

The GROMOS Software for (Bio)Molecular Simulation



Volume 4: Data Structures and Formats

August 24, 2012

REVIEW VERSION

This is only a preview version of the GROMOS Manual and User Guide.
The full version can be purchased on <http://www.gromos.net/>

PREVIEW VERSION

Contents

Chapter 1. Introduction	4-1
Chapter 2. Block Structure and Title Record of GROMOS Files	4-3
Chapter 3. Topological Information	4-5
3.1. Introduction	4-5
3.2. Molecular Topology	4-6
3.3. Perturbation Molecular Topology	4-18
3.4. Atom-Atom Distance Restraints	4-23
3.5. Dihedral Angle Restraints or Constraints	4-26
3.6. ³ J-Coupling Constant Restraints	4-27
3.7. Crystallographic Restraints	4-28
3.8. Local-Elevation Coordinates	4-31
3.9. Local Elevation Umbrella Sampling Database File	4-32
3.10. Atomic Friction Coefficients	4-32
3.11. Position Restraining or Constraining Atom Specification List	4-32
3.12. B-factor Restraining	4-33
3.13. Backwards Compatibility with GROMOS96	4-34
Chapter 4. Configurational Information	4-37
4.1. Introduction	4-37
4.2. Atomic Coordinates	4-38
4.3. Atomic Velocities	4-39
4.4. Atomic Forces	4-40
4.5. Atomic Stochastic Integrals	4-40
4.6. Periodic Box	4-41
4.7. Nose-Hoover Chain Thermostat and Barostat Variables	4-42
4.8. Roto-Translational Constraints Reference Variables	4-43
4.9. Perturbation Data	4-44
4.10. Atom-Atom Distance Restraints	4-44
4.11. ³ J-Coupling Constant Restraints	4-45
4.12. Crystallographic Restraints	4-46
4.13. Local-Elevation Data	4-46
4.14. Ball and Stick Local-Elevation Data	4-47
4.15. Time or Step Number Data	4-50
4.16. Energies, Pressure, Volume and Free Energy Data in PROMD	4-50
4.17. Energies, Pressure, Volume and Free Energy Data for MD++	4-57
4.18. Atomic B-Factors and Positional Fluctuations	4-62
4.19. Backwards Compatibility with GROMOS96	4-62
Chapter 5. Molecular Topology Building Blocks	4-65
5.1. Introduction	4-65
5.2. Separate Molecules	4-65
5.3. Linking of Building Blocks	4-73
5.4. Other Building Blocks	4-73
5.5. End-Groups	4-74
5.6. Backwards Compatibility with GROMOS96	4-74
5.7. Contents of the MTB File	4-75

Chapter 6. Interaction Function Parameters	4-77
6.1. Introduction	4-77
6.2. Mass Atom Types	4-77
6.3. Covalent Bond-Stretching Interaction Parameters	4-78
6.4. Covalent Bond-Angle Bending Interaction Parameters	4-78
6.5. Improper Dihedral-Angle Interaction Parameters	4-78
6.6. Dihedral-Angle Torsional Interaction Parameters	4-79
6.7. Van der Waals Interaction Parameters and Integer Atom Codes	4-79
6.8. Atomic Charges and Charge Group Codes	4-81
6.9. Excluded Neighbours	4-81
6.10. Backwards Compatibility with GROMOS96	4-81
6.11. Contents of the IFP File	4-81
Chapter 7. Library Files for GROMOS++	4-83
7.1. Introduction	4-83
7.2. Interaction Function Parameter Renumbering	4-83
7.3. Atomic Naming Conventions	4-84
7.4. Definition of File-Names and Joblists	4-85
7.5. Energy Trajectory Block Definition	4-87
7.6. Hydrogen-Bond Donors and Acceptors	4-88
7.7. Crystallographic Transformations	4-88
7.8. NOE Analysis	4-89
7.9. SASA Implicit Solvent Model	4-91
Chapter 8. Input File for PROMD and MD++	4-93
Chapter 9. Output Files for PROMD and MD++	4-121
Chapter 10. Files Accessed by PROMD and MD++ for Reading or Writing	4-123
Chapter 11. Other Non-GROMOS Formats	4-129
Chapter 12. List of GROMOS Blocknames	4-131
Chapter 13. Recommendations for Standard Input and Output File Names	4-135
Bibliography	4-i

CHAPTER 1

Introduction

GROMOS knows different types of data and data files, which are described in this volume. Two types of information concerning a molecular system can be distinguished.

