ARTICLES

Reaction-MQL: Line Notation for Functional Transformation

Felix H. Reisen, Gisbert Schneider, and Ewgenij Proschak*

Institute of Organic Chemistry and Chemical Biology, Goethe-University Frankfurt, Siesmayerstrasse 70, D-60323 Frankfurt am Main, Germany

Received June 27, 2008

Representation of chemical reactions is pivotal for different purposes in cheminformatics. We present an extension of the molecular query language (MQL), which combines readable style with meaningful rules for string representation of reactions and unambiguous product formation. The concept of functional groups is used to describe the transformations. Functional groups are defined in terms of substructure queries and are processed by graph transformations. Molecular educt graphs are transformed by application of beginning-, end-, and reaction-matrices to obtain the product graph without consideration of stereochemistry. Both directions of a transformation are possible. We implemented the concept of Reaction-MQL in Java employing the Chemistry Development Kit.

INTRODUCTION

The representation of chemical reactions is an important subject of cheminformatics.[1] Several theories and concepts have been developed to interpret the outcome and underlying mechanisms of chemical reactions. These theories are based on physicochemical effects, e.g. charge distribution of the participating molecules. For the representation of reactions *in silico* the description of molecular properties can be transferred into computer models.[2] These models are used for different applications such as information storage in databases,[3] reaction classification[4] and prediction,[5] synthesis planning[6,7] and evaluation of synthetic accessibility,[8] modeling of metabolic processes,[9] and the creation of virtual combinatorial compound libraries.[10,11] Computer based representations of chemical reactions require correct and unambiguous descriptors for the participating molecules.[1] Additionally, exact declarations of bond and atom changes that occur during the reaction process have to be denoted.

Rxnfiles and RDfiles, which are part of the Symyx's Ctfiles,[12] are common formats for storage of reactions. Both are based on the„mol"-format that describes molecules in terms of adjacency lists.[13] Sankar and Aghila proposed an XML-derived reaction storage format,[14] and molecule drawing software like ChemDraw,[15] ISIS/Draw,[16] and ChemSketch[17] offer a graphical input and storage system. The linear text-based language SMILES,[18] which was originally developed for the description of molecules, also allows for the specification of reactions by its extension Reaction SMILES. The SMIRKS language[19] enables the formulation of reaction expressions by means of chemical substructures. The software Reactor by ChemAxon[20] provides a possibility to apply reaction rules to a set of molecules given in different molecular representation formats such as SMILES or the "mol"-format.

Here, we introduce a novel line-notation for the description of molecular transformations. This line notation, called "Reaction-MQL", is based on the Molecular Query Language (MQL)[21] and allows for an intuitive definition of transformation schemes.

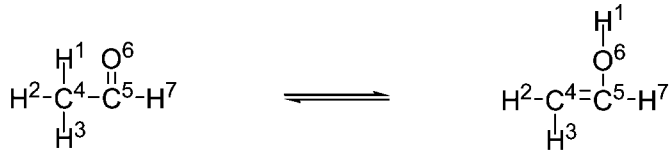
To simplify the handling of physicochemical properties responsible for the behavior of molecules in a reaction, only functional groups are considered. This is possible because these groups are the reacting parts of the molecules, although the whole molecular structure influences reactivity. Functional groups which participate in the reaction are annotated as MQL-queries. Graph transformations are used for the underlying reaction engine. Here, we present the implementation and application of Reaction-MQL with the Chemistry Development Kit (CDK).[22]

THEORETICAL BACKGROUND OF THE SOFTWARE IMPLEMENTATION

In terms of graph theory, reaction schemes are transformation rules for molecular graphs. Transformation rules can be represented by *B*-, *E*-, and *R*-matrices (*B*: beginning, *E*: end, *R*: reaction). This concept was introduced by Dugundji and Ugi more than 30 years ago.[23] *B*- and *E*-matrices encode for the educts and products, respectively. Unique indices are assigned to each atom: In a *B*- and an *E*-matrix, the entries $b_{ij,i\neq j}$ and $e_{ij,i\neq j}$ correspond to the order of a bond between the atoms with index *i* and *j*. Diagonal entries $b_{ij,i=j}$ and $e_{ij,i=j}$ represent the number of lone pair electrons of an atom with index *i*. It is possible to deduce the associated molecular graphs G_B and G_E of the educts and the products from the *B*- and *E*-matrices. The *R*-matrix contains information about bond changes during a transformation, so that $B + R = E$. With regard to the molecular constitution, these matrices allow a correct description of molecular transformations. Figure 1 shows an example for the use of the matrices by means of an enolization.

* Corresponding author e-mail: proschak@bioinformatik.uni-frankfurt.de.

