

From Structure Diagrams to Visual Chemical Patterns

Karen Schomburg, Hans-Christian Ehrlich, Katrin Stierand, and Matthias Rarey*

Research Group for Computational Molecular Design, Center for Bioinformatics, University of Hamburg,
Bundesstrasse 43, D-20146 Hamburg, Germany

Received May 26, 2010

The intuitive way of chemists to communicate molecules is via two-dimensional structure diagrams. The straightforward visual representations are mostly preferred to the often complicated systematic chemical names. For chemical patterns, however, no comparable visualization standards have evolved so far. Chemical patterns denoting descriptions of chemical features are needed whenever a set of molecules is filtered for certain properties. The currently available representations are constrained to linear molecular pattern languages which are hardly human readable and therefore keep chemists without computational background from systematically formulating patterns. Therefore, we introduce a new visualization concept for chemical patterns. The common standard concept of structure diagrams is extended to account for property descriptions and logic combinations of chemical features in patterns. As a first application of the new concept, we developed the SMARTSviewer, a tool that converts chemical patterns encoded in SMARTS strings to a visual representation. The graphic pattern depiction provides an overview of the specified chemical features, variations, and similarities without needing to decode the often cryptic linear expressions. Taking recent chemical publications from various fields, we demonstrate the wide application range of a graphical chemical pattern language.

INTRODUCTION

Countless applications in various chemical fields depend on representations of molecular patterns. These extremely powerful specifications of a set of chemical features range in complexity from very simple substructure descriptions in analyses of molecular similarity up to elaborate logical combinations of functional groups in drug design approaches. The two-dimensional (2D) depiction of molecules as structure diagrams aids chemists to get a quick estimation on the chemical characteristics of compounds despite apparent complicated names. Structure diagrams are often called as being the language of chemistry. However, no comparable standard strategies for visualization of chemical patterns exist so far. Our aim is to initiate a discussion within the chemical society on the development of such a standard. Here we introduce a new approach to a graphic representation of patterns that is based on the recommendations of the IUPAC for chemical structure diagram drawing.¹ In order to demonstrate the relevance of chemical patterns, we applied our visualization concept to examples from recent chemical publications.^{2–4}

The most known and used employment of molecular patterns is database searching, where they are used to filter a set of molecules for compounds related to a query. Starting compounds for organic synthesis with certain functional groups can also be found this way. In drug design, known active ligands for a target are analyzed, and the parts presumed to be responsible for activity are mapped to a molecular pattern for finding new active ligands in a database.⁵ Compound filtering is another well-used applica-

tion of molecular patterns. Unwanted chemical properties, like highly reactive functional groups, are excluded in advance from compound libraries used in high-throughput and virtual screenings^{6–8} or are avoided in *denovo* drug design approaches.⁹ In combinatorial chemistry, patterns are used for characterizing bonds of complete molecules that are allowed to be broken.¹⁰ The groups of Lewell et al.¹¹ and Vieth et al.¹² successfully applied pattern-based rules of breaking bonds for creating fragment libraries. Other applications are the use of patterns as 2D molecular descriptors¹³ and for pharmacophore matching.¹⁴ An exotic application was published by Hou et al.,¹⁵ who used patterns for describing functional groups of molecules in the course of predicting their catabolic transformation in microbes. Not to be forgotten is the use of molecular patterns in the form of Markush structures in patents. These structures mostly consist of a generic core with several variable but defined R-group moieties. Markush notations are used to define a set of structures that are covered by the patent.¹⁶ However, most of the above-mentioned methods rely on much more generic pattern representations and depend on chemoinformatic tools and algorithms.

Therefore, the respective patterns are represented by computationally processable molecular pattern languages. The most prominent molecular pattern language is the SMILES arbitrary target specifications (SMARTS) language.¹⁷ It originates from the simplified molecular input line entry system (SMILES),¹⁸ a linear notation of molecules, and is an extension of its concept. In addition to the means needed to describe a complete molecule, atoms and bonds can be further specified with properties. Additionally, all specifications can be connected logically by AND, OR, and NOT. Other molecular pattern languages, including molec-

* Corresponding author. E-mail: rarey@zbh.uni-hamburg.de. Telephone: +49-40-42838-7350.

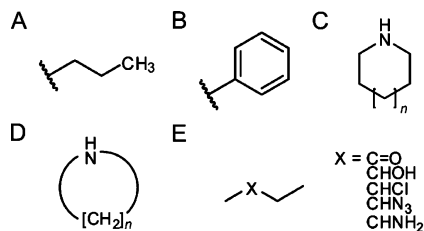


Figure 1. Examples from the IUPAC recommendations on structure diagram drawing for variable structures show graphic display of bonds to unknown moieties (A, B), rings with unknown size (C, D), and a list of variable substituents (E). The figures are adopted from IUPAC recommendations.¹

ular query language (MQL)¹⁹ and Sybyl line notation (SLN)²⁰ are built up comparably, differing mainly in the syntax of the language and only slightly in the semantics. In contrast to the traditionally used concept of varying moieties and functional groups at an otherwise fixed compound, these languages provide the means to construct more elaborate chemical feature descriptions.