1. *Topological information*: data on the covalent structure, atomic masses, charges, van der Waals parameters, atom-atom distance restraints specification, 3J -value restraints specification, local-elevation coordinate specification, etc.
2. *Configurational information*: atomic coordinates and atomic coordinate dependent or related quantities, such as velocities and forces, atom-atom distances, dihedral angles, 3J -values, energies, size of the computational box, etc.

These two types of information are generally stored in separate files, since configurations change continuously during a simulation, whereas molecular topological data generally do not change. Both types of files, *topological files* and *configurational files*, for a specific molecular system are related through the requirement that in both the sequence of the quantities is the same, e.g.

1. sequence of atoms
2. sequence of atom-atom distance restraints
3. sequence of dihedral angle restraints
4. sequence of 3J -value restraints

This identity of sequence could be checked e.g. by comparing atom names occurring in topological files with those from the configurational files. However, in order to avoid dependence on naming conventions and to maintain maximum flexibility, this is not done in the GROMOS programs. When molecular information, such as residue numbers and names or atom sequence numbers or names, is present both in a topological file and in a configurational file of a molecular system, the program generally uses the data from the topological file and ignores the corresponding data on the configurational file.

The units of the quantities contained in the different files are all derived from the basic units: nm (length), ps (time), atomic mass units and electronic charge, leading in particular to $\text{kJ} \cdot \text{mol}^{-1}$ as unit of energies. The angles are always given in degrees in the files. In PROMD, however, all charges are internally amplified by a factor $(4\pi\epsilon_0)^{-1/2}$ and all angles are generally converted to radians (the box angles being an exception). The above conversions are not performed in MD++ and GROMOS++.

GROMOS data files have a block structure, which is defined in Chap. 2. Topological quantities, variables, blocks and files are described in Chap. 3. Configurational quantities, variables, blocks and files are described in Chap. 4. Two other types of data, molecular topology building blocks and interaction function parameters are described in Chap. 5 and Chap. 6. Library files to be used by the analysis programs of GROMOS ++ are described in Chap. 7. Chap. 8 describes the PROMD/MD++ input file.

This is only a preview version of the GROMOS Manual and User Guide.
The full version can be purchased on <http://www.gromos.net/>

PREVIEW VERSION

CHAPTER 2

Block Structure and Title Record of GROMOS Files

GROMOS files are composed of a sequence of blocks, which may be of different type. A block begins with a line (record) containing the *blockname* or *blockidentifier* beginning in the first position. The block ends with a line (record) containing the character string *END* beginning in the first position. A blockname or blockidentifier is a unique string of maximally 25 characters. It may not contain a # symbol in the first position and may not be an END string. Block names are given in upper case only. The currently defined blocknames and their functions are listed in Chap. 12.

Each input or output file of the programs PROMD or MD++, which execute a simulation, starts with a *Title block* (Blockname: TITLE), which may contain any character type of data and is meant to specify the contents of the file. When PROMD or MD++ reads a file, the title record is always printed in order to check whether the wanted file has been assigned to a specific (reading) unit number. This convention is not followed by GROMOS++, which relies on file names rather than units, and ignores title blocks.

Generally, blocks may be listed in any order. However, when there are obvious dependences between data in different blocks, e.g. the definition of bond-angle types and sequence of bond-angles of a molecule, a specific order is required (the latter after the former).