10.1021/ci800215t CCC: $40.75
Published on Web 12/18/2008



B	1	2	3	4	5	6	7
1	0	0	0	1	0	0	0
2	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0
4	1	1	1	0	1	0	0
5	0	0	0	1	0	2	1
6	0	0	0	0	2	4	0
7	0	0	0	0	1	0	0

B +

R	1	2	3	4	5	6	7
1	0	0	0	-1	0	1	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	-1	0	0	0	1	0	0
5	0	0	0	1	0	-1	0
6	1	0	0	0	-1	0	0
7	0	0	0	0	0	0	0

R =

E	1	2	3	4	5	6	7
1	0	0	0	0	0	1	0
2	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0
4	0	1	1	0	2	0	0
5	0	0	0	2	0	1	1
6	1	0	0	0	1	4	0
7	0	0	0	0	1	0	0

E

Figure 1. Representation of an enolization by *B*-, *E*-, and *R*-matrices.

For Reaction-MQL we extended the theories of Dugundji and Ugi for the description of chemical reaction schemes by functional groups. In this case, *B* and *E* have to contain the constitutional information about the different functional groups before and after the transformation. *R* covers changes in the reaction center, namely the reacting functional groups. We assume that the connected components of the graphs G_B and G_E can be used to verify whether a set of molecules *M* given as molecular graphs can react according to a given transformation scheme by means of subgraph isomorphisms. This is the case, if every connected component of G_B is a subgraph of at least one of the given molecular graphs: Let B' be a matrix that represents the constitution of all members of *M*. Let further $|b|$ and $|b'|$ be the number of lines and rows of matrix *B* and B', respectively. We use the associated subgraph isomorphism to permutate the rows and lines of matrix B', so that the submatrix *S* of B' composed out of the first $|b|$ lines and rows of B' is equal to *B* (Figure 2). Let B_S' be the permutated matrix B'. The transformation can then be applied by adding *R* to *S*, without changing the entries outside of *S*. This can formally be done by creating a matrix $R' = (r'_{ij})_{1 \le i,j \le |b'|}$, with

$$r'_{ij} = \begin{cases} r'_{ij}, \text{ if } i,j \le |b| \\ 0 \text{ else.} \end{cases}$$

The product matrix E' is obtained by adding R' to B_S'. The resulting molecular graphs can then be deduced by E' (Figure 2).

B-, *E*-, and *R*-matrices can be extended to encode more information than bond orders and bond order changes. Therefore, let *B**, *E**, and *R** be extended matrices that are able to store additional information about bonds. Such information can be for instance bond length, topological information, or aromaticity. Besides *B**, *E**, and *R**, vectors B_A, R_A, and E_A for atoms of the described substructures have to be introduced. They contain information about different atom properties of the substructures. Examples for these atom properties are partial charge or hybridization state.

For the application of such an extended transformation scheme a subgraph isomorphism has to be found in a similar way as with the basic model. For this purpose, the bonds and atoms of the graphs are colored by the different properties. Accordingly, the found isomorphism is then used to resort the $B^{*\prime}$-matrices and the B_A'-vectors that encode educt properties. Using numeric encoding of non-numeric properties allows for adding the $R^{*\prime}$-matrix to $B^{*\prime}$ to apply a transformation.

This extended model of the *B*-, *E*-, and *R*-matrices provides the theoretical background of the representation of reactions in the Reaction-MQL-package.

MQL

The Molecular Query Language (MQL) is a context-free grammar[24] for a linear text-based description of chemical substructures.[21] MQL provides the opportunity to define chemical query structures and enables searching for these structures in given molecules. The general syntax of a MQL-expression is

$$<\text{atom}>(<\text{bond}><\text{atom}>)_{n,n\ge 0}$$

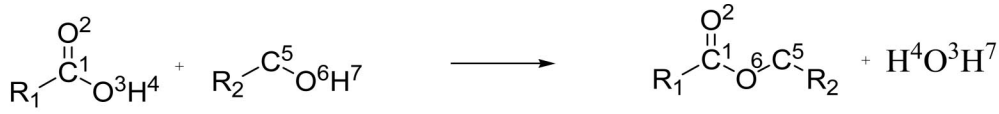
where the placeholders "<atom>" and "<bond>" are substituted by appropriate atom respectively bond descriptions. These descriptions are formulated through a property keyword system.

GENERAL STRUCTURE OF A REACTION-MQL-EXPRESSION

Reaction-MQL is a text-based language for the description of molecular transformation schemes. Its syntax is similar to a typical description of a reaction by a chemical equation. Therefore, it consists of educts, a reaction arrow, and products. Both, educts and products are represented by MQL-expressions and separated by the symbol "++". The reaction arrow is a combination of the symbols "≪" and "≫". Depending on the directionality (uni- or bidirectional) it starts either with "≪" or "≫" and ends with "≫". In between these symbols any types of annotations are allowed. For a clear distinction between the individual parts of a Reaction-MQL-expression it is possible to set space characters. Figure 3 shows the formal structure of a Reaction-MQL-expression.