However, linear notations of molecules and the pattern languages derived from them are designed for effective computational processing. Molecular patterns more closely resemble regular expressions of programming languages than intuitive descriptions of chemical features. As a consequence, the interpretation and the building of a pattern with one of these languages demand a significant learning effort. Due to the syntax of these languages, patterns may get hardly interpretable by humans, even if they are very familiar with the languages. The SMARTS pattern for a sulfonamide group in ionic or neutral form is:

```
[#16;$([#16X4]([NX3])(=[OX1])(=[OX1])(#6)),
$([#16X4 + 2]([NX3])([OX1 - ])([OX1 - ])(#6))]
```

Although the pattern is not very complicated, the syntax with many brackets leads to a hardly readable expression. Therefore, a more suitable pattern representation is needed. Similar to structure diagrams, a visual depiction of patterns may support scientists in understanding the structural features without much effort. By providing a way to convert this visual depiction to the pattern languages needed for computational methods, the immediate interference of users with these languages can be avoided. The International Union of Pure and Applied Chemistry (IUPAC) recommendations on graphical representations of chemical structure diagrams¹ apply an accepted standard on visualization of compounds in 2D.

However, the recommendations are very limited with respect to the visualization of chemical patterns. Figure 1 presents an assembly of the recommendations concerning variable structures.

Some chemical structure editors offer the possibility of drawing a substructure as a query for database searching.^{21–23} The recently developed PubChem chemical structure sketcher can convert such an input to the SMARTS format and also generate a structure out of a SMARTS input string.²³ However, the graphical depiction of query properties is very elementary in a way that the properties are simply written at the concerned atom. For query bonds new symbols with nonobvious denotations are introduced. Another editor that allows the drawing of molecular substructures is the molecular editor Symyx/Draw.²² Still, the capabilities for generat-

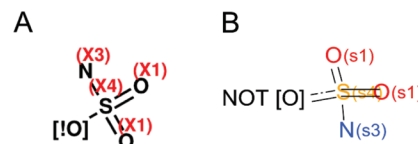


Figure 2. Graphical depiction of the molecular pattern of a sulfonamide (SMARTS: [SX4]([NX3])(=[OX1])(=[OX1])[!O]) generated by the PubChem sketcher²³ (A) and Symyx/Draw²² (B). The property “total number of connections” (SMARTS: X<n>) is printed in both depictions next to the atom element letter. The logic NOT connected to the oxygen is once depicted by an exclamation mark, which is the corresponding SMARTS symbol and once by the text “NOT”.










ing queries do not cover much of the power of molecular pattern languages. The use of logic operators is not supported for most of the features, and the graphical depiction of query properties is comparable to that of the PubChem sketcher. Since both editors are not capable of depicting recursive SMARTS expressions, only one part of the SMARTS pattern representing sulfonamides (see above) was used as input to both editors (Figure 2).







The PubChem sketcher generates the depiction (Figure 2A) automatically, for the depiction by Symyx/Draw (Figure 2B) the pattern has to be drawn manually. Being very alike, both show the limitations of the depiction concept by containing textual overlap with the structure. The depictions are not of great help to scientists not familiar with a pattern language since they still have to figure out details like the meaning of X<n> which denotes “the number of total connections of the atom” (depicted by PubChem by the letter “X” next to the atom and by Symyx/Draw by the letter “s”).

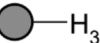



Due to the lack of existing methods for full depiction of chemical patterns, we developed a visualization concept that is capable of depicting patterns described by chemical features as defined in the SMARTS language. The SMARTS language was chosen as a basis, since it is the most prominent among the linear molecular pattern languages. We considered the following points crucial in developing the visualization concept: First, the effort to learn the new concept has to be significantly less than learning a molecular pattern language itself. Therefore, we based our concept on the well-known representation of molecules as structure diagrams. Second, the depiction should be clear to users without knowledge of molecular pattern languages. Addressing these points, special effort was made to visualize the complete power of the language but to stay consistent with the concept of structure diagrams. Finally, the visualization concept should be generically applicable to different pattern representations. Although the developed visualization concept is based on the SMARTS language, it can be applied—with minor adaptations—to other pattern languages like MQL or SLN, since the semantics and the power of these languages differ only slightly.

In a first application, the concept was realized in a tool that automatically generates a visualization of a SMARTS string named SMARTSviewer available on the Internet (see <http://smartsview.zbh.uni-hamburg.de>). In the following, first the visualization concept of graphic elements decoding the SMARTS primitives and logic is presented proceeded by a description of the SMARTSviewer implementation. Finally, some exemplary visualizations are discussed.

