoint)|
    |java+method:///org/hsqldb/cmdline/sqltool/SqlFileScanner/yylex()|
    |java+method:///org/hsqldb/server/ServerConnection/receiveOdbcPacket(char)|
    |java+method:///org/hsqldb/ParserRoutine/readDefaultClause(org.hsqldb.types.Type)|
    |java+method:///org/hsqldb/FunctionSQL/getSQL()|

|java+method:///org/hsqldb/ParserRoutine/compileSQLProcedureStatementOrNull(org.hsqldb.Routine,org.hsqldb.StatementCompound)|
    |java+method:///org/hsqldb/ParserDDL/compileRightGrantOrRevoke(boolean)|
    |java+method:///org/hsqldb/StatementSession/getResult(org.hsqldb.Session)|
    |java+method:///org/hsqldb/ExpressionLogical/getValue(org.hsqldb.Session)|
|java+method:///org/hsqldb/ExpressionLogical/resolveTypes(org.hsqldb.Session,org.hsqldb.
Expression)|

|java+method:///org/hsqldb/FunctionCustom/resolveTypes(org.hsqldb.Session,org.hsqldb.Ex
pression)|

|java+constructor:///org/hsqldb/StatementSchema/StatementSchema(java.lang.String,int,jav
a.lang.Object%5B%5D,org.hsqldb.HsqlNameManager.HsqlName%5B%5D,org.hsqldb.Hsql
NameManager.HsqlName%5B%5D)|
   |java+constructor:///org/hsqldb/FunctionCustom/FunctionCustom(int)|
   |java+method:///org/hsqldb/persist/Logger/getValueStringForProperty(java.lang.String)|

|java+method:///org/hsqldb/dbinfo/DatabaseInformationFull/generateTable(org.hsqldb.Sessio
n,org.hsqldb.persist.PersistentStore,int)|

Percentages of LOC per risk category for Unit Size:
Percentage very high:   16.92170876000
Percentage high:        14.12799462000
Percentage medium:      35.72613793000
Percentage low:         17.22591560000
Percentage LOC outside methods: 15.99824309000

Percentages of LOC per risk category for Unit Complexity:
Percentage very high:   8.173457547000
Percentage high:        9.721062392000
Percentage medium:      11.98499029000
Percentage low:         54.12224668000
Percentage LOC outside methods: 15.998243091000

Results
--------------------------
Volume:             +
Unit Size:          --
Unit Complexity:      --

Analysability:  o
Changeability:  --
Testability:    --

Overall maintainability: -
```

Testing

output of the tests, which are done by loading all the modules and doing :test in the rascal terminal. They depend on the java test_project, so this project should be in the workspace.

```
rascal>:test
Reloading module UnitComplexity
Running tests for LinesOfCode
Test report for LinesOfCode success
        all 1/1 tests succeeded
Running tests for UnitComplexity
Test report for UnitComplexity
        all 3/3 tests succeeded
bool: true
```

Time metrics

output of the time metrics, in miliseconds. It does the methods, so method output is also printed but I leave this away for now, the actual results:

```
rascal>getMetricsTiming();
map[str, num]: ("hsqldb":648879,"testProject":124,"smallsql":245901)
```

If we go from miliseconds to minutes (/1000, /60)
hsqldb takes about 11 minutes, without code duplication
smallsql takes about 4 minutes, including code duplication

Proposal for new metric

CBO coupling between classes:
amount of dependencies/amount of classes/modules

Why would this be a good measure?
because dependencies mean that something can't be changed easily, because a change could have influence on the working of the complete system. With a lot of dependencies, you might not see which class depends on which. This means you can change something that is used somewhere else so you break the program, or you have to do a lot of research before changing something to validate nothing depends on it to work a certain way. Some coupling is of course always necessary but a lot of coupling makes changing code very difficult and since maintaining code involves making changes, it makes maintainability more difficult. According to Jabangwe, the CBO metric had a significant positive relationship with maintainability in 60 different dataset, in multiple different articles.It is not implemented here, but there are possibilities for doing that using an AST.

source: Jabangwe, Ronald, et al. "Empirical evidence on the link between object-oriented measures and external quality attributes: a systematic literature review." *Empirical Software Engineering* 20.3 (2015): 640-693. (http://link.springer.com/article/10.1007/s10664-013-9291-7)

Maintainability of the own code

Our code has a total of 21 methods, from which 7 have more than 10 lines of code. The biggest method is 49 lines of code, just below the high risk percentage. This method, the duplication method, is also the only method with a complexity higher than 10 (11). In total there are 4 test methods. They do not test everything this project does, because a lot of methods print out the results instead of returning them. This makes the project less testable, but the output better. Also there is some dependence