MAPPING BETWEEN PRODUCTS AND EDUCTS

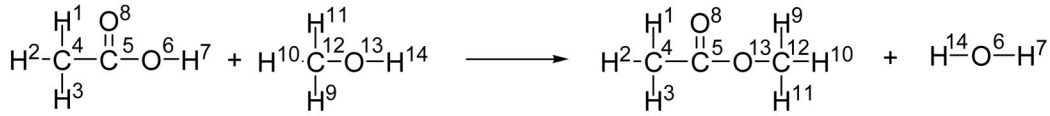
For the correct and unambiguous description of a transformation, atom-mapping between products and educts is



B	1	2	3	4	5	6	7
1	0	2	1	0	0	0	0
2	2	4	0	0	0	0	0
3	1	0	4	1	0	0	0
4	0	0	1	0	0	0	0
5	0	0	0	0	0	1	0
6	0	0	0	0	1	4	1
7	0	0	0	0	0	1	0

R	1	2	3	4	5	6	7
1	0	0	-1	0	0	1	0
2	0	0	0	0	0	0	0
3	-1	0	0	0	0	0	1
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0
6	1	0	0	0	0	0	-1
7	0	0	1	0	0	-1	0

E	1	2	3	4	5	6	7
1	0	2	0	0	0	1	0
2	2	4	0	0	0	0	0
3	0	0	4	1	0	0	1
4	0	0	1	0	0	0	0
5	0	0	0	0	0	1	0
6	1	0	0	0	1	4	0
7	0	0	1	0	0	0	0



B'	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	0	0	0
4	1	1	1	0	1	0	0	0	0	0	0	0	0	0
5	0	0	0	1	0	1	0	2	0	0	0	0	0	0
6	0	0	0	0	1	4	1	0	0	0	0	0	0	0
7	0	0	0	0	0	1	0	0	0	0	0	0	0	0
8	0	0	0	0	2	0	0	4	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	1	0	0
11	0	0	0	0	0	0	0	0	0	0	0	1	0	0
12	0	0	0	0	0	0	0	0	1	1	1	0	1	0
13	0	0	0	0	0	0	0	0	0	0	0	1	4	1
14	0	0	0	0	0	0	0	0	0	0	0	0	1	0

v	$f_{1,1}^{-1}(v)$
5	1
6	3
7	4
8	2

v	$f_{2,2}^{-1}(v)$
12	5
13	6
14	7

B_S'	5	8	6	7	12	13	14	2	9	10	11	1	3	4
5	0	2	1	0	0	0	0	0	0	0	0	0	0	1
8	2	4	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	4	1	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	1	0	0	1	1	1	0	0	0
13	0	0	0	0	1	4	1	0	0	0	0	0	0	0
14	0	0	0	0	0	1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4	1	0	0	0	0	0	0	1	0	0	0	1	1	0

R'	5	8	6	7	12	13	14	2	9	10	11	1	3	4
5	0	0	-1	0	0	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	-1	0	0	0	0	0	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	0	0	0	0	0	-1	0	0	0	0	0	0	0
14	0	0	1	0	0	-1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$B_S' + R'$

E'	5	8	6	7	12	13	14	2	9	10	11	1	3	4
5	0	2	0	0	0	1	0	0	0	0	0	0	0	1
8	2	4	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	4	1	0	0	1	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	1	0	0	1	1	1	0	0	0
13	1	0	0	0	1	4	0	0	0	0	0	0	0	0
14	0	0	1	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4	1	0	0	0	0	0	0	1	0	0	0	1	1	0

Figure 2. A transformation scheme (a) and its application to given molecules (b). B' contains the educt molecules. $f_{1,1}$ and $f_{2,2}$ are the subgraph isomorphisms that map the functional groups of B to the given molecules in B'. B_S' was created by using the subgraph isomorphisms for the permutation of the lines and rows of B'. The product matrix E' results from the addition of R' to E'. The reaction center is highlighted in the matrices B, R, E, B_S', R', and E'.

required. MQL-identifiers allow the assignment of unambiguous integer labels to atoms. Direct mapping from an educt atom to a product atom is only needed when its properties or its bonds are changed during the transformation. Therefore, atoms and bonds are classified into those belonging to the so-called "reaction center" and others. The reaction

a: MQL (++ MQL)$_n$ >> reaction terms >> MQL (++ MQL)$_n$
b: MQL (++ MQL)$_n$ << reaction terms >> MQL (++ MQL)$_n$

Figure 3. General structure of Reaction-MQL-expressions. Expressions can be defined as unidirected (a) or bidirected (b).

center consists of all atoms and bonds whose properties are explicitly changed during the Reaction-MQL transformation. In addition, the reaction center comprises unchanged bonds if both associated atoms are part of it. An identifier is obligatory for all atoms of the reaction center except for atoms that appear on the product side only.

