The t values are 2.9, −6.8, −18.9, and 54.5, respectively for the coefficients; $r^2 = 0.9872$, $s = 4.89$ °C, $F = 1749$, and $n = 72$.

Calculated values and residuals for the latter correlation are presented in Table III. The two outliers with absolute residuals larger than 12 are the same as the most serious ones in correlation 1 above.

For 44 Sulfides (with mean BP = 150.7):

$$BP = 20.01(\pm 7.49) + 43.93(\pm 0.83)^1\chi - 6.34(\pm 1.20)J_{het} \quad (5)$$

The t values are 2.7, 52.7, and −5.3, respectively; $r^2 = 0.989$; $s = 4.28$ °C, and $F = 1807$.

$$BP = 50.59(\pm 23.95) - 32.91(\pm 13.45)ES_s - 9.86(\pm 0.68)N_{Me} + 24.10(\pm 0.54)K\alpha 1 \quad (6)$$

The t values are 2.1, −2.4, −14.5, and 44.3, respectively; $r^2 = 0.989$; $s = 4.3$ °C; and $n = 44$.

Although in this case both types of correlations yield practically the same statistical results, we reproduce in Table IV only the second batch of data. In both correlations there is just one outlier with a boiling point whose experimental accuracy is subject to doubt (ethyl heptyl sulfide).

One may conclude this section by stating that for ethers the best correlation affords a fit error of 4.2 °C and for sulfides a fit error of 2.8 °C.

In conclusion, we have found that four parameters (the molecular connectivity $^1\chi$, the topological index J modified for the presence of heteroatoms, the electrotopological state S of the heteroatoms, and the number N_s of sulfur atoms) give a good correlation with BPs for unknown compounds belonging to these classes of compounds and this range of carbon atoms and heteroatoms.

ACKNOWLEDGMENT

A.T.B. thanks Sterling Drug Inc. for support. The assistance of Mr. B. Brown for computer programming was much appreciated.

REFERENCES AND NOTES

(1) Balaban, A. T.; Joshi, N.; Kier, L. B.; Hall, L. H. *J. Chem. Inf. Comput. Sci.* **1992**, *32* (preceding paper in this issue).
(2) Stanton, D. T.; Jurs, P. C.; Hicks, M. G. *J. Am. Chem. Soc.* **1991**, *31*, 301.
(3) Bordwell, F. G.; Anderson, H. M.; Pitt, B. M. *J. Am. Chem. Soc.* **1954**, *76*, 1082.
(4) Jurecek, M.; Vecera, M. *Chem. Listy* **1954**, *48*, 542.
(5) SAS Institute Inc. *SAS User's Guide: Statistics, Version 5*; SAS: Cary, NC; 1985.
(6) The programs MOLCONN and MOLCONN2 can be obtained from Prof. L. H. Hall, Hall Associates Consulting, 2 Davis Street, Quincy, MA 02170.
(7) Randić, M. *J. Am. Chem. Soc.* **1975**, *97*, 6609.
(8) Kier, L. B.; Hall, L. H. Molecular Connectivity in Chemistry and Drug Research; Academic Press: New York, 1976.
(9) Kier, L. B.; Hall, L. H. "*Molecular Connectivity in Structure–Activity Analysis*" Research Studies Press and Wiley: New York, 1986.
(10) Balaban, A. T. *Chem. Phys. Lett.* **1982**, *80*, 399.
(11) Balaban, A. T. *Pure Appl. Chem.* **1983**, *55*, 199.
(12) Balaban, A. T. *Math. Chem.* **1986**, *21*, 115.
(13) Kier, L. B. *Quant. Struct. Act. Relat.* **1985**, *4*, 109.
(14) Kier, L. B. *Med. Res. Rev.* **1987**, *7*, 417.
(15) Kier, L. B. *Quant. Struct.–Act. Relat.* **1986**, *5*, 7.
(16) Kier, L. B.; Hall, L. H. *Pharm. Res.* **1990**, *7*, 801.
(17) Hall, L. H.; Mohney, B.; Kier, L. B. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 76.
(18) White, P. T.; Barnard-Smith, D. G.; Fidler, F. A. *Ind. Eng. Chem.* **1952**, *44*, 1430.
(19) *CRC Handbook of Physics and Chemistry, 68th ed.*; CRC: Boca Raton, FL.
(20) *Dictionary of Organic Compounds*, 5th ed.; Chapman and Hall: New York, 1982.

Downloaded via UNIVERSITA DI ROMA LA SAPIENZA on January 31, 2021 at 19:29:10 (UTC).
See https://pubs.acs.org/sharingguidelines for options on how to legitimately share published articles.

Description of Several Chemical Structure File Formats Used by Computer Programs Developed at Molecular Design Limited

ARTHUR DALBY, JAMES G. NOURSE,* W. DOUGLAS HOUNSHELL, ANN K. I. GUSHURST, DAVID L. GRIER, BURTON A. LELAND, and JOHN LAUFER

Molecular Design Limited, 2132 Farallon Drive, San Leandro, California 94577

Received January 23, 1992

A series of file formats used for storing and transferring chemical structure information that have evolved over several years at Molecular Design Limited are described. These files are built using one or more connection table (Ctab) blocks. The Ctab block format is described in detail. The file formats described are the MOLfile for a single (multifragment) molecule, the RGfile for a generic query, the SDfile for multiple structures and data, the RXNfile for a single reaction, and the RDfile for multiple reactions and data. The relationships of these files are given as well as examples.

1. INTRODUCTION

This paper describes the chemical table file (CTfile) formats currently used in a wide variety of chemical structure-manipulating computer programs. These file formats were developed by a large number of people at Molecular Design Limited (MDL) over the past 13 years.[1] While the formats were developed for use with the various MDL programs, their use has gone well beyond this role.

The evolution of the CTfile formats did not proceed by a well-defined plan. Changes were frequently made over the years to accommodate new program features or application needs. While the changes often added new data fields or appendices, they were done such that older files would remain valid and older programs could read the portion of new files they could interpret. Because of this policy, as well as the widespread use of programs which read and write CTfiles, there is probably more valid chemical structure information in existence in these formats than in any other format, current or proposed.

While the evolution of these CTfile formats was closely tied to the development of the various MDL computer programs, the purpose of this paper is to describe just the file formats and not the various computer programs and their features. In general, references to the various programs which use these file formats will be made only when necessary to describe the purpose of a particular part of a file format. Literature references will be given in place of detailed descriptions of various program features.

0095-2338/92/1632-0244$03.00/0 © 1992 American Chemical Society

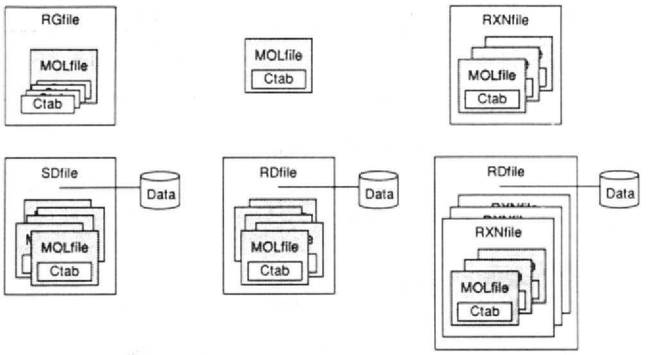

Figure 1. CTfiles: Structure and interrelationships.

Table I. Properties and Identifying Icons Applicable to Various CTfile Types

		CTfile type				
icon	property	MOLfile	RGfile	SDfile	RXNfile	RDfile
[G]	generic	+	+	+	+	+
[Sg]	Sgroup	+	+	+		
[Rg]	Rgroup	+	+	+		
[3D]	3D	+	+	+		
[CP]	CPSS	+		+		+
[Rx]	reaction				+	+
[Q]	query	+	+		+	

The file formats described and their interrelationships are indicated in Figure 1. The key piece describing a chemical structure is the connection table (Ctab) block indicated in all the files illustrated in Figure 1. The Ctab block is not a file by itself but is used as a building block for the various files. The Ctab block is described in Section 2. The simplest complete file is the MOLfile, which contains just one Ctab block. This is described in Section 3. The RGfile (for Rgroup file) is a special query file format for use in generic searching.[2] It is described in Section 4. The SDfile (for Structure-Data file) can contain many MOLfiles combined with data for each. It is intended as a format for moving large numbers of chemical structures and associated data between databases. The SDfile is described in Section 5. The RXNfile (for Reaction file) contains the reactants and products of a single chemical reaction. The RXNfile is described in Section 6. The RDfile (for Reaction-Data file) is a file format that contains multiple RXNfiles and associated data. Since the RDfile is more general in that it can have MOLfiles in place of RXNfiles, it is illustrated twice in Figure 1. The RDfile is described in Section 7.