Data files may contain *comment lines*, which may occur at any position and in any number. A comment line is recognized by the # symbol *in the first position* of the line. GROMOS++ also recognizes as comment any text following a # symbol anywhere on the line. In GROMOS, files are *written* using *fixed format* and are *read* using *free format*.

Finally, note that PROMD will accept record lines (within the blocks) up to a length of 90 characters (the occurrence of a longer line will result in an error message).

This is only a preview version of the GROMOS Manual and User Guide.
The full version can be purchased on <http://www.gromos.net/>

PREVIEW VERSION

CHAPTER 3

Topological Information

3.1. Introduction

A *molecular topology* file contains information about the topology of a molecular system. In its simplest form it would contain lists of covalent bonds, angles, masses, charges, etc. for all the atoms in the molecular system. When the system contains topologically identical molecules, like water molecules in an aqueous solution or corresponding molecules in different asymmetric units in a unit cell in a crystal, these atom lists would contain redundant information. For PROMD, the topological information of topologically identical solute molecules need only be stored for one molecule. For MD++ and GROMOS++, however, this option has been removed, and the topology has to be specified explicitly for all identical solute units. Since a solvent generally consists of simple molecules like H₂O or CCl₄, it would generally be advantageous to avoid the overhead of handling the possibility of occurrence of internal dihedral angle degrees of freedom, non-bonded interactions, etc. within a solvent molecule. Therefore, a distinction is made between a general part (solute) and a more restricted part (solvent) of a molecular topology file.

For historical reasons the *general part* of a molecular topology file is denoted by the notation “*solute*” *molecular topology*, although it may contain any collection of molecules including solvent molecules. The *restricted part* of a molecular topology file is denoted by the notation “*solvent*” *molecular topology*. In general, this part contains topological data on a single type of solvent molecule, unless a solvent molecule does not fit with the following *restrictions*:

- a solvent molecule must be rigid: no internal interactions like bond-stretching, bond-angle bending, (improper) dihedral torsion and non-bonded interactions are allowed,
- the internal structure of a solvent molecule is maintained by application of distance constraint forces between its atoms,
- a solvent molecule consists of one charge group, the position of the first atom of a solvent molecule is taken to represent the position of this charge group,
- a solvent molecule corresponds to a single “temperature group” and a single “pressure (virial) group”,
- the residue or molecule name cannot be specified, it is predefined as SOL,
- position restraining should only be applied to the first atom of a solvent molecule,
- fixed position constraints cannot be applied to solvent atoms,
- distance restraining cannot be applied to atoms of solvent molecules,
- solvent parameters cannot be changed using a molecular topology perturbation file for obtaining free energy differences.

If a solvent molecule does not comply with these rules, its topological data must be included in the general or solute part of the molecular topology file.

A molecular topology file often contains fewer atoms than a corresponding configuration file. Let us assume that the former contains a set of molecules forming a “solute” of NPM atoms and a solvent molecule with NRAM atoms. In order to *match* this *molecular topology file*, a *configuration file* must contain the following sequence of atoms (for each block of atomic quantities):

1. if the molecular system contains NPM “solute” molecules, the atomic coordinates or related quantities of the NPM*NRP “solute” atoms,
2. if the molecular system contains NSM solvent molecules, the atomic coordinates or related quantities of the NSM*NRAM solvent atoms.

Solvent coordinates always appear after solute coordinates in the various blocks of configuration files. The “solute” and solvent parts of the molecular topology file are to be chosen as the smallest topologically identical units of each type. Note that in MD++ and GROMOS++ NPM is restricted to the value 1, so all

solute atoms should be included explicitly in the molecular topology file.

Multiple Lennard-Jones interactions, bonds, angles, dihedrals, improper dihedrals, proper dihedrals and cross-dihedrals are ordered in such a way that atom *a* of object *u-1* is always smaller than atom *a* of object *u*.