Atoms and bonds that are not part of the reaction center merely serve as an exact description of the functional group. It is not necessary to mention them on the product side again because they are not changed during the transformation. Atoms and bonds which do not have an identifier are not rewritten in the product side of Reaction-MQL-expressions. Identifiers can be assigned to these atoms optionally.

BOND CHANGES

Bond changes are encoded by differences between properties of bonds in the corresponding educt and product Reaction-MQL-expressions. If the educts contain a bond between two atoms of the reaction center which is not present in the products it vanishes during the transformation. Accordingly a bond between two atoms of the reaction center is formed when it appears on the product side only.

STOICHIOMETRY

Reaction-MQL-expressions are not necessarily stoichiometrically correct. Since not all products are of interest for the user, it is possible to omit molecules on the product side. An atom "disappears" if it has an identifier on the educt side but is absent on the product side. The example in Figure 4a explains this feature: the ester oxygen O^2 is part of the reaction center and does not appear on the product side. Therefore all bonds from O^2 to atoms of the reaction center are broken (the bond to C^1). This does not affect bonds outside the reaction center (the bond to the ethyl moiety). O^2 vanishes together with all atoms that remain connected to it.

Disappearing atoms can still be part of the output, when they are connected indirectly to remaining reaction center atoms after transformation. The handling of this special case is exemplary shown in Figure 4b. Although, the direct connection of O^2 and C^1 is broken on the product side, O^2 still remains, due to its indirect connection to C^1 by four carbon atoms ($O^2-C-C-C-C-C^1$).

It is also possible to add new atoms on the product side of a reaction. These new atoms do not have an identifier (Figure 4c). Note, that the "disappearing" hydrogen is implicitly treated and does not explicitly participate in the transformation.

PROPERTIES

Besides the declaration of bond changes it is sometimes necessary to assign distinct attributes to atoms and bonds. This is done via the property assignment of MQL. Property changes are annotated by differences between corresponding educt and product MQL-expressions. When properties are not explicitly formulated in the description of bonds and atoms, default values are assigned (Table 1). Therefore, it is necessary to rewrite nonstandard property values if they are not changed during the transformation.

RESTRICTIONS

Reaction-MQL-expressions have to be interpretable by the computer in a particular way. MQL enables a user to formulate general atom and bond descriptions that may result in multiple products (Figure 5). For this reason only restricted MQL-expressions may be used for Reaction-MQL.

To avoid ambiguities but to keep the possibility to model a broad range of reaction types, property keywords of MQL are classified in three different categories: The first category contains so-called "setable" keywords which enable the user to set different property values such as "charge", "explicitHydrogens", "implicitHydrogens", "allHydrogens", "aromatic", and "aliphatic". The second category contains "checkable" features: Keywords belonging to this group are "bound", "ring", "valence", "totalConnection", "explicitConnections, "sp1", "sp2", and "sp3. These keywords can be employed for a detailed annotation of functional groups on the educt side. Furthermore they can be used on the product side to restrict certain transformations, such as improbable hybridization states. Transformations are carried out only if the product molecules fulfill the postulated criteria of the MQL-expressions. The last group of keywords contains element symbols. Additionally it contains keywords for atom sets such as "Heavy", "Halogen", or "Hetero".

Generally, restrictions are required only for atoms and bonds in the reaction center. Other atoms and bonds do not underlie any MQL-specific constraints. The use of "checkable" features does not need any constraints, either. Setable features have to be utilized in a distinct way in the reaction center. As a consequence, the operators '<' and '>' must not be used for atoms and bonds that are changed during a transformation. With the exception of Boolean setable features the combination of the negation symbol '!' with a setable feature is not allowed. Disjunctions by the boolean operator '|' (logical "OR") are forbidden.

Element symbols have to be handled in a distinct way, too. Element type changes are forbidden during a transformation. Descriptions of atoms which are created during a transformation are restricted to only one element type. For example, the MQL-expression "C(=O)-Halogen" generally describes the group of carbon acid halogenides. When this group is formed during a transformation the computer has several opportunities to replace the container atom "Halogen" by an explicit halogen atom such as chlorine. Consequently, atoms which are only part of the educts have to be well-defined in bidirected transformation schemes, because they are added when the backward transformation is applied. Note, that "vanishing" atoms in a unidirected transformation are not subject to these restrictions. For a better understanding of the different use of property types Table 2 presents some examples of allowed and forbidden expressions.