Because of the very large number of chemical structure properties, it is convenient to group them by class. The classes are symbolized by icons, shown in Table I, which also indicates which of the various CTfile types can currently contain structures with these properties. The Generic class includes basic chemical structure properties that can be registered or used as queries. These can appear in any of the CTfiles. The Sgroup class includes properties that are associated with identified substructures of a chemical structure. These are used for a variety of purposes, often involving more complex chemical substances, and have been described in detail elsewhere.[3] The Rgroup class includes properties that can only be used in Markush-like queries.[2] The 3D class includes properties associated with a three-dimensional chemical structure. These can be query or registerable properties and have been described in detail elsewhere.[4] The CPSS class includes some fairly routine structural properties that are interpreted only by the CPSS (Chemist's Personal Software Series) programs. These are generally duplicated elsewhere in other CTfile locations. The Reaction class includes query and registerable properties that are associated with chemical

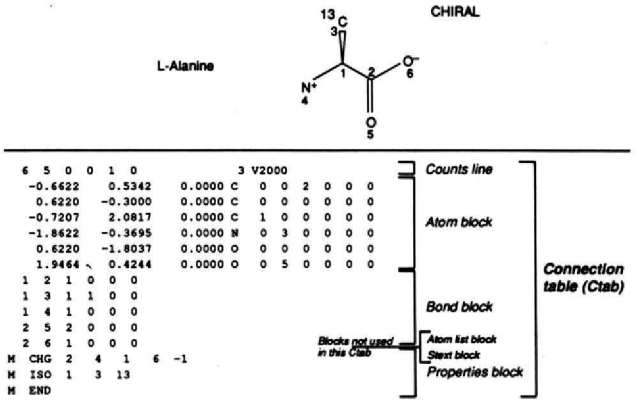

Figure 2. Connection table (Ctab) organization illustrated using alanine.

reactions. The more elaborate ones have been described elsewhere.[5] The query class includes properties that can only be used in search queries.

2. THE CONNECTION TABLE [CTAB]

A connection table (Ctab) contains information describing the structural relationships and properties of a collection of atoms. The atoms may be wholly or partially connected by bonds. Such collections may, for example, describe molecules, molecular fragments, substructures, substituent groups, polymers, alloys, formulations, mixtures, and unconnected atoms. The connection table is fundamental to all of the CTfile formats.

Figures 2, 5, and 6 show the connection tables of a simple molecule (alanine), an Sgroup structure (polymer), and a 3D query, respectively. The various data blocks in each Ctab are identified. The atom numbers on the structures in Figures 2 and 5 correspond to atom numbers in the Ctabs. An atom number is assigned according to the order of the atom in the Atom Block (see Section 2.2.).

The format for a Ctab block is

Counts line	Important specifications here relate to the number of atoms, bonds, and atom lists, the chiral flag setting and the Ctab version
Atom block	Specifies the atomic symbol and any mass difference, charge, stereochemistry, and associated hydrogens for each atom
Bond block	Specifies the two atoms connected by the bond, the bond type, and any bond stereochemistry and topology (chain or ring properties) for each bond
Atom list block	Identifies the atom (number) of the list and the atoms in the list
Stext block	Structural text descriptor block used by CPSS programs
Properties block	Provides for future expandability of Ctab features, while maintaining compatibility with earlier Ctab configurations

The detailed format for each block outlined above follows.

Note: A blank *numerical* entry of any line should be read as "0" (zero). Spaces are significant and correspond to one or more of the following:

Absence of an entry
Empty character positions within an entry
Spaces between entries; single unless specifically noted otherwise

2.1. The Counts Line.

aaabbblllfffcccsssxxxrrrpppiiimmmvvvvvv

Field	Meaning	Values	Notes
x y z	atom coordinates		[G]
aaa	atom symbol	entry in periodic table or L for atom list, A, Q, * for unspecified atom, and LP for lone pair, or R# for Rgroup label	[G] [Q] [G] [3D] [Rg]
dd	mass difference	−3, −2, −1, 0, 1, 2, 3, 4 (0 if value beyond these limits)	[G] Difference from mass in periodic table. Wider range of values allowed by M ISO line, below. Retained for compatibility with older Ctabs, M ISO takes precedence.
ccc	charge	0 = uncharged or value other than these, 1 = +3, 2 = +2, 3 = +1, 4 = doublet (^), 5 = −1, 6 = −2, 7 = −3	[G] Wider range of values in M CHG and M RAD lines below. Retained for compatibility with older Ctabs, M CHG and M RAD lines take precedence.
sss	atom stereo parity	0 = not stereo, 1 = odd, 2 = even, 3 = either or unmarked stereo center	[G] Ignored when read. See Section 2.8 (Stereo Notes)
hhh	hydrogen count + 1	1 = H0, 2 = H1, 3 = H2, 4 = H3, 5 = H4	[Q] H0 means no H atoms allowed unless explicitly drawn. Hn means atom must have *n* or more H's in excess of explicit H's.
bbb	stereo care box	0 = ignore stereo config of this atom, 1 = stereo config of atom must match	[Q]
vvv	valence	0 = no marking (default) (1 to 14) = (1 to 14) 15 = zero valence	[G] Shows number of bonds to this atom, including bonds to implied H's.
HHH	H0 designator		[CP]
rrr	reaction component type	reactant = 1, product = 2, intermediate = 3	[CP]
iii	reaction component number	0 to (n−1)	[CP]
mmm	atom-atom mapping number	1–255	[Rx]
nnn	inversion/retention flag	1 = configuration is inverted, 2 = configuration is retained, 0 = property not applied	[Rx]
eee	exact change flag	1 = change on atom must be exactly as shown 0 = property not applied	[Rx] [Q]

Figure 3. Meaning of values in the atom block.

where

aaa	= number of atoms (current max 255) [G]
bbb	= number of bonds (current max 255) [G]
lll	= number of atoms lists (max 30) [Q]
fff	= (obsolete)
ccc	= chiral flag; 0 = not chiral, 1 = chiral [G]
sss	= number of stext entries [CP]
xxx	= number of reaction components + 1 [CP]
rrr	= number of reactants [CP]
ppp	= number of products [CP]
iii	= number of intermediates [CP]
mmm	= number of lines of additional properties, including the M END line [G]
vvvvvv	= current Ctab version: ′V2000′ [G]

For example, the counts line in the Ctab shown in Figure 2 shows six atoms, five bonds, the CHIRAL flag *on*, and three lines in the properties block:

```
  6  5  0  0  1  0            3 V2000
```

2.2. The Atom Block. Made up of atom lines, one line per atom with the following format:

```
xxxxx.xxxxyyyyy.yyyyzzzzz.zzzz aaaddcccssshhhbbbvvv
                                        HHHrrriiimmmnnneee
```

where the values are described in Figure 3.

Note: With Ctab version V2000, the dd and ccc fields have been superseded by the M ISO, M CHG, and M RAD lines in the properties block, described below. For compatibility, newer programs write appropriate values in both places if the values are in the old range and read the atom block fields if there are no M ISO, M CHG, or M RAD lines in the properties block.

2.3. The Bond Block. Made up of bond lines, one line per bond, with the following format:

```
111222tttsssxxxrrrccc
```

where the values are described in Figure 4.

2.4. The Atom List Block [Q]. Note: Newer programs use the M ALS item in the properties block in place of the atom list block. The atom list block is retained for compatibility, but information in an M ALS item supersedes atom list block information.