In Sec. 3.2 the content of a molecular topology file is specified. This information is always kept in formatted form. In some applications of GROMOS, like calculating the free energy difference between two different states A and B of a system, it is required to change the molecular topology of the system from one corresponding to state A to another one corresponding to state B. In Sec. 3.3 the way a perturbation (change from A to B) of a molecular topology is to be specified will be discussed. The topological specification of atom-atom distance restraints is described in Sec. 3.4. The topological specification of dihedral angle restraints or constraints is described in Sec. 3.5. The topological definition of ³*J*-value restraints is described in Sec. 3.6. The topological specification of dihedral angles used in the local-elevation search technique is described in Sec. 3.8. The following three sections contain atomic property specifications: friction coefficients and position restraining or fixing indicators (Sec. 3.9-Sec. 3.11).

3.2. Molecular Topology

A molecular topology is characterized by some or all of the following quantities, which are stored in a molecular topology file (the variable names refer to the PROMD program; MD++ and GROMOS++ rely on different data structures, but read the same molecular topology files):

FPEPSI	$(4\pi\epsilon_0)^{-1}$, ϵ_0 = permittivity of vacuum
HBAR	$\hbar = h/(2\pi)$, h = Planck's constant
SPDL	c = speed of light
BOLTZ	k_B = Boltzmann's constant
TPVER	real number characterizing the version of the molecular topology
NRATT	number of (van der Waals) atom types
TYPE[1..NRATT]	names of the different atom types as a function of the integer atom code that defines an atom type (at most 5 characters)
MPAC[1..NRATT,1..NRATT]	pair codes for atom pairs as a function of their integer atom codes I and J (\leq NRATT*(NRATT+1)/2 = NRATT2), the pair code is defined as $I+J*(J-1)/2$ when $I \leq J$ and as $J+I*(I-1)/2$ when $J \leq I$; these values are not stored in the molecular topology file, since they can be and are calculated upon reading the file
NRAA2	number of residues in a solute
AANM[1..NRAA2]	residue names as a function of the residue sequence number (at most 5 characters)
NRP	number of solute atoms
MRES[1..NRP]	residue sequence number of solute atoms (\leq NRAA2)
WMAS[1..NRP]	mass of solute atoms
PANM[1..NRP]	atom name of solute atoms (at most 5 characters)
IAC[1..NRP]	integer atom code of solute atoms, determining the type of van der Waals interaction of an atom (\leq NRATT)
INE12[1..NRP]	number of neighbour atoms that are excluded from the non-bonded interaction with a solute atom