Table 1. Visualization of Chemical Features That the SMARTS Language Offers to Specify an Atom

Feature	Carbon atom	Aliphatic carbon atom	Aromatic carbon atom	Charge	Mass
SMARTS	[#6]	C	c	[N+1]	[13C]
Color coding				 ⁺¹	 13
Element letters	C			 ⁺¹	 13C

Feature	Number of total connections	Number of connections to non-hydrogen atoms	Number of connections in a ring
SMARTS	[CX4]	[CD3]	[Cx2]
Color coding	 4— 	 3—  H	 2— 

Feature	Number of connected hydrogens	Size of ring	Number of rings	Valency
SMARTS	[CH3]	[Cr5]	[CR1]	[Cv4]
Color coding	 —H ₃	 5	 n	

VISUALIZATION CONCEPT

In the SMARTS language, the concept of SMILES specifying the molecule graph with symbols for atoms and bonds is extended by logical operators and several additional symbols specifying properties. Therefore, the visualization concept of structure diagrams has to be extended to cover these additional features in order to achieve a complete visualization. In the concept presented here, the layout of the coordinates of atoms is fully adopted from complete molecules, meaning that, for example, the atoms of a benzene ring are laid out on a hexagon. New graphic symbols are introduced for the depiction of query features. However, all these symbols are based on existing IUPAC recommendations, which are only slightly adapted.

Atoms and Atomic Properties. Additionally to the element, query properties for specifying atoms in SMARTS include features like aromaticity, number of connections, mass, valency, number of connected hydrogens, and several more. Table 1 shows how each of these properties is depicted in the visualization concept. Atoms are drawn as circles that are colored by element. Instead of the color coding, the element can also be depicted by the element letters printed into the circle. In both cases, the stroke of the circle encodes the aromaticity. If none is specified, then no stroke is applied, and if the atom is specified as aliphatic, then a black stroke is applied, and in the case of an aromatic specification, this stroke is dashed. Note that, although aromaticity is a property of a group, most pattern languages consider it as an atom/bond property in order to allow the partial specification of aromatic systems. The value of a given charge is printed at the upper right corner of the atom circle and the mass into the middle of the circle. Several features are available in the SMARTS language to describe the connectivity of an atom. In general, for visualizing this property, a bond ending in a waved line (similar to the IUPAC recommendation for a moiety, see Figure 1A and B) is used. The value of the connectivity given in the SMARTS is printed onto this aborted bond. For the number of connections to nonhydrogen atoms a red colored letter 'H', and for the connections that are in a ring, a hexagon as a symbol for a ring is added

Table 2. Visualization of the Logic Operators OR and NOT in Chemical Patterns^a

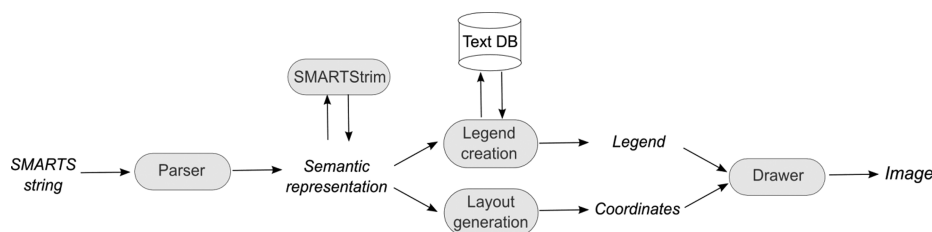
Logic	Feature	Elements	Number of connections to non-hydrogen atoms	Bonds
OR	SMARTS	[N,C,O]	[C;D2,D3]	C-,=C
	Depiction			
NOT	SMARTS	[!C]	[C!D1]	C!:C
	Depiction			

^a Red codes the logic NOT. For depiction of the logic OR, the given values are enumerated or the color blue is used.

behind the waved line. A short bond leading to the letter 'H' subscripted with the value given in the SMARTS indicates the number of connected hydrogen atoms. The features of membership in a ring of a given size and membership in a given number of rings are depicted very similarly with a symbol based on the IUPAC recommendation for variable rings (see Figure 1C and D). While the SMARTS language allows several values for the number of rings an atom participates in (SMARTS primitive 'R<n>'), the visualization only covers two cases: "being in a ring" (SMARTS primitive 'R1') and "not being in a ring" (SMARTS primitive 'R0'). Higher values of <n> are not recommended since this property is defined over the smallest set of smallest rings (SSSR) base, which is not unique²⁴ and therefore not considered in our visualization concept. The valency is depicted by small dots drawn inside the atom circle.

Bond Specifications. Bonds are illustrated by one, two and three lines for single, double, and triple bonds, respectively, and by a pair of solid and dashed lines for aromatic bonds, concurring to structure diagram drawing. Wedged bonds are used for depicting tetrahedral stereochemistry (being intrinsically a feature of an atom in the SMARTS language). Additionally, the SMARTS language allows bonds to be specified as a ring or any bond. A bond specified as "any" is depicted by a single line in a light gray color, indicating the unspecified nature of it. For depicting a ring bond, again the symbol of a hexagon for a ring is used and printed onto the bond. Cis and trans configurations around double bonds, being in SMARTS also a feature of bonds, lead to a respective layout of the coordinates of the regarding atoms. The Supporting Information contains a graphic display of all bond types.