Field	Meaning	Values	Notes
111	first atom number		[G]
222	second atom number		[G]
ttt	bond type	1 = Single, 2 = Double, 3 = Triple, 4 = Aromatic, 5 = Single or Double, 6 = Single or Aromatic, 7 = Double or Aromatic, 8 = Any	[Q] Values 4 through 8 are for SSS queries only.
sss	bond stereo	Single bonds: 0 = not stereo, 1 = Up, 4 = Either, 6 = Down Double bonds: 0 = Use x-, y-, z-coords from atom block to determine cis or trans, 3 = Cis or trans (either)	[G] The small (pointed) end of the stereo bond is at the first atom (Field 111 above)
xxx	*not used*		
rrr	bond topology	0 = either, 1 = Ring, 2 = Chain	[Q] SSS queries only.
ccc	reacting center status	0 = unmarked, 1 = a center, −1 = not a center, Additional: 2 = no change, 4 = bond made/broken, 8 = bond order changes 12 = 4+8 (both made/broken and changes); 5 = (4 + 1), 9 = (8 + 1), and 13 = (12 + 1) are also possible	[Rx] (query only)

Figure 4. Meaning of values in the bond block.

Made up of atom list lines, one line per list, with the following format:

```
aaa kSSSSn 111 222 333 444 555
```

where

aaa	= number of atom (L) where list is attached
k	= T = [NOT] list, F = normal list
n	= number of entries in list; maximum is 5
111...555	= atomic number of each atom on the list
S	= space

2.5. The Stext Block [CP]. Made up of two-line entries.[6]

2.6. The Properties Block. Made up of ppp lines of additional properties, where ppp is the number in the counts line described above. If a version stamp is present, ppp is ignored and the file is read until an M END line is encountered.

Most lines in the properties block are identified by a prefix of the form M XXX where two spaces separate the M and XXX. Exceptions are

G XXX (two-line entry), A XXX (two-line entry), and V XXX which indicate CPSS properties (group abbreviation, atom alias, and atom value, respectively.[6] [CP]

S SKPnnn which causes the next nnn lines to be ignored.

The prefix M END terminates the properties block. All lines that are not understood by the program are ignored.

The descriptions below use the following conventions for values in field widths of 3:

n15	number of entries on line; value = 1–15
nn8	number of entries on line; value = 1–8
nn6	number of entries on line; value = 1–6
nn4	number of entries on line; value = 1–4
nn2	number of entries on line; value = 1 or 2
nn1	number of entries on line; value = 1
aaa	atom number; value = (1 to number of atoms)

The format for the properties included in this block follows. The format shows one entry; ellipses (...) indicate additional entries.

Charge [G]

```
M CHGnn8 aaa vvv ...
```

vvv	−15 to +15. Default of 0 = uncharged atom. When present, this property super-

sedes *all* charge and radical values in the atom block (Section 2.2), forcing a 0 charge on all atoms not listed in an M CHG or M RAD line.

Radical [G]

M RADnn8 aaa vvv ...

vvv — Default of 0 = no radical; 1 = singlet (:); 2 = doublet (∧); 3 = triplet (∧ ∧). When present, this property supersedes *all* charge and radical values in the atom block (section 2.2), forcing a 0 (zero) charge and radical on all atoms not listed in an M CHG or M RAD line.

Isotope [G]

M ISOnn8 aaa vvv ...

vvv — Absolute mass differing from natural abundance within the range −18 to +12. When present, this property supersedes *all* isotope values in the atom block. Default (no entry) is natural abundance.

Ring Bond Count [Q]

M RBDnn8 aaa vvv ...

vvv — Number of ring bonds allowed: default of 0 = off; −1 = no ring bonds (r0); −2 = as drawn (r*); 2 = (r2); 3 = (r3); 4 or more = (r4).

Substitution Count [Q]

M SUBnn8 aaa vvv ...

vvv — Number of substitutions allowed: default of 0 = off; −1 = no substitution (s0); −2 = as drawn (s*); 1, 2, 3, 4, 5 = (s1) through (s5); 6 or more = (s6).

Unsaturated Atom [Q]

M UNSnn8 aaa vvv ...

vvv — At least one multiple bond: default of 0 = off; 1 = on.

Link Atom [Q]

M LINnn4 aaa vvv bbb ccc ...

vvv,bbb,ccc — Link atom (aaa) and its substituents, other than bbb and ccc, may be repeated 1 to vvv times, (vvv ≥ 2).

Atom List [Q]

M ALS aaann5 e 11112222333344445555

aaa — Atom number; value = (1 to #atoms).
nn5 — Number of entries on line.
e — Exclusion, value is T if a 'NOT' list, F if a normal list.
1111... — Atom symbol of list entry in field of width 4.
Note: This line contains the atom symbol rather than the atom number used in the atom list block. Any information found in this item supersedes information from the atom list block.

Attachment Point [Rg]

M APOnn2 aaa vvv ...

vvv — Indicates whether atom aaa of the Rgroup member is the first attachment point (vvv = 1), second attachment point (vvv = 2), both attachment points (vvv = 3); default of 0 = no attachment.

Rgroup Label Location [Rg]

M RGPnn8 aaa rrr ...

rrr — Rgroup number, value from 1 to 32, labels position of Rgroup on root.

Rgroup Logic, Unsatisfied Sites, Range of Occurrence [Rg]

M LOGnn1 rrr iii hhh ooo

rrr — Rgroup number, value from 1 to 32.
iii — Number of another Rgroup which must only be satisfied if rrr is satisfied (IF rrr THEN iii).
hhh — RestH property of Rgroup rrr; default is 0 = off, 1 = on. If this property is applied (on), sites labeled with Rgroup rrr may only be substituted with a member of the Rgroup or with H.
ooo — Range of Rgroup occurrence required: *n* = exactly *n*; *n*–*m* = *n* through *m*; >*n* = greater than *n*; <*n* = fewer than *n*; default (blank) is > 0. Any noncontradictory combination of the preceding values is also allowed; for example: 1, 3–7, 9, >11.

Sgroup Type[3] [Sg]

M STYnn8 sss ttt ...

sss — Sgroup number.
ttt — SUP = superatom; MUL = multiple group; SRU = SRU type; MON = monomer; MER = mer type; COP = copolymer; CRO = crosslink; MOD = modification; GRA = graft; COM = component; MIX = mixture; FOR = formulation; DAT = data Sgroup; ANY = anypolymer; GEN = generic.
Note: For a given Sgroup, an STY line giving its type must appear before any other line that supplies information about it. For a data Sgroup, an SDT line must describe the data field before the SCD and SED lines that contain the data (see Data Sgroup Data below). When a data Sgroup is linked to another Sgroup, the Sgroup must already have been defined. (See Figure 5.)

Sgroup Subtype [Sg]

M SSTnn8 sss ttt ...

ttt — Polymer Sgroup subtypes: ALT = alternating; RAN = random, BLO = block.

Sgroup Labels [Sg]

M SLBnn8 sss vvv ...

vvv — Unique Sgroup identifier (integer label from 1 to 512).

Sgroup Connectivity [Sg]

M SCNnn8 sss ttt ...

ttt — HH = head-to-head; HT = head-to-tail; EU = either unknown. Left justified.

Sgroup Expansion [Sg]

M SDS EXPn15 sss ...

sss — Sgroup index of expanded superatoms.

Sgroup Atom List [Sg]

M SAL sssn15 aaa ...

aaa — Atoms in Sgroup sss.

Sgroup Bond List [Sg]

M SBL sssn15 bbb ...

bbb — Bonds in Sgroup sss (For data Sgroups, bbb's are the containment bonds, for all

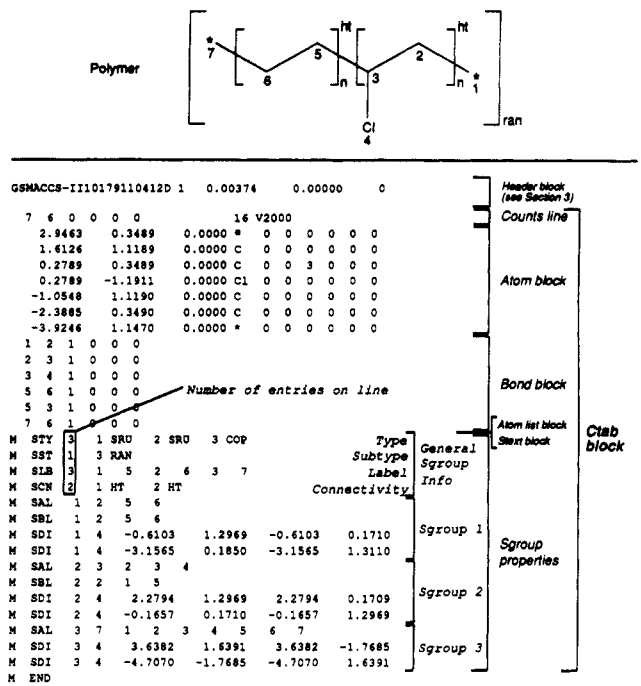

Figure 5. Connection table organization of an Sgroup structure.

other Sgroup types, bbb's are crossing bonds[3].)