KNE12[1..NRP]	denotes where the atom sequence numbers of the excluded neighbours of a solute atom can be found in array JSNE12 (solute, \leq NAEX12); these values are not stored in the molecular topology file, since they can be and are calculated upon reading the file using $KNE12[1] = 0$ and $KNE12[I+1] = KNE12[I] + INE12[I]$
JSNE12[1..NAEX12]	excluded neighbours (solute, \leq NRP); sequence numbers J of atoms that are excluded from the non-bonded interaction with the atom with sequence number I, are positioned at positions $KNE12[I]+1, \dots, KNE12[I]+INE12[I]$ in array JSNE12; it is assumed that $I < J$ and that the J's appear in ascending order
NAEX12	total number of excluded atoms (solute); this value is also not stored in the molecular topology file, since it can be and is calculated upon reading the file using $NAEX12 = KNE12[NRP] + INE12[NRP]$
INE14[1..NRP]	number of third-neighbour atoms of solute atoms, for which special 1-4 van der Waals interaction parameters are used when evaluating the non-bonded interaction
KNE14[1..NRP]	denotes where the atom sequence numbers of the third neighbours of a solute atom can be found in array JSNE14 (solute, \leq NAEX14); these values are not stored in the molecular topology file, since they can be and are calculated upon reading the file using $KNE14[1] = 0$ and $KNE14[I+1] = KNE14[I] + INE14[I]$
JSNE14[1..NAEX14]	third neighbours (solute, \leq NRP); sequence numbers J of atoms for which the 1-4 van der Waals parameters are used when calculating the non-bonded interaction with the atom with sequence number I, are positioned at positions $KNE14[I] + 1, \dots, KNE14[I]+INE14[I]$ in array JSNE14; it is assumed that $I < J$ and that the J's appear in ascending order
NAEX14	total number of third-neighbour atoms (solute); this value is also not stored in the molecular topology file, since it can be and is calculated upon reading the file using $NAEX14 = KNE14[NRP] + INE14[NRP]$
INE[1..NRP]	total number of excluded and third neighbours of a solute atom (i.e. sum of INE12 and INE14; used internally)
KNE[1..NRP]	denotes where the atom sequence number of the excluded plus third neighbours of a solute atom can be found in array JSNE (used internally)
JSNE[1..NAEX]	excluded plus third neighbours (solute, \leq NRP; i.e. combined list from JNE12 and JNE14; used internally)
CG[1..NRP]	charge of solute atoms; upon reading a molecular topology file by PROMD, the charges are multiplied by $(4\pi\epsilon_0)^{-1/2}$ and stored in CG; this conversion is not performed in MD++ and GROMOS++
NCAG	number of charge groups in a solute; this number is not stored in the molecular topology since it can be and is calculated upon reading the file
ICG[1...NRP]	Atomic charge group codes. The last atom of a charge group is defined by $ICG=1$, the others must have $ICG=0$
INC[1..NCAG]	charge group pointer list; $INC[I]$ specifies the position of the last atom of the I-th charge group in the atom sequence (\leq NRP); the atoms of a charge group must have sequential atom sequence numbers; the values of NCAG and $INC[1...NCAG]$ are not stored in the molecular topology file, but are calculated upon reading the atomic charge group codes $ICG[1...NRP]$, which are stored in the file
NCGB[L]	number of coarse-grained regions
NRCGF[1..NCGB], NRCGL[1..NCGB]	first and last atom sequence number of a coarse-grained region

MSCAL[1..NCGB]	scaling factor for pressure correction of a coarse-grained region
NLJEX	number of LJ-exceptions
ILJEX, JLJEX[1..NLJEX]	atom sequence numbers of atoms i and j to interact with special LJ-interactions given by LJ-exceptions, i is always smaller than j
LJEXC12[1..NLJEX]	coefficient of the $1/r^{12}$ term in the non-bonded interaction for the corresponding atom pair
LJEXC6[1..NLJEX]	coefficient of the $1/r^6$ term in the non-bonded interaction for the corresponding atom pair
NBTY	number of covalent bond types
CB[1..NBTY]	force constant of the bond-stretching term of the interaction as a function of the bond-type code, based on a quartic potential
CHB [1..NBTY]	force constant of the bond-stretching term of the interaction as a function of the bond-type code, based on a harmonic potential
BO[1..NBTY]	bond length at minimum energy of the bond-stretching term as a function of the bond-type code
NBONH	number of bonds involving H-atoms in the solute
IBH, JBH[1..NBONH]	atom sequence numbers of the atoms forming a bond i-j as a function of the bond sequence number (\leq NRP), i is always smaller than j
ICBH[1..NBONH]	bond-type code as a function of the bond sequence number (\leq NBTY)
NBON	number of bonds <i>NOT</i> involving H-atoms in the solute
IB, JB[1..NBON]	atom sequence numbers of the atoms forming a bond i-j as a function of the bond sequence number (\leq NRP), i is always smaller than j
ICB[1..NBON]	bond-type code as a function of the bond sequence number (\leq NBTY)
NBONCG	number of bonds involving coarse grained particles in the solute
IBCG, JBCG[1..NBONCG]	sequence numbers of the coarse grained particles forming a bond i-j as a function of the bond sequence number (\leq NRP)
ICBCG[1..NBON]	bond-type code as a function of the bond sequence number (\leq NBTY)
NTTY	number of bond-angle types
CT[1..NTTY]	force constant of the bond-angle bending term of the interaction as a function of the bond-angle type code, based on a potential harmonic in the angle cosine
CHT[1..NTTY]	force constant of the bond-angle bending term of the interaction as a function of the bond-angle type code, based on a potential harmonic in the angle (in energy units per degree ²); upon reading a molecular topology file by PROMD, the force constant is converted to energy per rad ² and stored in CHT
TO[1..NTTY]	bond angle (in degrees) at minimum energy of the bond-angle bending term as a function of the bond-angle type code; upon reading a molecular topology file by PROMD or MD++, the bond angle is converted from degrees to radians; this conversion is not performed in GROMOS++
NTHEH	number of bond-angles involving H-atoms in the solute