Logic Operators. The greatest challenge is the visualization of the SMARTS language's powerful concept of logic. Features describing atoms or bonds can be combined by a logic AND or a logic OR or can be excluded by the logic NOT. In the presented concept, logic is encoded by color. Red indicates the logic NOT and blue the logic OR. Table 2 shows logic combinations of elements, values for the connectivity to non-hydrogen atoms, and bonds. Logical combinations of elements for one atom are depicted by dividing the circle among the elements. Bonds that are connected by OR are placed into a box and colored blue. The logic NOT is depicted by a red circle for elements, a red colored value for connectivities, and a red colored bond. A limitation of the concept is found in the depiction of many elements combined by the logic OR. A division of the circle

Scheme 1. Flowchart of the SMARTSviewer Implementation^a

^a An input SMARTS string is parsed into a semantic representation. Before being further processed, the semantic is checked with the SMARTStrim procedure. Then the layout generation provides atom coordinates, and the legend is put together out of a textual database. Both the legend and the coordinates serve as input to a drawer which generates the image.

among more than four elements becomes indistinct. Consequently, if this very rare case occurs, the elements are written as a list in place of the circle (for an example see Supporting Information, Figure 5B). Another special case occurs, if several elements are logically connected to different feature descriptions at one atom. Then an unambiguous assignment of the features to the element is needed, and the features cannot be depicted by their graphic elements. A so-called property bubble is drawn containing the features in form of SMARTS primitives with a connecting line to the respective element fraction (for an example see Supporting Information, Figure 5A).

Recursive Specifications. Recursive expressions, which in SMARTS define the chemical environment of an atom, can be connected by logic operators as well. These specifications are treated as independent molecular graphs and drawn into boxes next to the main molecular graph of the pattern. The color of the stroke of the boxes maps the logic. Consistently, red codes for the logic NOT and blue codes for the logic OR.

Dynamic Legend. The amplification of the visualization concept with a legend serves the purpose to support users in getting accustomed to the new concept. This obstacle is addressed by the first part of the legend, which provides a textual description of each distinct atom and bond. More experienced computational chemists might be familiar with the SMARTS language and therefore only have to get acquainted with the meaning of the graphic symbols. This need is addressed by the second part of the legend, which maps graphic symbols to the respective SMARTS primitives. Both parts of the legend are generated for each SMARTS string individually and consequently contain only information relevant for the depicted pattern.

COMPUTATIONAL METHODS

The presented visualization concept is realized in a first application as a tool that automatically depicts a SMARTS string (see <http://smartsview.zbh.uni-hamburg.de>). The implementation consists of three main steps which are outlined in Scheme 1. A SMARTS string is processed by a parser which retrieves the semantic information. An interpretation of the semantic is followed by the layout generation and the legend creation. The legend and the coordinates are then further processed by the drawer to generate an image. These steps are described in the following sections.

SMARTS Parsing. In the parsing routine, the input SMARTS string is converted into a tree-like data structure that represents the semantic of the pattern. For this purpose, the SMARTS language is modeled as a context free

grammar.²⁵ The grammar allows to check the correct syntax of the SMARTS string and to extract the contained semantic information. The resulting tree-like representation also contains the precedence of logic operators of the SMARTS language. Therefore, the relevant information, for example, the exact specification of an atom of the pattern, can be extracted easily.

SMARTS Trim. In addition to the syntax test performed by the parser, the semantic is inspected by a procedure called SMARTStrim. This routine is divided into three parts with differing emphases. The SMARTStrim error-check part removes semantic errors. An example is an impossible connection of properties by a logic AND, like an atom that is specified as being a carbon AND a nitrogen. The SMARTStrim simplification part removes redundant information from the SMARTS string, like combinations of property specifications and wildcards. An example is a bond that is specified as “any and ring bonds”, which is semantically the same as “ring bond”. The third part called SMARTStrim interpretation identifies correlations in the pattern structure and the specifications. An example is an atom that is specified with the property of “being in a ring of size six” but is structurally already in a six-membered ring. All cases which apply to the three parts are listed in the Supporting Information.

Legend Generation. For every atom and bond property which is part of the SMARTS language, a descriptive word or phrase is stored. In the legend creation procedure, the distinct atoms and bonds are identified to avoid redundant legend content. For creating a textual description of an atom or a bond, the respective descriptive words are pieced together with conjunctions concurring with the property and logic specifications of the atom or bond.

Layout Generation. The general layout of a pattern as a chemical graph is consistent with the layout of chemical structure diagrams. Pattern descriptions consist of atoms and bonds as well as any complete molecule, only the specifications of the atoms and bonds differ. Therefore, the assignment of coordinates to the atoms of a pattern is the same problem as that of complete molecules, called a structure diagram generation problem.²⁶ For every recursive expression in the pattern, a separate chemical graph is created. The relative coordinates are assigned to the atoms with the method of structure diagram generation published by Fricker et al²⁷ and obey the IUPAC rules of 2D diagram drawing. Afterward, the molecular graphs are placed absolutely with the recursive diagrams next to the main diagram. In addition to structure diagrams, graphic elements that depict specifications are placed at atoms and bonds. In order to support recognition