Multiple Group Parent Atom List **[Sg]**

M SPA sssn15 aaa ...

aaa — Atoms in paradigmatic repeating unit of multiple group sss.

Sgroup Subscript **[Sg]**

M SMT sss m ...

m... — Text of subscript for Sgroup sss. For multiple groups, m ... is the text representation of the multiple group multiplier. For superatoms, m ... is the text of the superatom label.

Sgroup Correspondence **[Sg]**

M CRS sssnn6 bb1 bb2 bb3

bb1, bb2 — Crossing bonds that share a common bracket.

bb3 — Crossing bond in repeating unit that connects to bond bb1.

Sgroup Display Information **[Sg]**

M SDI sssnn4 x1 y1 x2 y2

x1,y1,x2,y2 — Coordinates of bracket endpoints (FORTRAN format 4F10.4).

Superatom Bond and Vector Information **[Sg]**

M SBV sss bb1 x1 y1

bb1 — Bond connecting to contracted superatom.

x1, y1 — Vector for bond bb1 connecting to contracted superatom sss (FORTRAN format 2F10.4).

Data Sgroup Field Description **[Sg]**

M SDT sss fff...fffgghhh...hhhiijjj...

sss — Index of data Sgroup.

f... — 30 character field name (in MACCS-II no blanks, commas, or hyphens).

gg — Field type (in MACCS-II: F = formatted, N = numeric, T = text).

h... — 20 character field units or format.

ii — Nonblank if data line is a query rather than Sgroup data: MQ = MACCS-II query, IQ = ISIS query, PQ = *program name code* query.

j... — Data relation operator (blank for MACCS-II).

Data Sgroup Display Information **[Sg]**

M SDD sss xxxxxxxxxxyyyyyyyyyy eeefgh i jjj kkk ll m noo

sss — Index of data Sgroup.

x, y — Coordinates (2F10.4).

eee — (Reserved for future use.)

f — Data display: A = attached, D = detached.

g — Absolute, relative placement: A = absolute, R = relative.

h — Display units: blank = no units displayed, U = display units.

i — (Reserved for future use.)

jjj — Number of characters to display (1...999 or ALL).

kkk — Number of lines to display (unused, always 1).

ll — (Reserved for future use.)

m — Tag character for tagged detached display (if nonblank).

n — Data display DASP position (1...9).

oo — (Reserved for future use.)

Data Sgroup Data **[Sg]**

M SCD sss d..

M SED sss d..

d... — Line of data for data Sgroup sss (69 chars/line, columns 12–80)

Note: A line of data is entered as text in 69-character substrings. Each SCD line adds 69 characters to a text buffer (starting with successive SCDs at character positions 1, 70, and 139). Following zero or more SCDs must be an SED, which may supply a final 69 characters. The SED initiates processing of the buffered line of text: trailing blanks are removed and right truncation to 200 characters is performed, numeric and formatted data are validated, and the line of data is added to data Sgroup sss. Left justification is not performed.

A data Sgroup may have more than one line of data, so more than one set of SCD and SED lines can be present for the same data Sgroup. The lines are added in the same order in which they are encountered.

If 69 or fewer characters are to be entered on a line, they may be entered with a single SED not preceded by an SCD. On the other hand, if desired, a line may be entered on up to 3 SCDs followed by a blank SED that terminates the line. The set of SCD and SED lines describing one line of data for a given data Sgroup must appear together, with no intervening lines for other data Sgroups' data.

Sgroup Hierarchy Information **[Sg]**

M SPLnn8 ccc ppp ...

ccc — Sgroup index of the child Sgroup.

ppp — Sgroup index of the parent Sgroup (ccc and ppp must already be defined via an STY line

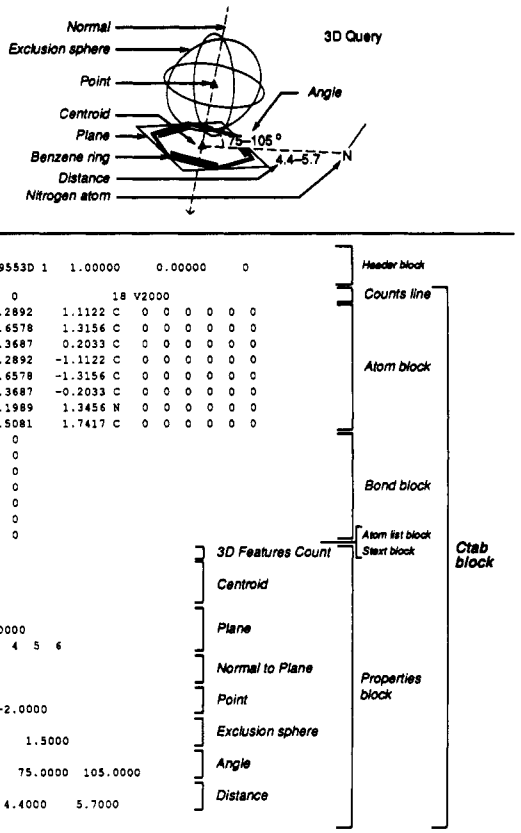

Figure 6. Connection table organization of a 3D query.

prior to encountering this line).

Sgroup Component Numbers **[Sg]**

M SNCnn8 sss ooo ...

sss	Index of component Sgroup.
ooo	Integer component order (1...256).

3D Feature Properties **[3D]**

M $3Dnnn

M $3D...

See below for information on the properties block of a 3D MOLfile. These lines must all be contiguous.

End of Block.

M END

This entry goes at the end of the properties block and is required for MOLfiles which contain a version stamp in the counts line.

2.7. The Properties Block for 3D Features [3D]. For each 3D feature, the properties block includes

one 3D features count line
one or more 3D features detail lines

The characters M $3D appear at the beginning of each line describing a 3D feature. The information for 3D features starts in column 7.

2.7.1. 3D Features Count Line. The first line in the properties block is the 3D features count line and has the following format:

M $3Dnnn

where nnn is the number of 3D features on a model.

2.7.2. 3D Features Detail Lines. The lines following the 3D features count line describe each 3D feature on a model. Each 3D feature description consists of an identification line and one or more data lines. The identification line is the first line and contains the 3D feature's type identifier, color, and name. Each data line describes the construction of the 3D feature.

The Identification Line. The 3D feature identification line has the following format:

M $3Dtttccc aaa...aaa ttt...ttt

where the variables represent

ttt	3D feature type
ccc	Color number (an internal MDL number which is terminal dependent)
aaa...aaa	3D feature name (up to 32 characters)
ttt...ttt	Text comments (up to 32 characters) used by MDL programs (see section 2.7.3 below)

The 3D feature type identifiers are

−1	Point defined by 2 points and a distance in angstroms (Å)
−2	Point defined by 2 points and a percentage
−3	Point defined by a point, a normal line, and a distance
−4	Best fit line defined by 2 or more points
−5	Plane defined by 3 or more points. (A best fit plane if more than three points)
−6	Plane defined by a point and a line
−7	Centroid defined by points
−8	Normal line defined by a point and a plane
−9	Distance defined by 2 points and a range (Å)
−10	Distance defined by a point, a line, and a range (Å)
−11	Distance defined by a point, a plane, and a range (Å)
−12	Angle defined by 3 points and a range (in deg)
−13	Angle defined by 2 intersecting lines and a range (in deg)
−14	Angle defined by 2 intersecting planes and a range (in deg)
−15	Dihedral angle defined by 4 points and a range (in deg)
−16	Exclusion sphere defined by a point and a distance (Å)
−17	Fixed atoms in the model
nnn	Positive integer indicates atom or atom-pair data constraints

The Data Line. The 3D feature defines the data line format. Each 3D object is treated as a pseudoatom and identified in the connection table by a number. The 3D object numbers are assigned sequentially, starting with the next number greater than the number of atoms. The data line formats for the 3D feature types are

type	description of data line
−1	The data line for a point defined by 2 points and a distance (Å) has the following format:

M $3Diiijjjddddd.dddd

where the variables represent:

iii	ID number of a point
jjj	ID number of a second point
ddddd.dddd	Distance from first point in direction of second point (Å), 0 if not used.