ITH, JTH, KTH [1..NTHEH]	atom sequence numbers of the atoms forming a bond-angle i-j-k as a function of the bond-angle sequence number (\leq NRP), i is always smaller than k
ICTH [1..NTHEH]	bond-angle type code as a function of the bond-angle sequence number (\leq NTTY)
NTHE	number of bond-angles <i>NOT</i> involving H-atoms in the solute
IT, JT, KT [1..NTHE]	atom sequence numbers of the atoms forming a bond-angle i-j-k as a function of the bond-angle sequence number (\leq NRP), i is always smaller than k
ICT [1..NTHE]	bond-angle type code as a function of the bond-angle sequence number (\leq NTTY)
NQTY	number of improper (harmonic) dihedral-angle types
CQ [1..NQTY]	force constant of the harmonic improper dihedral term of the interaction as a function of the improper dihedral-angle type code (in energy units per degree ²); upon reading a molecular topology file by PROMD or MD++, the force constant is converted to energy per rad ² and stored in CQ; this conversion is not performed by GROMOS++
Q0 [1..NQTY]	improper dihedral (in degrees) at minimum energy of the harmonic improper dihedral term as a function of the improper dihedral-angle type code; upon reading a molecular topology file by PROMD or MD++, the improper dihedral angle is converted from degrees to radians and stored in Q0; this conversion is not performed by GROMOS++
NQHIH	number of improper dihedrals involving H-atoms in the solute
IQH, JQH, KQH, LQH [1..NQHIH]	atom sequence numbers of the atoms forming improper dihedral i-j-k-l as a function of the improper dihedral sequence number (\leq NRP), j is always smaller than k
ICQH [1..NQHIH]	improper dihedral type code as a function of the improper dihedral sequence number (\leq NQTY)
NQHI	number of improper dihedrals <i>NOT</i> involving H-atoms in the solute
IQ, JQ, KQ, LQ [1..NQHI]	atom sequence numbers of the atoms forming improper dihedral i-j-k-l as a function of the improper dihedral sequence number (\leq NRP), j is always smaller than k
ICQ [1..NQHI]	improper dihedral type code as a function of the improper dihedral sequence number (\leq NQTY)
NPTY	number of (trigonometric) dihedral-angle types
CP [1..NPTY]	force constant of the trigonometric dihedral term of the interaction as a function of the dihedral-angle type code
PDL [1..NPTY]	phase-shift angle (in degrees) of the trigonometric dihedral term as a function of the dihedral-angle type code; upon reading a molecular topology file by PROMD or MD++, the phase-shift angle is converted from degrees to radians and stored in PDL; MD++ stores the phase-shift angle as a cosine in PDL. These conversion are not performed in GROMOS++
PD [1..NPTY]	corresponding phase-shift cosine (used internally by PROMD)
NP [1..NPTY]	multiplicity of the trigonometric dihedral term as a function of the dihedral-angle type code (1, 2, 3, 4, 5, or 6)
NPHIH	number of dihedrals involving H-atoms in the solute