HYDROGEN HANDLING

Hydrogen handling is done implicitly as long as no explicit information is given by means of "setable" features. After a transformation, atoms have bound as many hydrogen atoms as defined by standard valence properties. To create atoms

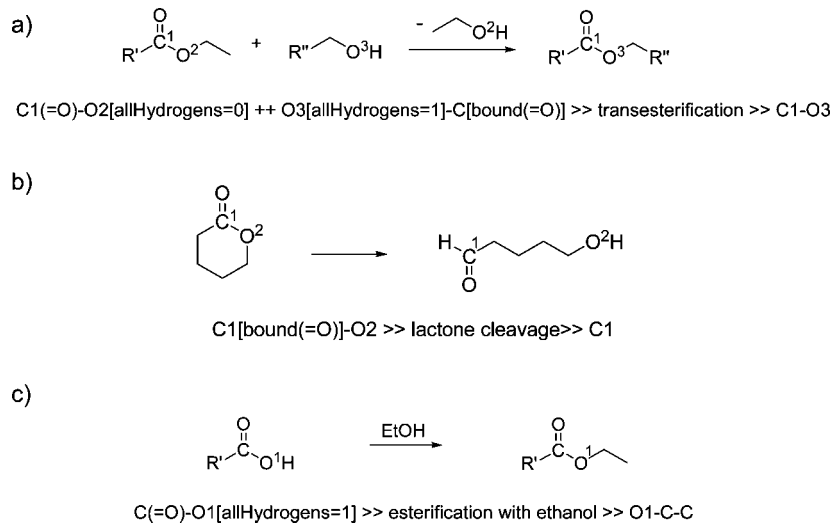


Figure 4. Stoichiometrically unbalanced Reaction-MQL expressions. a) Ester formation where ethanol is regarded as an un-needed byproduct. b) Lactone cleavage, where O^2 is indirectly connected to an atom with an identifier (C^1) on the product side. Therefore all atoms remain after the transformation. c) Ester formation where ethanol comes from the solvent.

Table 1. Standardized Keywords That Are Used in the Reaction Center As Long As No Other Values Are Given Explicitly

keyword	default value
aromatic	FALSE
aliphatic	TRUE
charge	0
allHydrogens	*depends on atom type*
valence	*depends on atom type*

with nontypical valences "setable" features such as "allHydrogens" may be used (Figure 6).

SOFTWARE

The Reaction-MQL-package was implemented in Java.[25] This allows for use on a variety of operating systems. The package employs the MQL-library for defining and searching of substructures. Furthermore it needs a cheminformatics toolkit for parsing and handling of molecules. The MQL-library was designed to operate with different cheminformatics toolkits. Because changes in the MQL-library should be adoptable for the Reaction-MQL-package as well, it can be used with different cheminformatics toolkits, too. Additionally, the Reaction-MQL-package can be adjusted to changes in the keyword system of the MQL. A workflow of the software package of Reaction-MQL is given in Figure 7. Two different molecule representations are used in the Reaction-MQL package. First, the molecule structure of the CDK[22] package is employed for handling input and output. The CDK package is employed for handling input and output, and then matrices are used for the substructure matching algorithm and for storage and application of transformation schemes.

In its current state, the Reaction-MQL-package provides the user the functionality of molecule transformation. Given a set of transformation schemes and molecules the package is able to enumerate a full combinatorial database of all possible product molecules.

C-C1~[order=3]C2-C >> >> C1~[order<3]C2

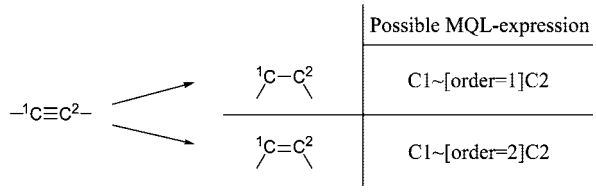


Figure 5. Example for an ambiguous use of MQL-properties in a transformation expression. The term,,order < 3" in the reaction center allows the outcome of different product molecules. These kinds of expressions are not allowed in Reaction-MQL.