The following example shows POINT_1 created from the atoms 1 and 3 with a constraint distance of 2 Å. The first line is the identification line. The second line is the data line.

M $3D −1 4 POINT_1

M $3D 1 3 2.0000

−2	The data line for a point defined by two points and a percentage has the format:

M $3Diiijjjddddd.ddddd

where the variables represent:

iii	ID number of a point
jjj	ID number of a second point
ddddd.dddd	Distance (fractional) relative to distance between first and second points, 0 if not used.

−3 The data for a point defined by a point, a normal line, and a distance (Å) has the format:

M $3Diiilllddddd.dddd

where the variables represent:

iii	ID number of a point
lll	ID number of a normal line
ddddd.dddd	Distance (Å), 0 if not used. **Note**: For chiral models, the distance value is signed to specify the same or opposite direction of the normal.

−4 The data lines for a best fit line defined by 2 or more points have the following format:

M $3Dpppttttt.tttt

M $3Diiijjj...zzz

...

where the variables represent:

ppp	Number of points defining the line
ttttt.tttt	Deviation (Å), 0 if not used.
iii	Each iii, jjj, and zzz is the ID number of an item in the
jjj	model that defines the line
...	(to maximum of 20 items per
zzz	data line).

The following line is defined by the four points 1, 14, 15, and 19 and has a deviation of 1.2 Å. The first line is the identification line. The second and third lines are the data lines.

M $3D −4 2 N_TO_AROM

M $3D 4 1.2000

M $3D 1 14 15 19

−5 The data lines for a plane defined by three or more points (a best fit plane if more than three points) have the following format:

M $3Dpppttttt.tttt

M $3Diiijjj...zzz

...

where the variables represent:

ppp	Number of points defining the plane
ttttt.tttt	Deviation (Å), 0 if not used.
iii	Each iii, jjj, and zzz is the ID number of an item in the
jjj	model that defines the
...	plane (to maximum of 20
zzz	items per data line).

The following plane is defined by three points. The first line is the identification line. The second and third lines are the data lines.

M $3D −5 4 PLANE_2

M $3D 3

M $3D 1 5 14

−6 The data line for a plane defined by a point and a line has the following format:

M $3Diiilll

where the variables represent:

iii	ID number of a point
lll	ID number of a line

The following plane is defined by the point 1 and the plane 16. The first line is the identification line. The second line is the data line.

M $3D −6 3 PLANE_1

M $3D 1 16

−7 The data lines of a centroid defined by points have the following format:

M $3Dppp

M $3Diiijjj...zzz

...

where the variables represent:

ppp	Number of points defining the centroid
iii	Each iii, jjj, and zzz is the ID number of an item in the
jjj	model that defines the
...	centroid (to maximum of 20
zzz	items per data line).

The following centroid, ARO_CENTER, is defined by three items 6, 8, and 10. The first line is the identification line. The second and third lines are the data lines.

M $3D −7 1 ARO_CENTER

M $3D 3

M $3D 6 8 10

−8 The data line for a normal line defined by a point and a plane have the following format:

M $3Diiijjj

where the variables represent:

iii	ID number of a point
jjj	ID number of a plane

The following normal line, ARO_NORMAL, is defined by the point 14 and the plane 15. The first line is the identification line. The second line is the data line.

M $3D −8 1 ARO_NORMAL

M $3D 14 15

−9 The data line for a distance defined by two points and a range (Å) has the following format:

M $3Diiijjjddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a point
jjj	ID number of a second point
ddddd.dddd	Minimum distance (Å)
zzzzz.zzzz	Maximum distance (Å)

The following distance, *L*, is between items 1 and 14 and has a minimum distance of 4.9 Å and a maximum distance of 6.0 Å. The first line is the identification line. The second line is the data line.

M $3D -9 6 *L*

M $3D 1 14 4.9000 6.0000

-10 The data line for a distance defined by a point, a line, and a range (Å) has the format:

M $3Diiilllddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a point
lll	ID number of a line
ddddd.dddd	Minimum distance (Å)
zzzzz.zzzz	Maximum distance (Å)

-11 The data line for a distance defined by a point, a plane, and a range (Å) has the format:

M $3Diiijjjddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a point
jjj	ID number of a plane
ddddd.dddd	Minimum distance (Å)
zzzzz.zzzz	Maximum distance (Å)

-12 The data line for an angle defined by three points and a range (in deg) has the following format:

M $3Diiijjjkkkddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a point
jjj	ID number of a second point
kkk	ID number of a third point
ddddd.dddd	Minimum degrees
zzzzz.zzzz	Maximum degrees

The following angle, THETA1, is defined by the three points 5, 17, and 16. The minimum angle is 80° and the maximum is 105°. The first line is the identification line. The second line is the data line.

M $3D-12 5 THETA1

M $3D 5 17 16 80.0000 105.0000

-13 The data line for an angle defined by two lines and a range (in deg) has the following format:

M $3Dlllmmmddddd.ddddzzzzz.zzzz

where the variables represent:

lll	ID number of a line
mmm	ID number of a second line
ddddd.dddd	Minimum degrees
zzzzz.zzzz	Maximum degrees

THETA2 is defined by the lines 27 and 26 with maximum and minimum angles of 45° and 80°. The first line is the identification line. The second line is the data line.

M $3D-13 5 THETA2

M $3D 27 26 45.0000 80.0000

-14 The data line for an angle defined by two planes and a range (in deg) has the following format:

M $3Diiijjjddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a plane
jjj	ID numbers of a second plane
ddddd.dddd	Minimum degrees
zzzzz.zzzz	Maximum degrees

-15 The data line for a dihedral angle defined by four points and a range (in deg) has the following format:

M $3Diiijjjkkkllldddddd.ddddzzzzz.zzzz

where the variables represent:

iii	ID number of a point
jjj	ID number of a second point
kkk	ID number of a third point
lll	ID number of a fourth point
ddddd.dddd	Minimum degrees
zzzzz.zzzz	Maximum degrees

DIHED1 is defined by the items 7, 6, 4, and 8 with minimum and maximum angles of 45° and 80°, respectively. The first line is the identification line. The second line is the data line.

M $3D-15 5 DIHED1

M $3D 7 6 4 8 -45.0000 80.0000

-16 The data lines for an exclusion sphere defined by a point and a distance (Å) have the following format:

M $3Diiiuuuaaaaddddd.dddd

M $3Daaabbb...zzz

...

where the variables represent:

iii	ID number of the center of the sphere
uuu	1 or 0. 1 means unconnected atoms are ignored within the exclusion sphere during a search; 0 otherwise.
aaa	Number of allowed atoms
ddddd.dddd	Radius of sphere (Å)
iii	Each iii, jjj, and zzz is an
jjj	ID number of an allowed
...	atom (to maximum of 20
zzz	items per data line).

The following exclusion sphere is centered on atom 24, has a radius of 5, and allows atom 9 within the sphere. The first line is the identification line. The second and third lines are the data lines.

M $3D-16 7 EXCL_SPHERE

M $3D 24 0 1 5.0000

M $3D 9

-17 The data lines of the fixed atoms have the following format:

M $3Dppp

M $3Diiijjj...zzz

...

where the variables represent:

ppp	Number of fixed points
iii	Each iii, jjj, and zzz is an
jjj	ID number of a fixed atom
...	(to maximum of 20 items per
zzz	data line)

The following example shows four fixed atoms. The first line is the identification line. The second and third lines are the data lines.