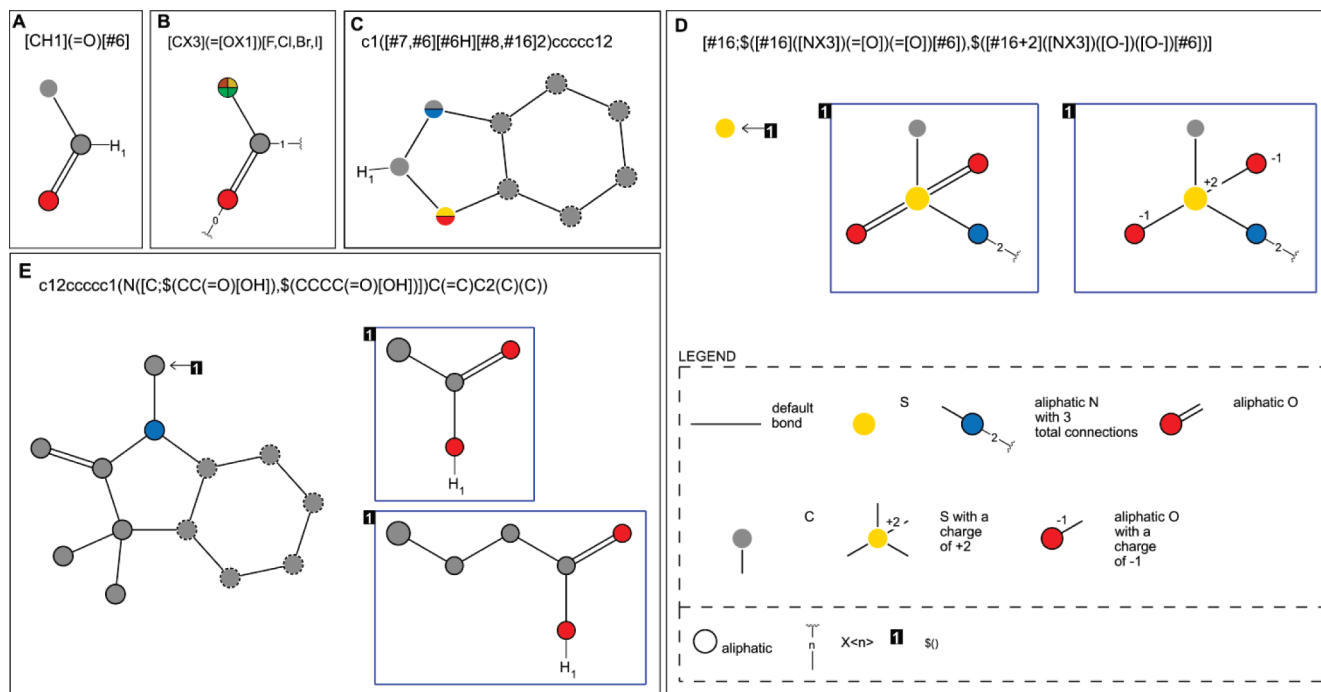


Figure 3. SMARTS expressions visualized by the SMARTSviewer. As the complexity ranges from a pattern describing an aldehyde group (A), via a reactive acyl halide group (B), a pattern matching an aromatic ring system with heteroatoms (C), a sulfonamide group (D), to an indoline group (E) with a chain of variable length to a carboxyl group also the visualization gets more complicated. The visualization of the pattern for a sulfonamide is shown together with the dynamic legend.

of the chemical structure of the pattern, the additional graphic elements are placed without altering the chemical graph layout.

SMARTSviewer Tool. The graphic elements that depict the properties of the pattern are drawn with the open source 2D graphics library cairo.²⁸ The SMARTSviewer tool can be either used interactively with a Qt²⁹ based graphical user interface or accessed via a web interface at <http://smartview.zbh.uni-hamburg.de>. As input, it takes the pure SMARTS string and generates either pixel- or vector-based images. The user has several possibilities to influence the generated image, among others choosing between color or element letter coding of elements or showing/hiding the legend.

Test Data. For testing the automated generation of the visualization, SMARTS strings that are employed in real applications are used. The SMARTS strings are collected from publications^{5,7–9,13,30,31} and the daylight webpage;³² however, only SMARTS strings that are conform to the Daylight SMARTS specifications are included in the test set. The resulting test set of 762 SMARTS strings ranges from simple patterns used in real applications to highly complicated patterns with many recursions found in the examples from the daylight webpage. The string length varies from 2 to 1008 characters, 247 strings of the set contain recursions. More details on the particular string sets can be found in the Supporting Information.

For showing the relevance of a visualization concept of chemical pattern beyond the application of visualizing SMARTS, some patterns occurring in recent organic chemistry publications^{2–4} are visualized according to the visualization concept.

RESULTS AND DISCUSSION

The visualization concept was successfully applied to all SMARTS strings of the test set. Figure 3 shows the visualization of five exemplary chemical patterns automatically generated with the SMARTSviewer. The first three examples are rather simple patterns, representing an aldehyde functional group (A), an acyl halide group (B), which may be used as a filter for reactive components, and an aromatic ring system with heteroatoms at specified positions (C), which was taken from Vechorkin et al.,² where the variable parts of the structure are listed. These three examples demonstrate the depiction of elements and aromaticity, while the second example (B) highlights the handling of connectivities. In the SMARTS string, the carbon atom is specified as having three connections (SMARTS primitive 'X3'). This feature is depicted by a truncated line with the value of the further connections placed onto it, meaning that the already covered connections are subtracted and only the unsaturated connections remain. In this case, two of the three connections are already covered by the structure. The fourth pattern (D) matches sulfonamide groups, which are the common functional group of antibacterial "sulfa drugs", either in the ionic or neutral form. Here the visualization of recursive SMARTS expressions in boxes next to the main graph is demonstrated as well as the two parts of the legend. Since the two recursive groups are connected by a logic OR, the borders of the boxes are colored blue. For a better identification of the recursively specified atom in the boxes, this atom is drawn with an enlarged radius. The upper part of the legend describes each distinct atom and bond with a short text, while the lower part maps the SMARTS symbols used in the string to the respective graphic elements. The fifth visualization example (E) shows a pattern for an indoline with a variable chain

```
c1c[c;$$(cc1ccccc1),$(cC(=O)OC),$(c-[Br,Cl,C]),$(cC(F)(F)F),$(cO[C;$(CH3)],$(C[CH3])),$(c[N;$(N([CH3])
[CH3]),$(N(=O)=O)))]cc([#7]=[#6](C[Br,Cl])[#16]2)c21
```