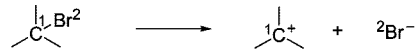
Table 2. Examples for Allowed and Forbidden Expressions in Reaction-MQL

allowed expression	forbidden expression
C1[ring|bound(!=O)]	C1[aromatic|allHydrogens=1]
C1[aromatic&{ring|bound(!=O)}]	C1[aromatic&{aromatic|allHydrogens=1}]
C1[!aromatic]	C1[!allHydrogens=1]
C1[!{ring & !allHydrogens=1}]	C1[!{ring and allHydrogens=1}]
C1[charge=2]	C1[charge < 2]
C~[{ring|aromatic}&order < 3]C1	C1 ~[{ring|aromatic}&order < 3]C2
*1[C|N]≫ ≫*1[C|N]	*1[C|N] ≫ ≫ C1

DISCUSSION AND OUTLOOK

A motivation for the development of Reaction-MQL was to define a language that is easy to learn and generally applicable to organic reactions. MQL provides the opportunity of an intuitive description of chemical substructures and therefore is suited as a basis for an easy-to-learn transformation language. Since MQL is extendable by further keywords, it provides generality. The syntax of Reaction-MQL is similar to the formulation of a chemical reaction by a reaction equation consisting of educts, a reaction arrow, and products. This alleviates the direct deduction of transformation schemes from given reactions.

In Reaction-MQL-expressions the reaction center is defined by a particular set of identifiers. This ensures that transformation schemes are unambiguous. Descriptions of transformation schemes with Reaction-MQL can become long since MQL-expressions tend to be long as well. Nevertheless, we developed the possibility of keeping them short with only atoms and bonds of the reaction center having to be rewritten on the product side. If all atoms are involved in the reaction center like in a Diels−Alder reaction or a [1,5]sigmatropic rearrangement, the number of atoms on the



C-C1(-C)-Br2 >> >> C1[allHydrogens=0&charge=1] ++ Br2[allHydrogens=0&charge=-1]

Figure 6. Creation of molecules that contain atoms with unusual valence. Automatic hydrogen assignment is avoided by the explicit use of the keyword,,allHydrogens".

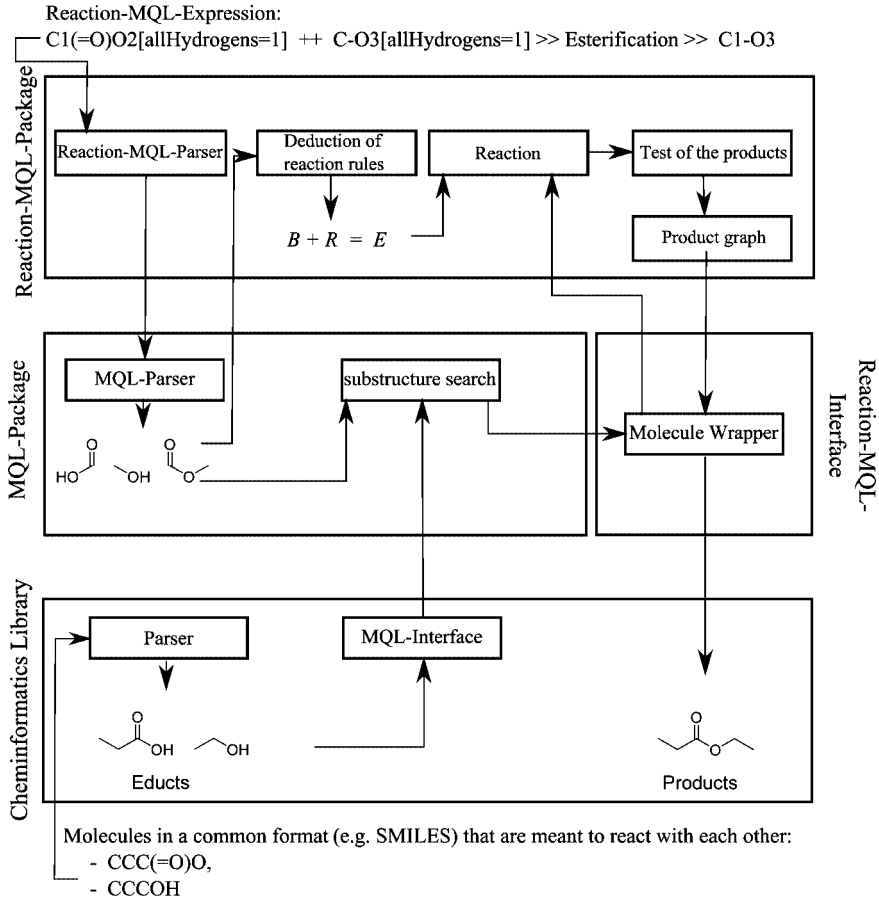


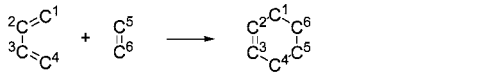
Figure 7. Workflow of a transformation with Reaction-MQL. The text-based Reaction-MQL expression is processed, and the transformation rules are deduced in the Reaction-MQL-package. The extracted MQL-queries are used to find the needed substructures in a set of given molecules. These molecules are handled via an arbitrary cheminformatics library (e.g., CDK). The molecules with the found substructures are translated into a Reaction-MQL specific educt graph by the molecule wrapper software, then the transformation is applied, and the resulting products are tested whether they are part of the output. This depends on the MQL expressions on the product side. The final product graph is then translated into the molecule representation of the particular cheminformatics library used.