M $3D-17

M $3D 4

M $3D 3 7 12 29

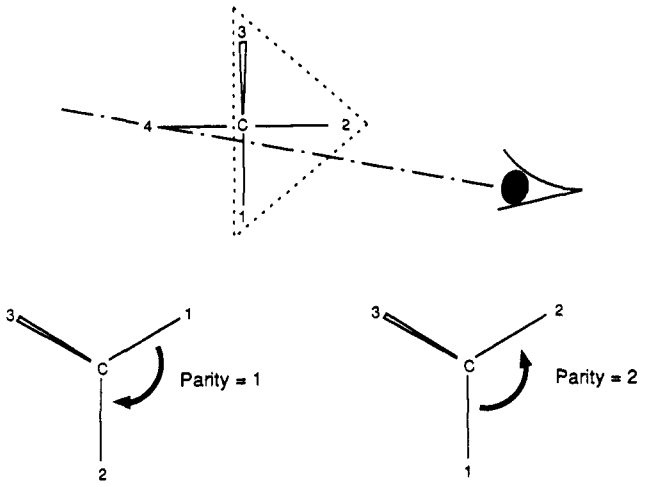

Figure 7. Illustration of parity.

2.7.3. 3D Data Constraints [3D] [Q]. A positive integer is used as a type identifier to indicate an atom or atom-pair data constraint. Two lines are used to describe a data constraint. The lines have the following format:

M $3Dnnncccaaa...aaabbbbbbbbbppppppppppsss...sss

M $3Diiijjjddd...ddd

where the variables represent:

nnn	Database-field number
ccc	Color
aaa...aaa	Database-field name (up to 30 characters)
bbbbbbbb	/BOX = box-number (source of data) (up to 8 characters)
ppppppppp	/DASP = *n*1,*n*2 where *n*1 and *n*2 are digits from 1 to 9 (data size and position) (up to 9 characters)
sss...sss	/DISP = 3DN (name), 3DV (value), 3DQ (query), NOT (no text). First three in any combination to maximum total of 15 characters
iii	ID number of an atom
jjj	ID number of a second atom for atom-pair data, 0 if data is atom data
ddd...ddd	Data constraint (based on format from database) (up to 64 characters)

The following example shows a numeric data constraint for the field CNDO.CHARGE on atom 12. The first line is the identification line. The second line is the data line.

M $3D 7 0 CNDO.CHARGE

M $3D 12 0 −0.3300 −0.1300

The following example shows a numeric data constraint for the field BOND.LENGTH on the atom pair 1 and 4. The first line is the identification line. The second line is the data line.

M $3D 9 0 BOND.LENGTH

M $3D 1 4 2.0500 1.8200

The following example shows a data constraint allowing any charge value for the field CHARGE on all the atoms. The first line is the identification line. The second line is the data line.

M $3D 12 0 CHARGE

M $3D999 0 @

2.8. Stereo Notes. Parity is illustrated in Figure 7. A bond attached at a stereo center is marked up or down to define the configuration. Number the atoms surrounding the stereo center with 1, 2, 3, and 4 in order of increasing atom number (position in the atom block) (a hydrogen atom should be made number 4). View the center from a position such that the bond

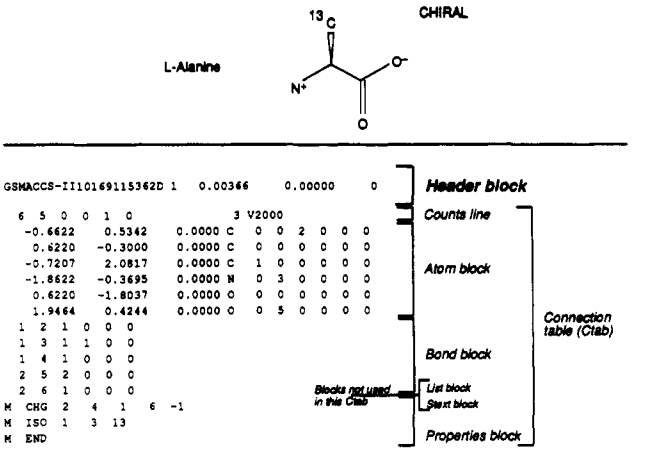

Figure 8. MOLfile organization of alanine.

connecting the highest-numbered atom (4) projects behind the plane formed by atoms 1, 2, and 3. **Note**: In Figure 7, atoms 1, 2, and 4 are all in the plane of the paper and atom 3 is above the plane.

Sighting toward atom number 4 through the plane (1,2,3), you see that the three remaining atoms can be arranged in either a clockwise or counterclockwise direction in ascending numerical order.

The Ctab lists a parity value of 1 for a clockwise arrangement at the stereo center and 2 for counterclockwise. A center with an Either bond has a parity value of 3. An unmarked stereocenter is also assigned a value of 3. The example at the top of Figure 7 has a parity value of 2.

3. MOLECULE FILES [MOLFILES]

A MOLfile consists of a header block and a connection table. Figure 8 shows the structure and corresponding MOLfile of alanine.

The format for a MOLfile is

Header block: This identifies the MOLfile with the molecular name, user's name, program, date, and other miscellaneous information and comments

Ctab block (described in Section 2)

The detailed format for the header block follows.

3.1. The Header Block.

Line 1 Molecule name. This line is unformatted, but like all other lines in a MOLfile may not extend beyond column 80.

Line 2 User's first and last initials (I), program name (P), date/time (M/D/Y,H:m), dimensional codes (d), scaling factors (S, s), energy (E) if modeling program input, internal registry number (R) if input through MDL form. This line has the format:

```
IIPPPPPPPPMMDDYYHHmmddSSssssssssssEEEEEEEEEEEERRRRRR
(FORTRAN:  A2    A8  <--A10--->A2I2  F10.5      F12.5   I6  )
```

A blank line can be substituted for line 2.

Line 3 A line for comments. If no comment is entered, a blank line must be present.

4. RGROUP QUERY FILES [RGFILES]

The format of an RGfile (Rgroup query file) is shown in Figure 9. Lines beginning with $ define the overall structure of the Rgroup query; the MOLfile header block is embedded in the Rgroup header block.

In addition to the primary connection table (Ctab block) for the root structure, a Ctab block defines each member within each Rgroup. Figure 10 shows the structure and corresponding RGfile of an Rgroup query.

```
$MDL REV 1 date/time
$MOL
$HDR
[MOLfile Header Block (see Section 3) = name, pgm info, comment]
$END HDR
$CTAB
[Ctab Block (see Section 2) = count + atoms + bonds + lists + props]
$END CTAB
$RGP
 rrr   [where rrr = Rgroup number]
$CTAB
*r
 *m   [Ctab Block]
$END CTAB
$END RGP
$END MOL
```

Figure 9. Format of an RGfile. The Rgroup format (*r) is repeated up to a maximum of 32. The Rgroup member format (*m) is repeated up to a maximum of 255 total atoms and bonds in the Rgroup.

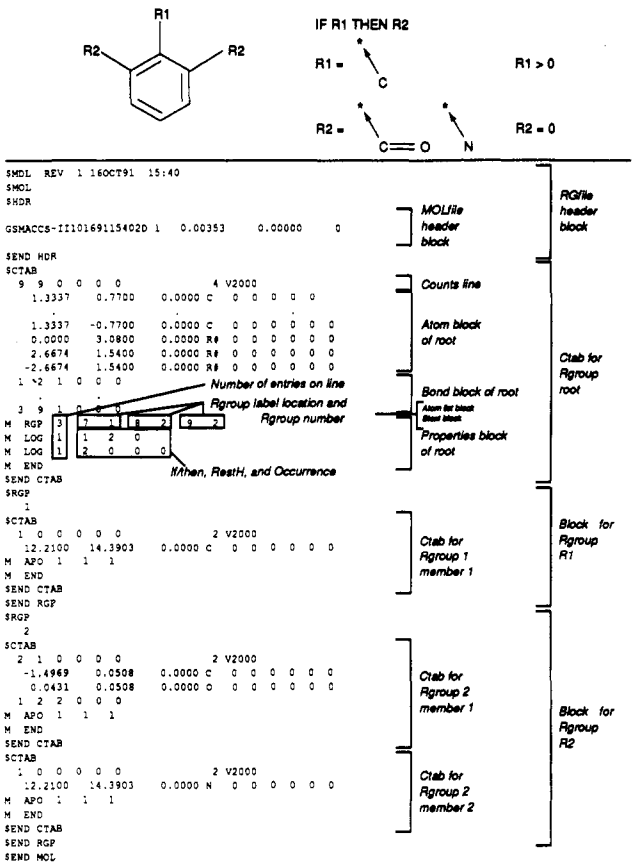

Figure 10. Structure and RGfile of an Rgroup query.