IPH, JPH, KPH, LPH [1..NPHIH]	atom sequence numbers of the atoms forming dihedral i-j-k-l as a function of the dihedral sequence number (\leq NRP), j is always smaller than k
ICPH [1..NPHIH]	dihedral type code as a function of the dihedral sequence number (\leq NPTY)
NPHI	number of dihedrals <i>NOT</i> involving H-atoms in the solute
IP, JP, KP, LP [1..NPHI]	atom sequence numbers of the atoms forming dihedral i-j-k-l as a function of the dihedral sequence number (\leq NRP), j is always smaller than k
ICP [1..NPHI]	dihedral type code as a function of the dihedral sequence number (\leq NPTY)
NPPCH	number of cross-dihedrals involving H-atoms in the solute
APH, BPH, CPH, DPH, EPH, FPH, GPH, HPH [1..NPPCH]	atom sequence numbers of the atoms forming cross-dihedrals a-b-c-d and e-f-g-h as a function of the dihedral sequence numbers of the separate dihedrals (\leq NRP), a,b,c,d are always smaller or equal to e,f,g,h respectively
ICCH [1..NPPCH]	dihedral type code as a function of the cross-dihedral sequence number (\leq NPTY)
NPPC	number of dihedrals <i>NOT</i> involving H-atoms in the solute
AP, BP, CP, DP, EP, FP, GP, HP [1..NPPC]	atom sequence numbers of the atoms forming cross-dihedrals a-b-c-d and e-f-g-h as a function of the dihedral sequence numbers of the separate dihedrals (\leq NRP), a,b,c,d are always smaller or equal to e,f,g,h respectively
ICC [1..NPPC]	dihedral type code as a function of the cross-dihedral sequence number (\leq NPTY)
NRATT2	number of unique pairwise combinations of atom types (=NRATT*(NRATT +1)/2)
C12 [1..NRATT2]	coefficient of the $1/r^{12}$ term in the non-bonded interaction as a function of the occurring pair codes; so, the sequence of atom pairs with integer atom codes ranging from 1 to NRATT is: 1-1, 1-2, ..., 1-NRATT, 2-2, 2-NRATT, ..., NRATT-NRATT
C6 [1..NRATT2]	coefficient of the $-1/r^6$ term in the non-bonded interaction as a function of the occurring pair codes
CS12 [1..NRATT2]	coefficient of the $1/r^{12}$ term in the 1-4 non-bonded interaction between third-neighbour atoms as a function of the occurring pair codes
CS6 [1..NRATT2]	coefficient of the $-1/r^6$ term in the 1-4 non-bonded interaction between third-neighbour atoms as a function of the occurring pair codes
NPPOL	number of polarisable solute atoms (\leq NRP)
IPOLP [1..NPPOL]	atom sequence numbers of the polarisable solute atoms (\leq NRP)
ALPP [1..NPPOL]	polarisabilities of solute atoms IPOLP[1..NPPOL]
QPOLP [1..NPPOL]	size of charge-on-spring connected to polarisable solute atoms IPOLP[1..NPPOL]
ENOTP [1..NPPOL]	damping level for polarisation of solute atoms IPOLP[1..NPPOL]
EPP [1..NPPOL]	damping parameter for polarisation of solute atoms IPOLP[1..NPPOL]
NSPM	number of all separate (covalently linked) molecules within the solute topology (e.g. separate protein chains, co-solute molecules, counterions, co-solvent molecules)