product side has to be equal to the educt side (Figure 8a,b). An example for an especially concise definition of an intramolecular ring formation is given in Figure 8c. Generally, intramolecular cyclizations are possible if the atoms and bonds between the functional groups of the educt molecule are specified (wildcard placeholders are sufficient). This ensures that evolved cycles have the appropriate size. In case of multiple occurrences of functional groups in one molecule, our implementation allows only one moiety to react in a single transformation step. To enforce additional transformations, the product molecule has to undergo the transformation again. A particular drawback of the current implementation is that stereochemical aspects are neglected because MQL does not allow for their representation yet.

Reaction-MQL could be used for storage and managing transformation schemes in a database. Creation of combinatorial libraries by using a set of transformation schemes on a set of start molecules is another application on which our group is working currently. These libraries can be employed for virtual screening and fragment-based de novo design.[26−28] Resulting hits will have the advantage that they are synthesizable if the according Reaction-MQL-expressions model a viable reaction and the starting materials are accessible. Also, it is possible to discover new building blocks for a fragment based *de novo* design approach by fragmenting molecule databases according to a set of implemented *retro*-synthetic transformation schemes, like RECAP.[29] Reaction-MQL can also be utilized to transform molecules into a consistent protonation state. Moreover, it could be applied in the field of computer-aided synthesis planning or in the analysis and prediction of metabolic or chemical networks.[30]

In the future, it will be of interest to improve the language definition. Handling of sterochemical aspects is one of these import tasks since many chemical reactions proceed only with a clearly defined steoreochemistry.[1] In addition, new keywords can be implemented for MQL that describe physicochemical properties that are important for chemical

a)

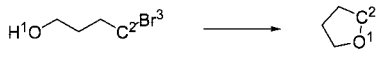


C1=C2-C3=C4 ++ C5=C6 << Diels-Alder cyclisation >> C1$1-C2=C3-C4-C5-C6$-1

b)

C1-C2=C3-C4=C5 <<[1,5]sigmatropic shift >> C1=C2-C3=C4-C5

c)



O1[allHydrogens=1]-C[!bound(=O)]-C-C-C2-Br3 >> ring closure >> O1-C2

Figure 8. Examples for the formulation of transformations with Reaction-MQL. a) In the Diels−Alder cyclization all mentioned atoms are part of the reaction center. Therefore, all atoms have to be rewritten on the product side. b) [1,5]Sigmatropic rearrangement is shown. Note, that hydrogen atoms are handled implicitly. c) The Reaction-MQL-expression of the intramolecular cyclization is short, because atoms that are not part of the reaction center are not rewritten on the product side.

reactions. These keywords may describe electron donating or withdrawing substituents, partial charges, or groups that can stabilize radicals. Due to the modular software architecture used for the implementation of Reaction-MQL such modifications and extensions can be introduced with minimal effort.

ACKNOWLEDGMENT

The authors are grateful to Yusuf Tanrikulu and Andreas Schüller for helpful discussions and technical support. This study was supported by the Beilstein-Institut zur Förderung der Chemischen Wissenschaften, Frankfurt am Main, by the LOEWE Lipid Signaling Forschungszentrum Frankfurt (LiFF) and the Deutsche Forschungsgemeinschaft (SFB 579, project A11.2).