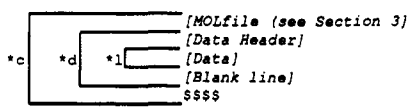

Figure 11. Format of an SDfile. The compound format (*c) is repeated for the length of the SDfile. The data item format (*d) is repeated for each data item in a compound. A separate line (*l) is used for each data value.

5. STRUCTURE-DATA FILES [SDFILES]

An SDfile (structure-data file) contains the structural information and associated data items for one or more compounds. The format for an SDfile is shown in Figure 11 and an example of an SDfile is given in Figure 12.

A [*MOLfile*] block has the MOLfile format described in Section 3.

A [*Data Header*] (one line) precedes each item of data, starts with a *greater than* (>) sign and contains at least one of the following:

The field name enclosed in angle brackets. For example: ⟨melting.point⟩.

The field number, DT*n*, where *n* represents the number assigned to the field in the database.

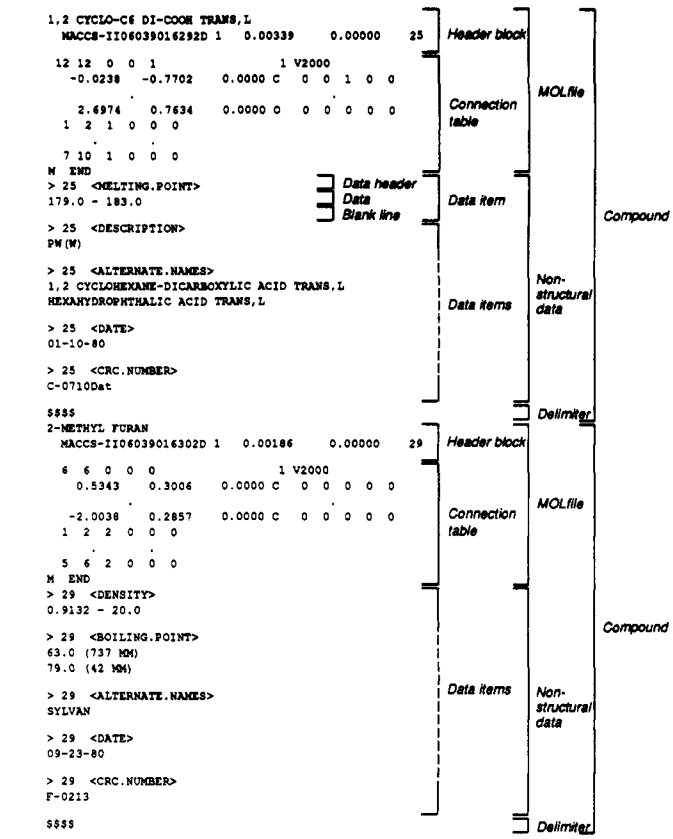

Figure 12. Example of an SDfile.

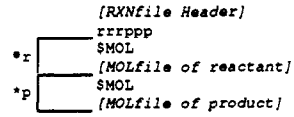

Figure 13. Format of a RXNfile. The reactant MOLfile format (*r) and product MOLfile format (*p) are repeated for each reactant and MOLfile, respectively.

Optional information for the data header includes:

The compound's external and internal registry numbers. External registry numbers must be enclosed in parentheses.

Any combination of information.

The following are examples of valid data headers:

```
> <MELTING.POINT>
> 55      (MD-08974)      <BOILING.POINT>   DT12
> DT12    55
> (MD-0894)    <BOILING.POINT>   FROM ARCHIVES
```

A [*Data*] value may extend over multiple lines containing up to 200 characters each. A blank line terminates each data item.

A line containing four dollar signs ($$$$) terminates each complete data block describing a compound.

A DATfile (data file) is an SDfile with no [*MOLfile*] descriptions or $$$$ delimiters. The [*Data Header*] in a DATfile must include either an external or internal registry number in addition to a field name or number.

6. REACTION FILES [RXNFILES]

RXNfiles contain structural data for the reactants and products of a reaction. Figure 13 shows the format for a RXNfile. Figure 14 shows a simple reaction and the corresponding RXNfile.

6.1. Header Block.

Line 1 $RXN in the first position on the line identifies the file as a reaction file.

Line 2 A line which is always blank.

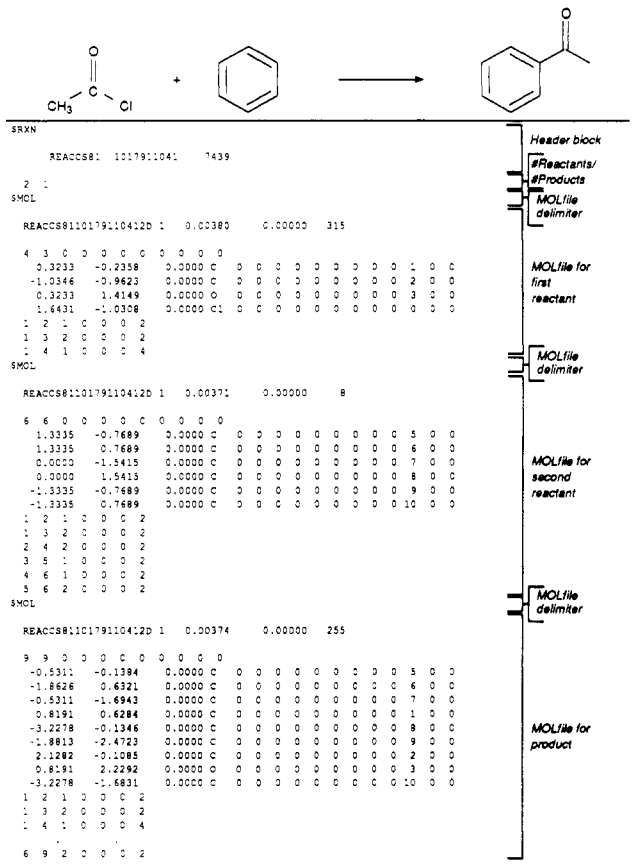

Figure 14. Simple reaction and its corresponding RXNfile.

Line 3 The program name and version (P), date/time (M/D/Y,H:m), and reaction registry number (R). This line has the format:

```
                    PPPPPPPPPMMDDYYHHmmRRRRRRRR
(FORTRAN:       A14  <--A12--->   I8    )
```

A blank line can be substituted for line 3.

Line 4 A line for comments. If no comment is entered, a blank line must be present.

6.2. Reactants/Products. A line identifying the number of reactants and products in that order. The format is

rrrppp

where the variables represent:

rrr Number of reactants

ppp Number of products

6.3. MOLfile Blocks. A series of blocks, each starting with $MOL as a delimiter, giving the MOLfile for each reactant and product in turn. The MOLfile blocks are always in the same order as the molecules in the reaction; reactants first and products second.

7. REACTION-DATA FILES [RDFILES]

An RDfile (reaction-data file) consists of a set of records. Each record defines a molecule or reaction, and its associated data. Figure 15 shows the format for an RDfile. An example RDfile incorporating the RXNfile described in Section 6 is shown in Figure 16.

Each logical line in an RDfile starts with a keyword in column 1 of a physical line. One or more blanks separate the first argument (if any) from the keyword. The blanks are ignored when the line is read. After the first argument, blanks are significant.

An argument longer than 80 characters breaks at column 80 and continues in column 1 of the next line. (The argument may continue on additional lines up to the physical limits on text length imposed by the database.)

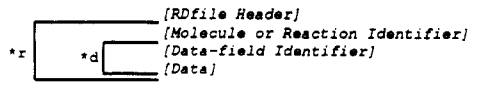

Figure 15. Format of an RDfile. The molecule or reaction format (*r) is repeated for each molecule or reaction. The data item format (*d) is repeated for each data item in a molecule or reaction.