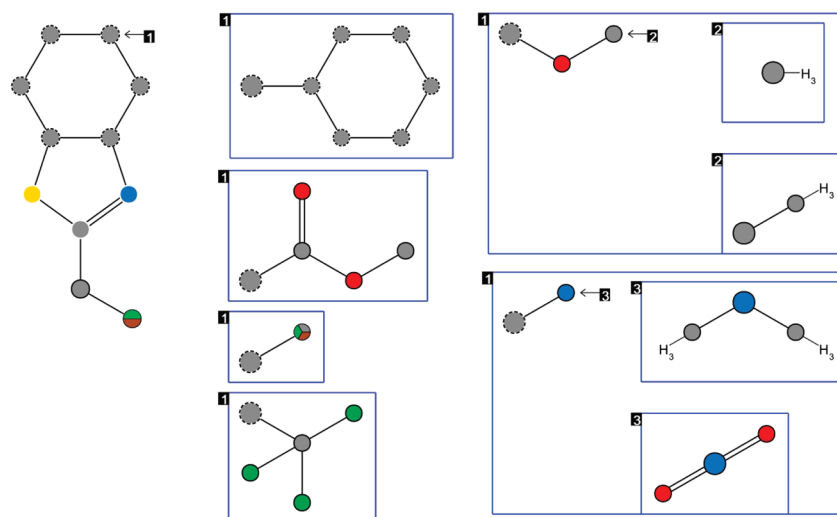


Figure 4. Visual representation of a chemical pattern that represents the starting compound in a synthesis of benzothiazole derivatives active in bacterial cell division inhibition.⁴ The compound is substituted with a variable moiety. The recursive expressions show the differences and similarities of the allowed substituents and provide an overview of all possibilities.

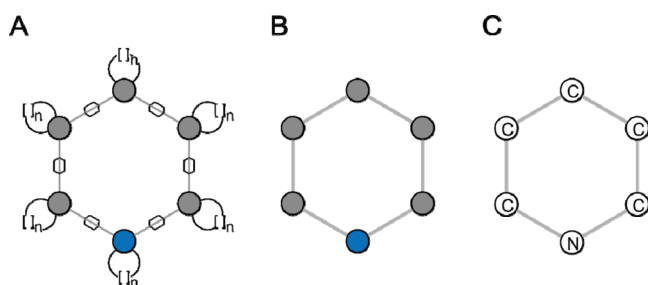


Figure 5. Visualization of the SMARTS pattern [CR]@1@[CR]@[CR]@[CR]@[CR]@[NR]@1 without the SMARTStrim procedure (A), with using the SMARTStrim procedure resulting in a much simpler but with regards to the chemical meaning identical visualization (B) and in the viewing mode of presenting elements not by color but by element letters (C).

length to the carboxyl group. This example was taken from Meguellati et al.,³ where two structures are drawn, one for each chain length. The depiction in form of a pattern highlights the common and variable parts of the structure.

A more complex pattern is shown in Figure 4. The pattern representing benzothiazole derivatives being used in the synthesis of inhibitors of a bacterial cell division protein was adopted from Haydon et al.,⁴ where the various substituents are listed in the form of abbreviations. An advantage of the representation in the form of a visual pattern is the possibility to take in all substituents at a glance instead of looking them up and interpreting the abbreviations. Additionally, the differences and similarities of the variations are emphasized by the depiction.

The effect of the SMARTStrim procedure is highlighted by the example in Figure 5. The visualization of the SMARTS string [CR]@1@[CR]@[CR]@[CR]@[CR]@[NR]@1 without applying the SMARTStrim procedure leads to an unnecessarily complicated figure (A). However, the SMARTStrim procedure recognizes that the pattern can be simplified without changing its meaning. All bonds in this pattern are specified as ring bonds, although the structure of the pattern already satisfies this condition. The same is true for the atoms

of the ring, which are specified as “being in a ring”. Therefore, the SMARTStrim procedure simplifies the visualization significantly (B). While in all previous figures, elements are depicted by color, Figure 5 C employs the element letter viewing mode. Both depictions convey certain advantages. The color coding supports a very quick recognition of the structure and highlights heteroatoms. The element letter representation is the classic way of depicting elements in 2D structure diagrams and may thus be easier to get acquainted with, since not every scientist may be familiar with the color code. In general, the choice will depend on every scientist’s personal preferences, and therefore, both viewing options are realized in the SMARTSviewer. Still, both modes rely on the use of color for depicting logic combinations. In principal, it is possible to create a pure black and white depiction by introducing graphic elements for logic expressions. However, we preferred the color depiction in order to avoid introducing more graphic elements in addition to the ones depicting features.