REFERENCES AND NOTES

(1) Gasteiger, J.; Engel, T. *Chemoinformatics A Textbook*; Wiley-VCH: Weinheim, 2003; pp169−200.
(2) Kerber, A.; Laue, R.; Meringer, M.; Rücker, C. Molecules in Silico: A Graph Description of Chemical Reactions. *J. Chem. Inf. Model.* **2007**, *47*, 805–817.
(3) Meehan, P.; Schofield, H. CrossFire: a structural revolution for chemists. *Online Inf. Rev.* **2001**, *25*, 241–249.
(4) Chen, L.; Gasteiger, J. Knowledge Discovery in Reaction Databases: Landscaping Organic Reactions by a Self-Organizing Neural Network. *J. Am. Chem. Soc.* **1997**, *119*, 4033–4042.
(5) Komeiji, Y.; Ishikawa, T.; Mochizuki, Y.; Yamataka, H.; Nakano, T. Fragment Molecular Orbital Method-Based Molecular Dynamics (FMO-MD) as a Simulator for Chemical Reactions in Explicit Solvation. *J. Comput. Chem.* **2008**, . in press.
(6) Hendrickson, G. J. B. The SYNGEN approach to synthesis design. *Anal. Chim. Acta* **1990**, *235*, 103–113.
(7) Gasteiger, J.; Ihlenfeldt, W. D.; Röse, P.; Wanke, R. Computer-assisted Reaction Prediction and Synthesis Design. *Anal. Chim. Acta* **1990**, *235*, 65–75.
(8) Boda, K.; Seidel, T.; Gasteiger, J. Structure and reaction based evaluation of synthetic accessibility. *J. Comput.-Aided Mol. Des.* **2007**, *21*, 311–325.
(9) Apostolakis, J.; Sacher, O.; Körner, R.; Gasteiger, J. Automatic Determination of Reaction Mappings and Reaction Center Information. 2. Validation on a Biochemical Reaction Database. *J. Chem. Inf. Model.* **2008**, *48*, 1190–1198.
(10) Aronov, A. M.; Narsimha, R. M.; Kuntz, I. D.; Wang, C. C. Virtual Screening of Combinatorial Libraries across a Gene Family: in Search of Inhibitors of Giardia lamblia Guanine Phosphoribosyltransferase. *Antimicrob. Agents Chemother.* **2001**, *45*, 2571–2576.
(11) Schüller, A.; Hähnke, V.; Schneider, G. SmiLib v2.0: A Java-Based Tool for Rapid Combinatorial Library Enumeration. *QSAR Comb.Sci.* **2007**, *3*, 407–410.
(12) Symyx, 3100 Central Expressway, Santa Clara, CA 95051.http://www.mdli.com/downloads/public/ctfile/ctfile.jsp (accessed October 14, 2008).
(13) Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C. *Introduction to Algorithms*; MIT Press: Cambridge, 2003; pp 528−529.
(14) Sankar, P.; Aghila, G. Ontology Aided Modeling of Organic Reaction Mechanisms with Flexible and Fragment Based XML Markup Procedures. *J. Chem. Inf. Model.* **2007**, *47*, 1747–1762.
(15) *ChemDraw Ultra, version 7.0.1*; CambridgeSoft: Cambridge, U.K., 2008.
(16) *ISIS/Draw, version 2.5*; MDL: San Leandro, CA, 2007.
(17) *ChemSketch, version 11.0*; Advanced Chemistry Development: Toronto, Canada, 2008.
(18) Weininger, D. SMILES, a chemical Language and Information System.1.Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31–36.
(19) *SMIRKS*; Daylight Chemical Information Systems: Aliso Viejo, CA, 2007.
(20) *Reactor, version 5.1.2*; Chemaxon Kft: Budapest. Hungary, 2008.
(21) Proschak, E.; Wegner, J. K.; Schüller, A.; Schneider, G.; Fechner, U. Molecular Query Langage (MQL) - A context-free grammar for substructure matching. *J. Chem. Inf. Model.* **2007**, *47*, 295–301.
(22) Steinbeck, C.; Han, Y.; Kuhn, S.; Horlacher, O.; Luttmann, E.; Willighagen, E. The Chemistry Development Kit (CDK): an open-source Java library for Chemo- and Bioinformatics. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 493–500.
(23) Dugundji, J.; Ugi, I. An algebraic model of constitutional chemistry as a basis for chemical computer programs. *Top. Curr. Chem.* **1973**, *39*, 19–64.
(24) Hopcroft, J.; Motwani, R.; Ullmann, J. D. *Introduction to Automata Theory, Languages and Computation*; Addison Wesley Publishing Company: Boston, MA, 2001; p 169.
(25) *Java, version 6.0, update 5*; Sun Microsystems Inc.: Santa Clara, CA, 2008.
(26) Hartenfeller, M.; Proschak, E.; Schüller, A.; Schneider, G. Concept of combinatorial de novo design of druglike molecules by particle swarm optimization. *Chem. Biol. Drug. Des.* **2008**, *72*, 16–26.
(27) Schüller, A.; Suhartono, M; Fechner, U.; Tanrikulu, Y.; Breitung, S.; Scheffer, U.; Göbel, M. W.; Schneider, G. The concept of template-based de novo design from drug-derived molecular fragments and its application to TAR RNA. *J. Comput.-Aided Mol. Des* **2008**, *22*, 59–68.
(28) Fechner, U.; Schneider, G. Flux (2): comparison of molecular mutation and crossover operators for ligand-based de novo design. *J. Chem. Inf. Model.* **2007**, *47*, 656–67.
(29) Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. RECAP - Retrosynthetic Combinatorial Analysis Procedure: A Powerful New Technique for Identifying Privileged Molecular Fragments with Useful Applications in Combinatorial Chemistry. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 511–522.
(30) Gupta, S.; Aires-de-Sousa, J. Comparing the chemical spaces of metabolites and available chemicals: models of metabolite-likeness. *Mol. Diversity* **2007**, *1*, 23–36.

CI800215T