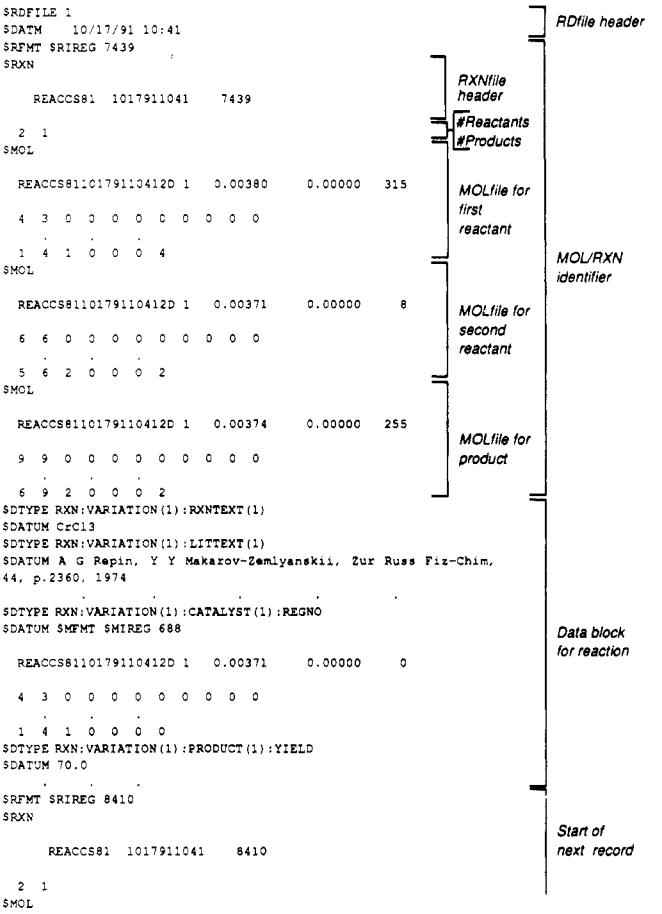

Figure 16. Example of a reaction RDfile incorporating the RXNfile described in Section 6.

The RDfile must not contain any blank lines except as part of embedded MOLfiles, RXNfiles, or data. An identifier separates records.

7.1. RDfile Header.

Line 1 $RDFILE 1: The [*Rdfile Header*] must occur at the beginning of the physical file and identifies the file as an RDfile. The version stamp "1" is intended for future expansion of the format.

Line 2 $DATM: Date/time (M/D/Y, H:m) stamp. This line is treated as a comment and ignored when the program is read.

7.2. Molecule and Reaction Identifiers. A [*Molecule or Reaction Identifier*] defines the start of each complete record in an RDfile. The form of a *molecule* identifier must be one of the following:

$MFMT [$MIREG *internal-regno* [$MEREG *external-regno*]] *embedded MOLfile*

$MIREG *internal-regno*

$MEREG *external-regno*

where

$MFMT defines a molecule by specifying its connection table as a MOLfile

$MIREG *internal-regno* is the internal registry number (sequence number in the database) of the molecule

$MEREG *external-regno* is the external registry number of the molecule (any uniquely identifying character string known to the database, for example, CAS number)

Square brackets ([]) enclose optional parameters.

An embedded MOLfile (see Section 3) follows immediately after the $MFMT line.

The forms of a *reaction* identifier closely parallel that of a molecule:

$RFMT [$RIREG *internal-regno* [$REREG *external-regno*]] *embedded RXNfile*

$PCRXN [$RIREG *internal-regno* [$REREG *external-regno*]] *embedded CPSS*

RXNfile [CP]

$RIREG *internal-regno*

$REREG *external-regno*

where

$RFMT defines a reaction by specifying its description as a RXNfile and $PCRXN [CP] defines a reaction by specifying its description as a CPSS-style RXNfile

$RIREG *internal-regno* is the internal registry number (sequence number in the database) of the reaction

$REREG *external-regno* is the external registry number of the reaction (any uniquely identifying character string known to the database)

Square brackets ([]) enclose optional parameters

An embedded RXNfile (see Section 6) follows immediately after the $RFMT line, and an embedded CPSS-style RXNfile follows immediately after the $PCRXN [CP] line

7.3. Data-Field Identifier. The [*Data-field Identifier*] specifies the name of a data field in the database. The format is

$DTYPE *field name*

7.4. Data. Data associated with a field follows the field name on the next line and has the form

$DATUM *datum*

The format of *datum* depends upon the data type of the field as defined in the database. For example: integer, real number, real range, text, molecule regno.

For fields whose data type is "molecule regno", the *datum* must specify a molecule and, with the exception noted below, use one of the formats defined above for a molecular identifier. For example

$DATUM $MFMT *embedded MOLfile*

$DATUM $MEREG *external-regno*

$DATUM $MIREG *internal-regno*

In addition, the following special format is accepted

$DATUM *molecule-identifier*

Here, *molecule-identifier* acts in the same way as *external-regno* in that it can be any text string known to the database that uniquely identifies a molecule. (It is usually associated with a data field different from the *external-regno*.)

8. CONCLUSION

A series of chemical structure file formats built up from one or more connection table blocks have been described. These formats allow for the storage and transfer of chemical structure information used typically for search queries, individual structures, or entire databases. It is hoped that these file formats will see even wider use.[6]

REFERENCES AND NOTES

(1) The various CTfile formats have been programmed, tested, and documented by a large number of people at MDL over the years. Besides the authors of this paper, these include S. Anderson, J. Barstow, R. Blackadar, T. A. Blackadar, R. Briggs, R. E. Carhart, B. D. Christie, J. D. Dill, G. Freitas, R. J. Greenberg, A. J. Gushurst, D. Henry, R. Hofmann, D. Horner, A. Hui, T. E. Moock, D. G. Raich, J. Steele, W. T. Wipke, and K. Wiseman-Sleeter.

(2) Wipke, W. T.; Nourse, J. G.; Moock, T. Generic Queries in the MACCS System. In *Computer Handling of Generic Chemical Structures*; Barnard, J. M., Ed.; Gower: Hampshire, 1984; pp 167-178.

(3) Gushurst, A. J.; Nourse, J. G.; Hounshell, W. D.; Leland, B. A.; Raich, D. G. The Substance Module: The Representation, Storage, and Searching of Complex Structures. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 447-454.

(4) Moock, T. E.; Christie, B.; Henry, D. MACCS-3D: A New Database System for Three-Dimensional Molecular Models in Chemical Information Systems. In *Beyond the Structure Diagram*; Bowden, D., Mitchell, E. M., Eds.; Ellis Howard: New York, 1990; pp 42-49.

(5) Moock, T. E.; Nourse, J. G.; Grier, D.; Hounshell, W. D. The Implementation of Atom-Atom Mapping and Related Features in the Reaction Access System (REACCS). In *Chemical Structures: The International Language of Chemistry*; Warr, W. A., Ed.; Springer-Verlag: New York, 1988; pp 303-313.

(6) For more details and information on future changes, contact Affinity, Molecular Design Limited, 2132 Farallon Drive, San Leandro, CA 94577.

Computer-Aided Molecular Formula Determination from Mass, ^{1}H, and ^{13}C NMR Spectra

B. G. DERENDJAEV,* S. A. NEKHOROSHEV, K. S. LEBEDEV, and S. P. KIRSHANSKY

Novosibirsk Institute of Organic Chemistry, USSR Academy of Sciences, Novosibirsk, USSR

Received March 5, 1991

A computer-aided technique for the determination of the molecular formula of a compound by its mass, ^{13}C, and ^{1}H NMR spectra is suggested. Efficiency of the method has been verified on 81 "unknowns". It has been shown that in 89% of instances the requested formula is found among the top three candidates of a computer answer, and in 45% of instances the computer suggests a single formula.

The use of computer systems for structure elucidation of organic compounds from a spectral data set is generally based on a known or assumed molecular formula.[1-3] This information was obtained by additional experiments (high-resolution mass spectrometry, CHN analysis, chemical analysis, etc.) or is postulated by the researcher from the background of the sample.

In the context of our work on a spectral data analysis system,[4-10] we have developed software to determine molecular formulas directly from analysis of the most simple and ac-

0095-2338/92/1632-0255$03.00/0 © 1992 American Chemical Society