While the visualization concept could be evaluated concerning the ability of visualizing the complex and powerful SMARTS specifications, the real evaluation will be through the chemical community. Since a visualization concept has to evolve through the needs of science it represents, the visualization concept shown here should be seen as a starting point for the development of a unified chemical pattern language. Current limitations of the concept that may be addressed in future discussions concern extensions of the drawing concept for pattern specifications that are not part of the SMARTS language. Chemical pattern languages, such as SMARTS, are extremely powerful. It is possible to describe highly complex patterns, for example, due to the extensive use of logic expressions. The graphic display of such a pattern is obviously not simple, and, in some cases, probably impossible. Therefore, the visualization concept concentrates on depicting patterns that a scientist may come across in real applications. In our opinion, one of the most important characteristics of a visual pattern language is that

simple patterns result in simple depiction, while more complicated patterns may result in more complicated graphical representations. The examples generated with SMARTS-viewer show that the visualization concept presented here achieves this aim in most cases.

CONCLUSION

In summary, we have introduced a new visual representation of chemical patterns. The natural way of communicating structures in the chemical society is via structure diagrams. Therefore, a visual pattern representation must not diverge extensively from the concept of structure diagram drawing. For the full depiction of chemical features that may be used to describe variable structures, this concept had to be extended with several new graphic symbols. For providing an easy way to get acquainted with these, the visualization concept comprises a legend that depicts the meaning of the pattern by a textual description. In a first application, the visualization concept was realized as an automated depiction of the often hardly interpretable SMARTS patterns. This new visual pattern representation has to be seen as a starting point in improving the communication of molecular patterns among scientists. It responds to two problems of handling chemical patterns: First, it offers a general representation of chemical patterns, and second, it improves the usability of computational molecular pattern languages by depicting the linear form graphically.

ADDITIONAL INFORMATION

The SMARTSviewer tool can be accessed freely via the Internet at <http://smartsview.zbh.uni-hamburg.de>.

Supporting Information Available: The complete visualization concept covering the full power of the SMARTS language and details of the SMARTStrim concept. This material is available free of charge via the Internet at <http://pubs.acs.org>.