NSP [1 .. NSPM]	atom sequence number of the last atom of the successive submolecules (\leq NRP)
NSTM	number of temperature atom groups (used to separate translational from internal-plus-rotational velocity components for kinetic energy evaluation and thermostatting) within the solute topology
NST [1 .. NSTM]	atom sequence number of the last atom of the successive temperature atom groups (\leq NRP)
NSVM	number of pressure (virial) atom groups (used to define a group-based pressure) within the solute topology
NSV [1 .. NSVM]	atom sequence number of the last atom of the successive pressure (virial) atom groups (\leq NRP)
NRAM	number of atoms per solvent molecule
WMASS [1 .. NRAM]	mass of solvent atoms
ANMS [1 .. NRAM]	atom name of solvent atoms (at most 5 characters)
IACS [1 .. NRAM]	integer atom code of solvent atoms determining the type of van der Waals interaction of an atom (\leq NRATT)
CGS [1 .. NRAM]	charge of solvent atoms; upon reading a molecular topology file by PROMD, the charges are multiplied by $(4\pi\epsilon_0)^{-1/2}$ and stored in CGS. This conversion is not performed in MD++ and GROMOS++.
NCONS	number of distance constraints within a solvent molecule
ICONS, JCONS [1 .. NCONS]	atom sequence numbers of the atoms forming the constraint i-j as a function of the constraint sequence number (\leq NRAM), i is always smaller than j
CONS [1 .. NCONS]	constraint length as a function of the constraint sequence number; upon reading a molecular topology file by PROMD, the constraint length is squared and stored in CONS
NVPOL	number of polarisable solvent atoms (\leq NRAM)
IPOLV [1 .. NVPOL]	atom sequence number of the polarisable solvent atoms (\leq NRAM)
ALPV [1 .. NVPOL]	polarisabilities of solvent atoms IPOLV[1..NVPOL]
QPOLV [1 .. NVPOL]	size of charge-on-spring connected to polarisable solvent atoms IPOLV[1..NVPOL]
ENOTV [1 .. NVPOL]	damping level for polarisation of solvent atoms IPOLV[1..NVPOL]
EPV [1 .. NVPOL]	damping parameter for polarisation of solvent atoms IPOLV[1..NVPOL]
NPID	number of path-integral (PI) discretizations (pseudoparticles or beads) per real atom
TPI	temperature value that has been used to compute the path-integral harmonic spring force constants CPI
IPIC [1 .. NRP]	path-integral atom type and sequence identification number ($0 \leq$ IPIC \leq NPID) as a function of the atom sequence number; zero for classical real atom; positive for atom being quantum bead with bead sequence number IPIC (\leq NPID)
NPIT	number of path-integral harmonic spring types
CPI [1 .. NPIT]	force constant of the path-integral harmonic spring term of the interaction function as a function of the path-integral harmonic spring type code

NPIB number of path-integral harmonic spring bonds between atoms (i.e. beads)

IPIB, JPIB[1..NPIB] atom sequence numbers of the atoms (i.e. beads) forming a harmonic spring bond i-j as a function of the path-integral harmonic spring bond sequence number (\leq NRP)

ICPIB[1..NPIB] path-integral harmonic spring type code as a function of the path-integral harmonic spring bond sequence number (\leq NPIT)

NPIA number of quantum atoms (to be discretized)

WMCL [1..NPIA] real, physical mass of the quantum (discretized) atoms

NRSASAA number of atoms to be considered for SASA implicit solvent interaction function

ISASA[1..NRSASAA] atom sequence numbers of the atoms to be included in the SASA implicit solvent interaction function

RADI[1..NRSASAA] atomic radii of the SASA atoms

PI[1..NRSASAA] atom type-specific parameters for reduction in SASA

SIGMAI[1..NRSASAA] atom type-specific scaling parameters for SASA energy term

The blocks of a *molecular topology file* are (apart from the *Title block*) the following:

Physical constants block

Blockname: PHYSICALCONSTANTS

```
WRITE (unit,12) FPEPSI
WRITE (unit,12) HBAR
WRITE (unit,12) SPDL
WRITE (unit,12) BOLTZ
12  FORMAT (E15.7)
```

This block replaces the TOPPHYSCON block of GROMOS96.

Version block

Blockname: TOPVERSION

```
WRITE (unit,13) TPVER
13  FORMAT (F3.1)
```

The version number expected by GROMOS is 2.0.

Van der Waals atom type sequence and name block

Blockname: ATOMTYPENAME

```
WRITE (unit,14) NRATT
DO 10 K=1, NRATT
10  WRITE (unit,15) TYPE[K]
14  FORMAT (5I5)
15  FORMAT (A5)
```