REFERENCES AND NOTES

- Brecher, J. Graphical Representation Standards for Chemical Structure Diagrams (IUPAC Recommendations 2008). *Pure Appl. Chem.* **2008**, 80 (2), 277–410.
- Vechorkin, O.; Proust, V.; Xile, H. The Nickel/Copper-Catalyzed Direct Alkylation of Heterocyclic C-H Bonds. *Angew. Chem.* **2010**, 122 (17), 3125–3128. *Angew. Chem., Int. Ed. Engl.*, **2010**, 49(17), 3061–3064.
- Meguellati, K.; Koripelly, G.; Ladame, S. DNA-Templated Synthesis of Trimethine Cyanine Dyes: A Versatile Fluorogenic Reaction for Sensing G-Quadruplex Formation. *Angew. Chem.* **2010**, 122 (15), 2798–2802. *Angew. Chem., Int. Ed. Engl.* **2010**, 49(15), 2738–2742.
- Haydon, D. J.; Bennet, J. M.; Brown, D.; Collins, I.; Galbraith, G.; Lancett, P.; Macdonald, R.; Stokes, N. R.; Chauhan, P. K.; Sutariya, J. K.; Nayal, N.; Srivastava, A.; Beanland, J.; Hall, R.; Henstock, V.; Noulia, C.; Rockley, C.; Czaplowski, L. Creating an Antibacterial with in Vivo Efficacy: Synthesis and Characterization of Potent Inhibitors of the Bacterial Cell Division Protein FtsZ with Improved Pharmaceutical Properties. *J. Med. Chem.* **2010**, 53 (10), 3927–3936.
- Enoch, S. J.; Madden, J. C.; Cronin, M. T. D. Identification of mechanisms of toxic action for skin sensitisation using a SMARTS pattern based approach. *SAR QSAR Environ. Res.* **2008**, 19, 555–578.
- Baurin, N.; Baker, R.; Richardson, C.; Chen, I.; Foloppe, N.; Potter, A.; Jordan, A.; Roughley, S.; Parrat, M.; Greaney, P.; Morley, D.; Hubbard, R. E. Druglike Annotation and Duplicate Analysis of a 23-Supplier Chemical Database Totalling 2.7 Million Compounds. *J. Chem. Inf. Comput. Sci.* **2004**, 44, 643–651.
- Walters, W. P.; Murcko, A. M. Prediction of 'drug-likeness'. *Adv. Drug Delivery Rev.* **2002**, 54, 255–271.
- Hann, M.; Hudson, B.; Lewell, X.; Lively, R.; Miller, L.; Ramsden, N. Strategic Pooling of Compounds for High-Throughput-Screening. *J. Chem. Inf. Comput. Sci.* **1999**, 39 (5), 897–902.
- Maass, P.; Schulz-Gasch, T.; Stahl, M.; Rarey, M. Recore: A Fast and Versatile Method for Scaffold Hopping Based on Small Molecule Crystal Structure Conformations. *J. Chem. Inf. Model.* **2007**, 47, 390–399.
- Schneider, G.; Fechner, U. Computer-Based De Novo Design of Drug-Like Molecules. *Nat. Rev. Drug Discovery* **2005**, 4 (8), 649–63.
- Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. RECAPs 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 (3), 511–522.
- Vieth, M.; Siegel, M. G.; Higgs, R. G.; Watson, I. A.; Robertson, D. H.; Savin, K. A.; Durst, G. L.; Hipskind, P. A. Characteristic Physical Properties and Structural Fragments of Marketed Oral Drugs. *J. Med. Chem.* **2004**, 47 (1), 224–232.
- Olah, M.; Bologa, C.; Oprea, T. I. An automated PLS search for biologically relevant QSAR descriptors. *J. Comput.-Aided Mol. Des.* **2004**, 18, 437–449.
- Van Drie, J. H.; Weininger, D.; Martin, Y. C. ALADDIN: An integrated tool for computer-assisted molecular design and pharmacophore recognition from geometric, steric, and substructure searching of three-dimensional molecular structures. *J. Comput.-Aided Mol. Des.* **1989**, 3, 225–251.
- Hou, B. K.; Wackett, L. P.; Ellis, L. B. M. Microbial Pathway Prediction: A Functional Group Approach. *J. Chem. Inf. Comput. Sci.* **2003**, 43 (3), 1051–1057.
- Lynch, M. F.; Barnard, J. M.; Welford, S. M. Computer Storage and Retrieval of Generic Chemical Structures in Patents. 1. Introduction and General Strategy. *J. Chem. Inf. Comput. Sci.* **1981**, 21, 148–150.
- Daylight Theory Manual, version 4.9; Daylight Chemical Information Systems, Inc.: Aliso Viejo, CA, 2008; <http://www.daylight.com/dayhtml/doc/theory/index.html>. Accessed July 20, 2010.
- 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.
- Proschak, E.; Wegner, J. K.; Schüller, A.; Schneider, G.; Fechner, U. Molecular Query Language (MQL) - A Context-Free Grammar for Substructure Matching. *J. Chem. Inf. Model.* **2007**, 47, 295–301.
- Homer, R. W.; Swanson, J.; Jilek, R. J.; Hurst, T.; Clark, R. D. SYBYL Line Notation (SLN): A Single Notation To Represent Chemical Structures, Queries, Reactions, and Virtual Libraries. *J. Chem. Inf. Model.* **2008**, 48, 2294–2307.
- Bruno, I. J.; Cole, J. C.; Edgington, P. R.; Kessler, M.; Macrae, C. F.; McCabe, P.; Pearson, J.; Taylor, R. New Software for Searching the Cambridge Structural Database and Visualizing Crystal Structures. *Acta Crystallogr., Sect. B: Struct. Sci.* **2002**, 58, 389–97.
- Symyx/Draw, version 3.2; Symyx Technologies, Inc.: Sunnyvale, CA, 2009.
- Ihlenfeldt, W. D.; Bolton, E. E.; Bryant, S. H. The PubChem Chemical Structure Sketcher. *J. Cheminf.* **2009**, 1, 20.
- Berger, F.; Flamm, C.; Gleiss, P. M.; Leydold, J.; Stadler, P. F. Counterexamples in Chemical Ring Perception. *J. Chem. Inf. Comput. Sci.* **2004**, 44, 323–331.
- Alfred, V. A.; Ullmann, J. D. In *The Theory of Parsing, Translation, and Compiling*; Prentice-Hall, Inc.: London, 1972; Vol. 1, Chapter 2, pp138–167.
- Helson, H. E. Structure Diagram Generation. In *Reviews in Computational Chemistry*, Wiley-VCH: New York, 1999; Vol. 13, pp 313–398.
- Fricker, P.; Gastreich, M.; Rarey, M. Automated Drawing of Structural Molecular Formulas under Constraints. *J. Chem. Inf. Comput. Sci.* **2004**, 44, 1065–1078.
- Cairo Graphics Library, version 1.8.8; Nokia Corporation: Oslo, Norway; Redwood City, CA, <http://www.cairographics.org>. Accessed July 20, 2010.
- Qt Application Development Framework, version 4.6.0; Nokia Qt Development Frameworks: .
- Abolmaali, S. F. B.; Wegner, J. K.; Zell, A. The compressed feature matrix - a fast method for feature based substructure search. *J. Mol. Model.* **2003**, 9, 235–241.
- Agrafiotis, D. K.; Gibbs, A. C.; Zhu, F.; Izrailev, S.; Martin, E. Conformational Sampling of Bioactive Molecules: A Comparative Study. *J. Chem. Inf. Model.* **2007**, 47, 1067–1086.
- Daylight SMARTS examples; Daylight Chemical Information Systems, Inc.: Laguna Niguel, CA; http://www.daylight.com/dayhtml_tutorials/languages/smarts/smarts_examples.html. Accessed May 25, 2010.